X-th International Conference

Knowledge-Dialogue-Solution

June 16-26, 2003, Varna (Bulgaria)

P R O C E E D I N G S

FOI-COMMERCE

SOFIA, 2003

Gladun V.P., Kr.K. Markov (editors)

Proceedings of the X-th International Conference “Knowledge-Dialogue-Solution” – Varna, 2003

Sofia, FOI-COMMERCE – 2003

ISBN: 954-16-0025-5
First Edition

The X-th International Conference “Knowledge-Dialogue-Solution” (KDS 2003) continues the series of annual international KDS events organized by Association of Developers and Users of Intelligent Systems (ADUIS).

The conference is traditionally devoted to discussion of current research and applications regarding three basic directions of intelligent systems development: knowledge processing, natural language interface, and decision making.

Edited by :
Association of Developers and Users of Intelligent Systems, Ukraine
Institute of Information Theories and Applications FOI ITHEA, Bulgaria
Printed in Bulgaria by FOI-COMMERCE
Sofia-1090, P.O.Box 775, Bulgaria
e-mail: foi@nlcv.net
www.foibg.com

ISBN 954-16-0025-5
C\o Jusautor, Sofia, 2003

PREFACE

This volume results from the scientific Tenth International Conference “Knowledge-Dialogue-Solution” which took place in June, 16-28, 2003 in Varna, Bulgaria.

Reports contained in the Proceedings correspond to scientific trends, which are reflected in the Conference name.

The Conference continues the series of international scientific meetings, which were initiated more then ten years ago. It is organized owing to initiative of Association of Developers and Users of Intelligent Systems (Ukraine) and IJ ITA - International Journal on Information Theories and Applications (Bulgaria), which have long-term experience of collaboration.

Now we can affirm that international conferences “Knowledge-Dialogue-Solution” in a great degree contributed to preservation and development of the scientific potential in the East Europe.

The conference is traditionally devoted to discussion of current research and applications regarding three basic directions of intelligent systems development: knowledge processing, natural language interface, and decision making.

The basic approach, which characterizes presented investigations, consists in the preferential use of logical and linguistic models. This is one of the main approaches uniting investigations in Artificial Intelligence. During the past few years new conceptions and methods emerged, particularly in such fields as intelligent agents, intelligent NL-text processing, neural networks, data mining and knowledge discovery. The proceedings deal with some of them.

KDS 2003 topics of interest include, but are not limited to:

- Decision making models and problems
- Informatization of scientific research
- Intelligent agents
- Intelligent NL text processing
- Intelligent robots
- Intelligent technologies of design
- Knowledge-based society
- Knowledge engineering
- Logical inference
- Machine learning
- Neural and growing pyramidal networks
- Philosophy and methodology of informatics
- Planning and Scheduling
- Problems of computer intellectualization
- Text, data mining and knowledge discovery

The official languages of the Conference are English, Russian, and Bulgarian. Sections as well as the papers in the sections are in alphabetical order. The Program Committee recommends the accepted papers for free publishing in English in the International Journal on Information Theories and Applications (IJ ITA).

The Conference was sponsored by FOI Bulgaria (www.foibg.com).

We appreciate the contribution of the members of the KDS 2003 Program Committee.

On behalf of all the conference participants we would like to express our sincere thanks to everybody who helped to make conference success and especially to N. Vashchenko and Kr. Ivanova.

V.P.Gladun, Kr.K. Markov

CONFERENCE ORGANIZERS

Ministry of Education and Science of Ukraine
National Academy of Sciences of Ukraine
IJ ITA - International Journal on Information Theories and Applications (Bulgaria)
Association of Developers and Users of Intelligent Systems
Russian Association for Artificial Intelligence
V.M.Glushkov Institute of Cybernetics of National Academy of Sciences of Ukraine
Institute of Artificial Intelligence of National Academy of Sciences of Ukraine
Institute of Mathematics of SD RAN (Novosibirsk)
Institute of Mathematics and Informatics BAS (Bulgaria)
Institute of Information Theories and Applications FOI ITHEA - (Bulgaria)
IEEE Joint Chapter in Bulgaria - IM/CS/SMC
Federation of Scientific-Technical Unions (Bulgaria)

PROGRAM COMMITTEE

Victor Gladun (Ukraine) - chair
Rumyana Kirkova (Bulgaria)- co-chair
Anatoly Shevchenko (Ukraine) - co-chair

Alexander Eremeev (Russia)
Natalia Filatova (Russia)
Konstantin Gaindrik (Moldova)
Tatyana Gavrilova (Russia)
Krassimira Ivanova (Bulgaria)
Natalia Ivanova (Russia)
Vladimir Jotsov (Bulgaria)
Julia Kapitonova (Ukraine)
Vladimir Khoroshevsky (Russia)
Valery Koval (Ukraine)
Oleg Kuznetsov (Russia)
Vitaliy Lozovskiy (Ukraine)
Krassimir Markov (Bulgaria)
Alfredo Milani (Italy)
Ilia Mitov (Bulgaria)
Xenia Naidenova (Russia)
Olga Nevzorova (Russia)
Genady Osipov (Russia)
Alexander Palagin (Ukraine)
Vladimir Pasechnik (Ukraine)
Yurie Pecherskii (Moldova)
Zinoviy Rabinovich (Ukraine)
Galina Rybina (Russia)
Vladimir Ryazanov (Russia)
Vasil Sgurev (Bulgaria)
Vladislav Shelepov (Ukraine)
Dimitar Shishkov (Bulgaria)
Ekaterina Solovyova (Ukraine)
Vadim Stefanuk (Russia)
Tatyana Taran (Ukraine)
Valery Tarasov (Russia)
Adil Timofeev (Russia)
Jury Valkman (Ukraine)
Neonila Vashchenko (Ukraine)
Alexey Voloshin (Ukraine)
Stanislav Wrycza (Poland)
Nikolay Zagoruiko (Russia)
Larissa Zainutdinova (Russia)
Arkady Zakrevskij (Belarus)

ORGANISING COMMITTEE

Krassimir Markov (Bulgaria) - chair
Neonila Vashchenko (Ukraine) - co-chair

Krassimira Ivanova (Bulgaria)
Natalia Ivanova (Russia)
Ilia Mitov (Bulgaria)
Olexandr Tkachov (Ukraine)

TABLE OF CONTENTS

Section 3 Intelligent Networks and Agents

Section 4 Intelligent Technologies

Section 1: Computer Intellectualization

АНАЛИЗ ПРОБЛЕМЫ СОЗДАНИЯ НОВЫХ ТЕХНИЧЕСКИХ СРЕДСТВ ДЛЯ РЕАЛИЗАЦИИ ЛИНГВИСТИЧЕСКОГО ИНТЕРФЕЙСА

М.Ф.Бондаренко, А.В.Карпухин, Г.Г.Четвериков

Аннотация: *Изложены принципы и новые подходы к построению многовходовых универсальных k-значных параллельных (пространственных) структур для автоматизированной обработки текстовой информации в системах искусственного интеллекта, а также представлена функционально-ориентированная система морфологического анализа с k-значным кодированием данных.*

Ключевые слова: *искусственный интеллект, многозначные элементы, многозначные структуры, универсальный многозначный функциональный преобразователь, морфологический анализ, морфологическая информация, словоформа, базовая машина, функционально ориентированная система.*

Задача интеллектуализации цифровых сетей и языковых систем

Развитие вычислительной техники является базой автоматизации умственной деятельности, и потому возникло новое понятие искусственного интеллекта (ИИ). Тем не менее, успехи в области интеллектуализации вычислительной техники незначительные, в особенности, если сравнивать достигнутое с ожидаемыми результатами и прогнозами. Приходится признавать, что термин „искусственный интеллект” отображает пока еще невыполнимые ожидания. Основные проблемы, перед которыми остановились ученые на первых этапах создания ЭВМ в 50-х годах нашего столетия, до этого времени еще не решены. Если машины обречены на интеллект, то мыслить они будут по тем же законам, что и человек. Нейрофизиологические исследования естественного интеллекта мозга показывают наличие в нем механизмов многозначного (k-значного) кодирования и пространственного характера активности сетей нервных клеток и организации мозговых механизмов.

Для создания соответствующих ИИ k-значных структур необходим новый подход, новая теоретическая база их построения, а для этого недостаточно исследований только в области одной науки. То есть, исследования отдельно в методах синтеза, кодирования, комбинаторики, надежности и точности не дают ответа на вопрос оптимального пути построения соответствующих по свойствам многозначных структур. Дело в том, что все перечисленные теории образовывают замкнутые математические системы и те истины, которые ими порождаются, не дают всестороннего подхода к построению оптимальных многозначных структур, и создается множество истин, явлений и факторов, которые поддаются познанию и пониманию, но которые не дают ответа на кардинальный вопрос, как и для каких задач можно и необходимо создавать многозначные структуры.

Все это порождает новые и непреодолимые трудности, в частности, радикальные изменения архитектуры вычислительных систем с элементами параллелизма, при условиях сохранения в основе неймановского процессора, не позволили существенно увеличить быстродействие и перейти к решению задач построения и реализации высокопроизводительных систем ИИ; повышение быстродействия во время обработки отдельных символов и макро- и примитивных операций, при условиях применения двузначного кодирования и вычислительного характера действия, не привело к решению задач интеллектуального уровня в необходимом объеме; тяжело разработать эффективные методы параллельного оперирования

знаниями, а не данными; невозможно решить задачи создания интеллектуального интерфейса и семантического расслоения знаний об объектах механизмов логического вывода [1, 2, 5].

Таким образом, можно сформулировать основные требования относительно свойств структур и элементов для построения новейших высокоэффективных систем ИИ. Они должны реализовать компараторные функции многозначной логики и кодирования, а также владеть свойствами универсальности, пространственности, гибридности, гибкой переналадки без изменения структуры, иерархичности, по сложности быть сопоставимыми со сложностью решаемых задач. Анализ показывает, что ближайшими в соответствии с указанными свойствами являются многозначные универсальные пространственные элементы и структуры, и развитию таких средств уделяется большое внимание во всем мире.

Основные направления разработки современной элементной базы информатики ориентированы на повышение производительности и снижение стоимости радиоэлектронных и информационно-вычислительных систем и сетей, благодаря развитию полупроводниковой технологии: увеличение функциональной плотности и меры интеграции радиоэлектронных схем. Но предельные физико-технические показатели относительно функциональной плотности и энергопотребления и рассеивания тепла достигают своего практического предела, и поэтому одним из возможных выходов из этой ситуации является создание и применение k-значных элементов и структур. Ведь в соответствии с одним из основных законов кибернетики, законом необходимого разнообразия (сложности), для нормальной работы управляемой системы, при которой обеспечивается полное использование ее потенциала, необходимо, чтобы разнообразие (сложность) управляющей системы было не меньшим, чем разнообразие управляемого объекта. Отсюда и вывод: двузначная элементная и структурная база не отвечает по сложности возлагаемым на нее задачам ИИ и подлежит дополнению универсальной k-значной элементной и структурной базой, которая по своей сложности (разнообразию) стоит значительно выше.

Предложено значительное количество подходов и методов построения и применения многозначных структур, но, тем не менее, отсутствуют их систематизация и упорядоченная система средств реализации, недостаточно обработаны принципы их построения и методы количественной и качественной оценки применения во время создания систем ИИ, что свидетельствует о недостаточном уровне развития теории построения таких структур. Дальнейший прогресс существенно зависит от обобщения и систематизации на единой методологической основе накопленного опыта, развития и усовершенствования системы понятий, которые должны согласовываться с теорией интеллекта.

Модель развития науки BRETAM, которую предложил в начале 70-х годов профессор Чикагского университета Д.Грейн, предусматривает шесть периодов развития науки: 1) прорыв (Brake); 2) копирование (Repeat); 3) эмпирика (Empirical); 4) теория (Theory); 5) автоматизация (Automation); в) зрелость (Mature). На современном этапе в области создания k-значных структур мы имеем все признаки прохождения исследователями 1-3 этапов, и настало время перехода к созданию теории обобщающих принципов построения структур, моделей, законов и методов исследований. В частности, необходимы обоснованные методы и принципы построения универсальных многозначных пространственных структур с соответствующей их формализацией.

Многозначными (k-значными) называют структуры цифровых и радиоэлектронных систем обработки информации, которые образованы множеством k-значных элементов и множеством соответствующих связей. Вычислительная и кибернетическая техника, программирование и робототехнические системы, цифровые сети и их протоколы, математическое моделирование процессов в больших интегрированных схемах, автоматика и телевидение, связь и радиолокация – вот далеко неполный перечень областей науки и техники, где сегодня используются k-значная логика, кодирование, элементы и структуры, хотя и в завуалированном виде из-за того, что отображение данных при этом является двузначным.

Основные положительные эффекты от применения k-значных элементов и структур можно свести к следующему: создание систем искусственного интеллекта, способных к самоорганизации и самопрограммированию, решение сверхсложных задач распознавания образов языковых и зрительных изображений; создание систем помехоустойчивого кодирования и защиты от несанкционированного доступа с применением теории конечных полей, которые являются по сути k-значными; развитие нового подхода к созданию высокоэффективной поточно-пространственной архитектуры систем с элементами искусственного интеллекта, адекватной к сложности выполняемых ими задач; упрощение структуры

цифровых устройств обработки данных за счет отсутствия потребности промежуточных преобразований десятичных чисел в двоичную форму и существенное увеличение скорости выполнения арифметических операций; уменьшение аппаратурных затрат за счет уменьшения длины кодовых изображений данных с ростом значительности и, как следствие, снижение стоимости и энергопотребления; рост производительности цифровых систем и ЭВМ за счет сокращения времени выполнения таких непроизводительных операций, как выравнивание порядков и нормализация; уменьшение числа связей на функциональном и системном уровнях и, как следствие, повышение надежности устройств передачи цифровых данных; создание высокоэффективных методов и средств аналого-цифрового преобразования; создание методов моделирования элементов и структур с совмещением процессов логического моделирования и количественного анализа (на основе большей детализации изображения формы реального физического сигнала); обеспечение более высокой скорости передачи цифровых сигналов в границах заданной полосы частот; оптимизация программ в соответствии с заданными критериями с использованием k-значных алгебр Поста и т.п.

И вместе с тем практические достижения в данной области поражающе бедны в сравнении с важностью проблемы и огромными усилиями, затраченными на протяжении трети века на их реализацию. Очень мало создано технических средств, которые оказались пригодными для решения реальных задач автоматизации работы человека. Причины такого положения не переосмыслены и на сегодняшний день.

На наш взгляд, дело в том, что в подходах к анализу и синтезу k-значных элементов и структур существует ряд концептуальных ошибок. Ошибочной является точка зрения, что для k-значных структур $k \geq 3$. Практически же $k \in \{0,1,2,..., k-1\}$, поэтому значность k=2 существует в непрерывной связи с любой другой значностью. Это очевидное и простое обстоятельство приводит нас к следующему неочевидному выводу, что не существует необходимости альтернативного размежевания и противопоставления дву- и k-значных элементов и структур, и наоборот, следует искать подходы, которые распространяли бы свойства сосуществования (симбиоза) этих логик на элементный и структурный уровни.

Во-вторых, присвоение k-значной структурой значений алфавита из множества $E_k \in \{0,1,..., k-1\}$ осуществляется с помощью многоуровневых сигналов. Распознавание значений уровней во время работы k-значных структур фактически сводится к измерению некоторого физического параметра X (например, напряжения, тока, электрического заряда и т.п.). Для того чтобы k-значный элемент или структура могли безошибочно распознавать отображенные сигналами значения алфавита, они должны измерять их с определенной мерой точности.

Под точностью измерения или формирования k-значных сигналов понимают интервал допустимых отклонений, в котором с установленной вероятностью находится их суммарная погрешность. Исследования [1, 4, 5] подтвердили, что главнейшей задачей во время создания k-значных структур является учет и обеспечение требований относительно точности их работы. Преобладающее большинство разработок k-значных структур базировалось на неадекватном теоретическом фундаменте двузначных элементов и структур, который вообще не предусматривает во время обработки сигналов их измерение с заданной точностью. Многие из исследователей, которые занимались проблемой создания и применения k-значных элементов и структур, и сегодня стараются создать функционально полные наборы предельно простых элементов для задач форматного синтеза цифровых систем с k-значным структурным алфавитом по аналогии с тем, как это принято в двузначных методах и подходах. В результате нет ни элементов, ни структур, ни систем.

В современных цифровых системах преимущественно используется двузначное кодирование, но поскольку объем передаваемых данных неуклонно возрастает, перед разработчиками становится задача повышения пропускной способности двузначных каналов обмена данными. Одним из путей ее решения является распараллеливание каналов передачи вплоть до побитной передачи каждым из каналов. При этом, чем больший объем данных необходимо передать, тем большее число связей необходимо иметь, а это приводит к увеличению веса, габаритов и стоимости аппаратуры, снижает ее надежность. Поэтому самым перспективным является использование пространственных (параллельных) схем, структур и систем k-значной логики, которые обеспечивают построение быстродействующих средств обработки информации и владеют высшими показателями относительно пропускной способности при условиях меньшего числа связей и компонент, чем двузначные.

Многозначные логические элементы (МЛЭ) [1, 4, 5] по своим структурным построениям и принципам действия являются преобразователями информационных сообщений, которые характеризуются

определенными информационными признаками. Если виды входного и выходного информационных признаков совпадают, то преобразователь называют однородным. Задача создания однородного преобразователя решается с использованием промежуточного преобразования, которое осуществляет переход от одного информационного признака к другому, используя элементарные неоднородные преобразователи.

Перспективной при этом считается следующая группа информационных признаков: S - статический признак (каждому из символов многозначного структурного алфавита ставится в соответствие один из уровней напряжения или тока); P - пространственный признак (символы алфавита отображаются возбужденным состоянием одного из k пространственных полюсов); D - динамический признак (символам алфавита отвечают определенные интервалы времени для выбранного вида периодических последовательностей импульсов). Для этой группы информационных признаков сообщений существует только 3 элементарных неоднородных преобразователя:

$$S \rightarrow D;\ D \rightarrow S;\ P \rightarrow S;$$
$$S \rightarrow P;\ D \rightarrow P;\ P \rightarrow D.$$

Изначально, самыми перспективными, в отношении простоты схемной реализации, были элементы, которые строятся по структуре $S \rightarrow D - D \rightarrow S$, но эти же элементы, к сожалению, наименее быстродействующие. С другой стороны, с усовершенствованием твердотелой интегральной схемотехники и технологии число компонент (вентилей) не играет преобладающей роли, и на первое место выходит требование обеспечения высокого быстродействия МЛЭ. Поэтому, в дальнейшем, статические пространственные k-значные структуры реализуются по схеме $S \rightarrow P - P \rightarrow S$ как имеющие предельно высокое быстродействие и могут быть реализованы с применением твердотелой технологии.

Во время реализации МЛЭ в соответствии с принципом базиса осуществляется их отладка путем перекоммутации базисных входов к соответствующим выходам источника базисных сигналов. Применение отдельного источника калиброванных базисных сигналов обуславливается необходимостью обеспечения соответствующего уровня точности формирование k-значных сигналов, и, как следствие, необходимой вероятности безотказной работы k-значной структуры в целом, а также возможностью обеспечения универсальности МЛЭ за счет мультиплексирования базисных входов. Хотя с другой стороны, применение k-значной $I^2Л$ - схемотехники и технологии привело к созданию распределенных в структуре БИС и квантованных по уровню тока инжекторов, которые выполняют роль локальных источников базисных сигналов. При этом были утрачены возможности обеспечения необходимой точности и повторяемости k-значных сигналов, и принципиально хорошая идея создания потоковых k-значных схем так и не нашла своего воплощения в промышленных образцах. Более того, исследования пространственных универсальных многозначных функциональных преобразователей (УМФП) потенциального типа, которые проведены в работах [1, 4, 5], также доказали перспективность использования распределенных в структуре параметрических формирователей опорных и базисных сигналов.

Итак, анализ состояния дел в области разработки автоматизированных систем управления с элементами искусственного интеллекта свидетельствует о том, что основными ориентирами на магистральном направлении ИИ является создание быстродействующих (пространственных), универсальных, гибко переналаживаемых структур, и большинством из этих признаков владеют, как раз, пространственные k-значные структуры, в частности комплекс, в состав которого входят преобразователи двузначных кодов в многозначные и наоборот, универсальные многозначные функциональные преобразователи и пространственно-объемные k-значные коммутаторы. Применение k-значной логики и универсальных k-значных структур оправдано также и тем, что языковые и зрительные образы естественного интеллекта многозначны, а его механизмы действия дискретны.

Анализируя состояние дел в теории информации, теории вычислительных систем, теории кодирования, теории автоматических систем управления, теории радиосистем, только в теории информации и кодирования и в k-значной логике находим те первоосновы, которые еще с самого начала, в качестве формальных математических понятий, используют понятия k-значных основ систем исчисления. Поскольку многозначные элементы и структуры по принципам действия являются преобразователями информационных сообщений, которые, как аппаратурные каналы, характеризуются определенными информационными признаками, то для формирования теории многозначных структур и соответствующих их исследований используются математические модели Шеннона каналов передачи данных с k-значным

кодированием при условиях возникновения помех. Итак, теория построения k-значных структур развивается со стыка нескольких дисциплин, ряда теорий k-значной логики и теорий кодирования, точности и чувствительности.

Следует отметить, что одним из самых перспективных направлений применения теории многозначных структур и кодирования в системах искусственного интеллекта на практике является моделирование естественного языка. Этот подход позволяет одновременно с развитием вариантов программной реализации полученных моделей языка осуществить другой подход – схемный. Мозг, при этом, рассматривается как отправная точка построения k-значных пространственных структур языковых систем ИИ.

Таким образом, с появлением и широким применением многозначных универсальных пространственных структур в системах ИИ возникает комплекс взаимосвязанных теоретических, методических и схемотехнических задач их построения и реализации, являющийся сложной проблемой. Ее решение является актуальным и имеет стратегическое значение для выхода из кризисной ситуации во время создания систем ИИ с уменьшением огромных затрат времени и средств финансирования, а краткий анализ состояния свидетельствует об актуальности и большом научном и прикладном значении данной проблематики.

Функционально-ориентированная система морфологического анализа с k-значным кодированием данных

Мыслители цивилизации четко и ясно определили главную проблему и кратчайший путь к созданию систем искусственного интеллекта – моделирование морально-этичных норм и законов. Очевидно, что единственным известным нам объективным носителем морали и интеллекта является человек, а выразителем, средством внешнего общения, носителем является человеческий язык. Поэтому, единственно возможным путем самого объективного, полного и эффективного создания систем ИИ является путь анализа, моделирования и синтеза языкового интеллектуального интерфейса с помощью алгебры конечных предикатов, средств k-значной логической системы и соответствующих структур и кодирования.

Основные понятия и определения. Морфологический анализ (МА) - эта обработка словоформ без связи с контекстом. Словоформа - это отрезок текста между двумя просветами. Разделительные знаки считаются отдельными словоформами. Задачей морфологического анализа является идентификация словоформ и присвоение каждой словоформе комплекса морфологической информации (КМИ). Такой комплекс состоит из морфологично-информационных рядов (строк) (МИ-строк), структура которых такая: номер, <(основа или признаки основы), МИ>, где номер - порядковый номер данной словоформы во фразе; основа (признак основы) - код семантического признака, номер синтаксической или семантической модели управления, которые присвоены данной основе в словаре основ; МИ - часть речи и ее грамматические категории: род, число, падеж, время, лицо и т.п. [1,2].

Алгоритмика. Для анализа украинского языка используется алгоритмический (процедурный) морфологический анализ [1,2]. При МА осуществляется расчленение словоформ на основу и окончание, и в словарях сохраняются как основы, так и их окончания. МА осуществляется путем поиска в составе анализируемой словоформы некоторой словарной основы и определенного словарного окончания. Потом осуществляется сравнение информации об основе и окончании и выдается КМИ для всей словоформы.

Первый шаг. Во время МА изменяемой словоформы ее конечную часть поочередно сравнивают с окончаниями словаря. Если сравнение состоялось, то ту часть словоформы, которая совпала, отделяют и получают допустимую основу (ДОС), допустимое окончание (ДОК) и допустимую морфологическую информацию (ДМИ). Данные о ДОК (ДМИ) считывают из словаря окончаний (морфологической информации). Потом переходят к поиску других ДОК, ДОС и ДМИ.

На втором шаге анализа словоформы выполняется идентификация ее возможных основ путем проверки сходимости полученных ДОС с основами машинного словаря основ.

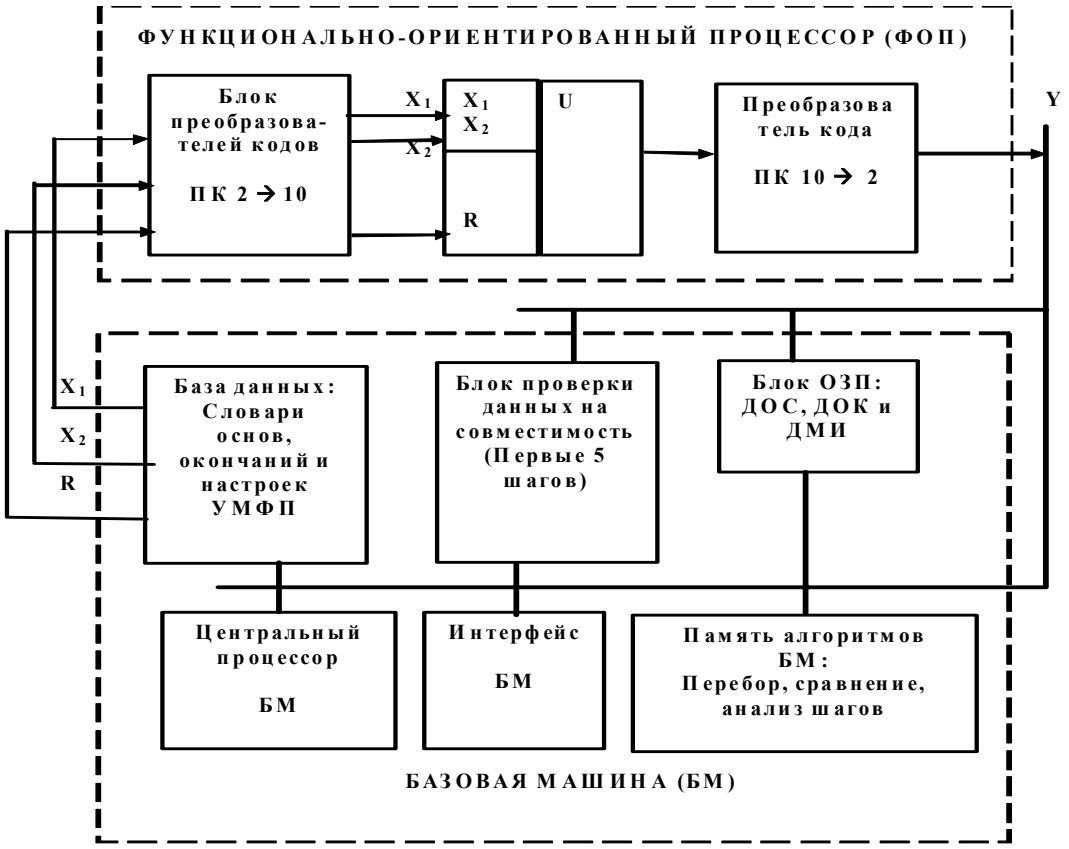
На третьем шаге МА словоформы сравнивается информация с теми ДОС и ДОК, которые получили подтверждение с помощью словаря основ.

Для представления значений грамматических категорий любой словоформы используем 9-ти разрядный 10-значный код. В p(1), p(2) - закодирована часть речи словоформы, p(3) - тип и класс предлога или

одушевленности (существительного, полного прилагательного); p(4) -глагол 1-3 лица соответственно; p(5) - код значения числа (единственное число, множественное число); p(6) - код падежа (именительный, родительный, дательный); p(7) - код категории пассивности-активности; p(8) - код времени (настоящее, прошедшее, будущее); p(9) - код категории вида (совершенный, несовершенный) окончания [2].

Для формирования одной МИ-строки ко всей словоформе сравнивают код основы и код окончания на соответствие их первых пяти разрядов, если сходимости нет, то данные несовместимые. Для сравнения выбирается очередной код окончания. Если соответствие установлено, то остальные разряды результирующего кода формируются по правилам 10-значной дизъюнкции значений соответствующих разрядов кодов основы и окончания. При этом предварительно проверяется условие сходимости операндов или равенство одного из них нулю.

Аппаратные построения. Относительно аппаратной реализации алгоритма МА ближайшим подходом является применение неоднородных систем типа "базовая машина + функционально ориентированный процессор" (БМ+ФОП). Пример такой системы представлен на рисунке.



Структура функционально-ориентированной системы
морфологического анализа текстовой информации

Заключение

Таким образом, основными преимуществами применения предложенного аппаратного решения задачи морфологического анализа является повышение меры регулярности структуры ФОП, сокращение сроков разработки системы, возможность использования работников низшей квалификации во время проектирования, снижение трудоемкости во время конструирования, гибкость архитектуры, упрощение ремонта и замены модулей, которые вышли из строя, удобство использования сокращенной эксплуатационной документации, уменьшение числа типов кристаллов и повышение регулярности внутренней структуры БИС и СБИС, повышение быстродействия за счет предельного параллелизма в работе структуры универсального элемента пространственного типа.

БИБЛИОГРАФИЯ

1. Бондаренко М.Ф., Коноплянко З.Д., Четвериков Г.Г. Основи теорії синтезу надшвидкодіючих структур мовних систем штучного інтелекту: Монографія. - К.:ІЗМН, 1997.-264с.
2. Бондаренко М.Ф., Осыка А.Ф. Автоматическая обработка информации на естественном языке.-К.:УМКВО,199І.-142с.
3. Функционально-ориентированные процессоры /А.И. Водяхо, В.Б. Смолов, В.У. Плюснин и др.; Под ред. В.Б. Смолова. - Л.: Машиностроение, 1988.-224 с.
4. Пат. 20462 Україна, МКВ НОЗК 19/08. Двовходовий багатозначний логічний елемент / М.Ф. Бондаренко, З.Д. Коноплянко, Г.Г. Четвериков. - 4 с. .іл.; Опубл. 15.07.97, Бюл.3.
5. Четвериков Г.Г. Формалізація принципів побудови універсальних k-значних структур мовних систем штучного інтелекту// Доповіді НАН України.-2001.-№1(41).-С.76-79.

Сведения об авторах

М.Ф.Бондаренко: д-р техн. наук; ХНУРЭ; e-mail: kav@kture.kharkov.ua
А.В.Карпухин: канд. техн. наук; ХНУРЭ; e-mail: kav@kture.kharkov.ua
Г.Г.Четвериков: канд. техн. наук; ХНУРЭ; e-mail: chetvergg@kture.kharkov.ua
ХНУРЭ - 14, пр. Ленина, Харьков-61166, Украина

INTELLIGENT SYSTEMS MEMORY STRUCTURING

V. Gladun

***Abstract**: The requirements for the memory structuring of intelligent systems are discussed. Simultaneously with the introduction of information into memory there should take place the processes of association links (bonds) formation, hierarchy systematizing, classification and concept formation. The growing pyramidal networks (GPN) meet these requirements. Many years of experience of GPN application for data analyses in chemistry and material studies proves their sufficiently high potential.*

***Keywords**: intelligent systems, growing pyramidal networks, data analysis.*

1. Perception, presentation and analysis of information in intelligent systems

1.1.Competition of a computer with a man in resolving of intelligent problems more and more often ends up in the victory of a computer. But there arises the evident contradiction: computer genius victoriously solves the most complicated multivariate problems in artificial, relatively poor media (chess, for example) and “stumbles” over solving simple (for a man) life problems, requiring still quick understanding and assessment of multi-component situation.

What compensates the evident advantage of a computer over a man in quickness? The answer is unanimous that the reason lies mainly in memory structure. We will try to identify the structural peculiarities of memory, which, to our mind, are necessary for formation of clear explanations of a human phenomenon of information processing.

1.2.The peculiarity of intelligent systems, which causes no doubts, is the ability to analyze the perceived information. The process of analysis consists in picking up the integral parts and characteristic attributes of the analyzed whole. The product of analysis is organized (for example, with the help of logic links) sum of object attributes.

Thus, irrespective of the type of the information perceived (continuous or discrete) at a certain stage of analysis there appears a discrete representation of objects in the form of arranged aggregate of information blocks – attributes. The attributes serve as building blocks for the following analytical processes, leading in the long run, towards formation of generalized information models of the objects perceived. So, discretization of the perceived information, which consists in demonstrating attributes of objects, is an important peculiarity of intelligent systems.

1.3.The prevailing tendency in developing of intelligent systems is the improvement of man – machine interaction until the achievement of partner level of man-machine relations. That is why it is important to use natural, pertaining to a man principles of problems, situations and media modeling in computers. Partner model types (a man and a computer) should be similar. In life activity of a man a very important role is played by logic - linguistic information models, i.e. such models where the main elements are not numbers and calculations but names and logical bonds. Logic - linguistic models are adequately described with natural language constructions, and it is one of their decisive merits for designing of a man – machine interface. In computers to come there should be created conditions for man – machine solving of problems in partner mode providing switching over from a computer to a man and vice versa within the process of solving of problem. Such mode could be set up only by means of adjustment of information model types, used by partners. Logic – linguistic models are the most acceptable model types for such an adjustment.

1.4.Formation of memory structure is done simultaneously with perception of information and under the impact of the information perceived and already stocked. The memory structure reflects the information perceived. Information structuring is an indispensable function of memory.

The main processes of structuring include formation of associative links by means of identifying the intersections of attributive representations of objects, hierarchic regulation, classification, forming up generalized logical attributive models of classes, i.e. concepts.

Under real conditions of information perception there is often no possibility to get whole information about an object at once (for example, because of faulty foreshortening or lighting during the reception of visual information). That is why the processes of memory formation should allow for the possibility of “portioned” construction of objects models and class models by parts.

1.5.In different processes of information processing objects are represented by one of the two means: by a name (convergent representation) or by a set of meanings of attributes (displayed representation). The structure of memory should provide convenient transition from one representation to another. Mechanisms, providing such transition in neuro system of a man at recognition or recollection are considered in the works of S.G. Voronkov and Z.L.Rabinovich [1].

Let us sum up the above mentioned theses in the form of requirements to memory structuring in intelligent systems.

- In intelligent systems knowledge of different types should be united into net-like structure, designed according to principles common for all types of knowledge.
- The network should reflect hierarchic character of real media and in this connection should be convenient for representation of gender-type bonds and structures of composite objects.
- Obligatory functions of the memory should be formation of association bonds by revealing intersections of attributive object representations, hierarchic structuring, classification, concept formation.
- Within the network there should be provided a two-way transition between convergent and displayed presentations of objects.

2. Growing Pyramidal Network

The above mentioned requirements are met by growing pyramidal network. The theory and practical application of growing pyramidal networks are represented in many publications [2-5]. In this paper we present somewhat changed rules of formation of growing pyramidal network, ensuring their construction at the introduction of object attributive descriptions by parts. The example of a growing pyramidal network is presented in Fig.1.

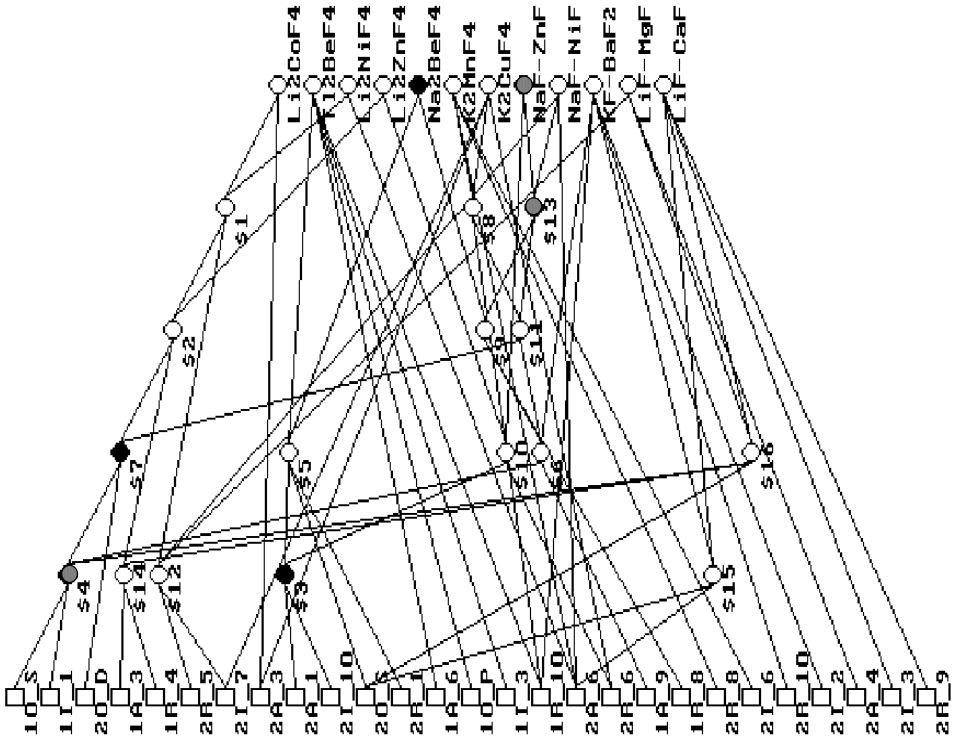


Fig. 1.

Table

Object	Class	1O	1A	1R	1I	2O	2A	2R	2I
Li2CoF4	A	S	3	4	1	D	3	5	7
Ti2BeF4	A	P	6	10	3	S	1	1	10
Li2NiF4	A	S	3	4	1	D	6	5	7
Li2ZnF4	A	S	3	4	1	D	1	6	10
Na2BeF4	A	S	9	8	1	S	1	1	10
K2MnF4	A	S	9	10	1	D	6	8	6
K2CuF4	A	S	9	10	1	D	3	6	7
NaF-ZnF2	B	S	9	8	1	D	1	6	10
NaF-NiF2	B	S	9	8	1	D	6	5	7
KF-BaF2	B	S	9	10	1	S	6	10	2
LiF-MgF2	B	S	3	4	1	S	4	5	7
LiF-CaF2	B	S	3	4	1	S	6	9	3

The pyramidal network is called a cycled oriented graph, where there are no vertices having one incoming arch. The vertices that have no incoming arch are called *receptors,* other vertices are called *conceptors*. The subgraph of the pyramidal network which includes *a* vertex and all vertices from which there are paths to *a* vertex, is called *pyramid* of *a* vertex. The vertices pertaining to *a* vertex pyramid form up its *subset*. The set of vertices towards which there are paths from *a* vertex is called *superset*.

In a subset and a superset of a vertex there are 0-subset and 0-superset that consist of those vertices that are immediately connected with it. While constructing a network the sets of meanings describing some objects (materials, aggregate states, situations, illnesses etc.) serve as incoming information. Receptors correspond to the meanings of attributes. In different problems they could be names of properties, relationships, states, actions, objects and objects classes. Conceptors correspond to the descriptions of objects as a whole and intersections of descriptions. The network shown in the Fig.1 is built on the basis of the Table where the objects are pairs of chemical elements, forming and not forming compounds.

In the Table the object descriptions are given, where 1O, 1A, 1R, 1I are the names of attributes, describing the first element of the compound; 2O, 2A, 2R, 2I are the names of attributes, describing the second element of the compound, and the letters and figures in cells are the meanings of the corresponding attributes.

In the initial state the network consists only of receptors. Conceptors are formed as a result of the work of algorithm of network construction. The algorithm described in a number of publications [2-5] is meant for the work in situations, where the attributive description of each object is fully known and is introduced as a whole. With appearing of new attributes, which characterize the object, it is necessary to form a new complete description of the object and to replace the pyramid that represents it with another one, which corresponds to the new description. But as it was mentioned in real situations of functioning of an intelligent agent simultaneous perception of all characteristics of an object is far from possible. In such cases the information about objects comes in parts. Then there arises the necessity to change a bit the algorithm of constructing a network to provide the possibility to include into the existing object pyramids new attributes according to their appearance without replacement of pyramids as a whole. Now we will present the description of the changed algorithm.

At the introduction of an attributive description of an object receptors corresponding to the meanings of attributes coming into the description are transferred into the state of *excitation*.

The excitation is propagated through the network. The conceptor is switched to the state of excitation if all vertices of its 0-subset are excited. Receptors and conceptors preserve the state of excitation within the period of performing all the operations of constructing of the network.

Let at the introduction of the description of some object F_a be the subset of the excited vertices of 0-subset of ***a***-vertex; *G* is the set of the excited vertices of the network having no other excited vertices in their supersets.

Introduction of new vertices and arcs is done according to the following rules.

Rule 1.

If *a* vertex is not excited and F_a set contains more than one element, then arcs, connecting vertices from F_a set with *a* vertex are annulled and a new conceptor is introduced, which is connected by incoming arcs with vertices of F_a set and by an outcoming arc with *a* vertex. The new vertex is in the state of excitation.

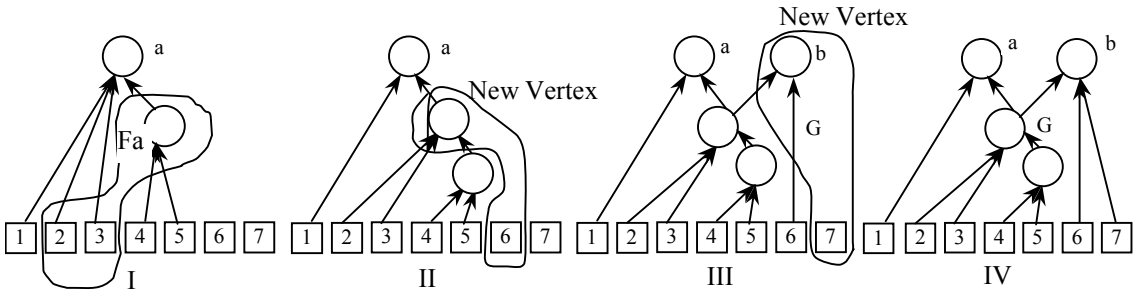


Fig. 2.

Fulfillment of the rule 1 is illustrated by Fig.2(I, II). The network II appears after excitation in the network I of 2,3,4,5 receptors. As it follows from the rule 1, the condition of introduction of a new vertex into the network is the situation when a certain vertex is not completely excited (not all vertices but not less than two of its 0-subset are excited). New vertices are introduced into 0-subsets of not completely excited vertices.

After introduction of new vertices into all areas where the condition 1 is met, the rules 2 or 3 are applied.

Rule 2.

If *G* set contains more than one element and does not include the vertex marked with the name of the object introduced, a new conceptor joins the network and is connected by incoming arcs with all vertices of *G* set. The new vertex is in the state of excitation.

Fulfillment of the rule 2 is illustrated in Fig.2 (II, III). The network III appears after excitation of receptors 2, 3, 4, 5, 6 in the network II.

Rule 3.

If *G* set contains the vertex, marked with the name of the object introduced, this vertex is connected by incoming arcs with other vertices of *G* set.

Fulfillment of the rule 3 is illustrated in Fig.2 (III, IV). The network IV appears after excitation of receptors 2, 3, 4, 5, 6, 7 in the network III under condition that this set of receptors corresponds to the description of *b* object.

In the changed algorithm the possibility of introduction of new attributes into the existing pyramids is provided by the rule 3.

Pyramidal networks are convenient for performing different operations of associative search. For example, one could choose all objects, containing a given combination of attribute meanings, following the paths coming out the vertex, which corresponds to this combination. For the access of all objects the descriptions of which intercross with the description of the given object, it is enough to trace the paths coming out of the vertices which form up its pyramid. All processes, connected with construction of the network at the processing of one description are localized in a relatively small part of the network, i.e. in the pyramid, corresponding to this description.

Hierarchical structure of the networks, which allows them to reflect the structure of composing objects and gender-species bonds naturally is an important property of pyramidal networks.

Conceptors of the network correspond to the combinations of attribute meanings, defining conjunctive classes of objects. With insertion of excited vertices into the pyramid of the object there takes place linking of the object with the classes, the definitions of which are represented by these vertices. Thus, while building a network, there form up conjunctive classes of objects, i.e. classification without a teacher takes place. Classifying properties of a pyramidal network are very important for automation of media and situation modeling.

The transfer from convergent representations of objects (conceptors) to displayed ones (sets of receptors) is fulfilled by a survey of pyramids in different directions.

In growing pyramidal networks there are realized the processes of forming generalized logical models of object classes, i.e. concepts.

The formed up concept of any complexity is represented in the network by an ensemble of specially picked out vertices. On the basis of network analysis a special procedure builds up a concept in the form of a logical expression.

Logical expressions defining classes of objects are united in Cluster Data Base (CDB). CDB contained information about object groups (clusters) that are specific for the domain under investigation. CDB are used for classification, diagnostics and prognostication.

When the concept for a certain class is formed, the problems of forecasting and diagnostics are reduced to the problem of classification. Classification of new objects is done by comparing their attributive descriptions with the concept, determining the class of objects to be forecasted or diagnosed. The objects could be classified calculating the meaning of logical expressions, representing the corresponding concepts.

In a pyramidal network the information is stocked by its reflection in the network structure. Information about objects and object classes is represented by ensembles of (pyramid) vertices, distributed along the whole network. Introduction of new information causes redistribution of bonds between the vertices of the network, i.e. change of its structure.

Of course, the benefits of pyramid networks are fully demonstrated with their physical realization, which allows parallel spreading of signals through the network.

There is the analogy between the main processes taking place in growing pyramidal networks and neuron networks. The decisive advantage of a growing pyramidal network is the fact that its structure is formed completely automatically depending on the introduced data. As a result there is achieved optimization of information presentation due to adaptation of the network structure to the structural peculiarities of the data. Unlike neuron networks the adaptation effect is achieved without introduction of a priori excess of the network. The learning process does not depend on predetermined network configuration. The drawback of neuron networks if compared to the growing pyramidal network is also the fact that generalized knowledge in them cannot be represented in the form of rules or logical expressions. It makes their interpretation and understanding by a man difficult.

The program system CONFOR (CONcept FORmation) that implements methods of data analysis on the basis of growing pyramid networks, has been tested by time. The typical applied problems, for solving of which this system was used are: forecasting new chemical compounds and materials with the indicated properties,

forecasting in genetics, geology, medical and technical diagnostics, forecasting malfunction of complex machines and sun activity.

3.Summary

Pyramidal network is a network memory, automatically tuned into the structure of incoming information. Unlike the neuron networks, the adaptation effect is attained without introduction of a priori network excess.

The research done on complex data of great scope showed high effectiveness of application of growing pyramidal networks for solving analytical problems. Such qualities as simplicity of change introduction, combining processes of information introduction with processes of classification and generalization, high associativity makes growing pyramid networks an important component of forecasting and diagnosing systems.

Bibliography

1. Voronkov G.S. Rabinovich Z.L. Natural media of memory and thinking: model // Intern.conf.: Knowledge-Dialogue-Solution-2001.-St.-Pb.-2001(in Russian).
2. Gladun V.P. Partnership with computers. Man-computer task-oriented systems.-Kiev: "Port-Royal",2000(in Russian).
3. Gladun V.P. Solution planning.-Kiev: Naukova dumka, 1987(in Russian).
4. Gladun V.P. Processes of new knowledge formations.-Sofia: "Pedagog 6",1994(in Russian).
5. Gladun V. Vaschenko N. Analytical processes in pyramidal networks. // International Journal: Information Theories and Applications.-2000, V.7, №3.

Author information

Victor Gladun – V. M. Glushkov Institute of Cybernetics, National Academy of Science of Ukraine, Prospect akad. Glushkova 40, 03680, Kiev, Ukraine; e-mail: glad@aduis.kiev.ua

(The work has been done within the framework of the project INTAS #00-397)

OPTIMIZATION OF GABOR WAVELET FOR FACE RECOGNITION

K.Murygin

Abstract: *The article describes researches of a method of person recognition by face image based on Gabor wavelets. Scales of Gabor functions are determined at which the maximal percent of recognition for search of a person in a database and minimal percent of mistakes due to false alarm errors when solving an access control task is achieved. The carried out researches have shown a possibility of improvement of recognition system work parameters in the specified two modes when the volume of used data is reduced.*

Key words: *person recognition, gabor wavelets.*

Introduction

Now the methods based on gabor wavelets, draw more and more attention of the researchers engaged in image recognition, including face recognition. One of explanations of growing popularity of the given approach are results of biologists researches, shown similarity of two-dimensional Gabor kernels with the form of receptor field of visual cells in the primary visual cortex [1,2]. Besides the great positive experience of Gabor filters use in tasks connected with person recognition by face image has been already saved up [3,4,5,6,7,8,9].

Gabor functions [6,8,9] are localized in spatial and frequency area and look like a plane wave with a wave vector $\vec{k}$, restricted by a Gaussian envelope function with width σ / k, where $\sigma = 2\pi$:

$$\psi_j(\vec{x}) = \frac{k_j^2}{\sigma^2} \exp\left(-\frac{k_j^2 \vec{x}^2}{2\sigma^2}\right)\left[\exp\left(i\vec{k}_j\vec{x}\right) - \exp\left(-\frac{\sigma^2}{2}\right)\right] \quad (1)$$

The normalizing factor, the second exponent in square brackets, is received from a condition of equality to integral zero:

$$\int \psi_j(\vec{x}) d^2\vec{x} = 0 \quad (2)$$

The following wavelet transformation gives complex factors which then are used as elements of feature vectors, describing the initial image $I(\vec{x})$ in a point $\vec{x}$:

$$J_j(\vec{x}) = \int I(\vec{x}')\psi_j(\vec{x} - \vec{x}')d\vec{x}' \quad (3)$$

Complex factors $J_j(\vec{x})$ can be written down as $J_j(\vec{x}) = a_j(\vec{x})\exp\left(i\varphi_j(\vec{x})\right)$ where $a_j(\vec{x})$ - is slowly varying amplitude, and the phase $\varphi_j(\vec{x})$ changes with characteristic frequency of corresponding Gabor kernel (1). Factors $J_j(\vec{x})$ make sense, similar to factors of Fourier transformation. However, as functions (1) are localized in spatial area, $J_j(\vec{x})$ characterize not the image $I(\vec{x})$ completely, but its some part, which size is defined by parameter $\frac{k_j^2}{\sigma^2}$, and position – by argument $\vec{x}$.

Vectors of features J_j and J'_j received on the basis of expression (3) are convenient for comparing with the use of a similarity measure as a corner cosine between them:

$$S(J_j, J'_j) = \frac{\sum_j J_j J'_j}{\sqrt{\sum_j J_j^2 \sum_j J'^2_j}} \quad (4)$$

The combination of metrics (4) and condition (2) allows to exclude influence of any linear transformations of initial images on result of comparison. Thus, influence of brightness and contrast of initial images on result of comparison is minimized.

Each person entered into a database, is represented as set of local feature vectors received in beforehand-defined points of a face. The chosen set of such points in aggregate with the feature vectors received in them refers to a face graph (see fig. 1).

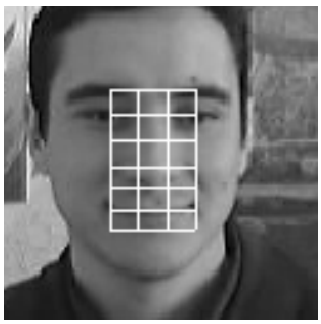


Figure 1 – Arrangement of face graph points

Automatic recognition with use of face image graphs is divided into two stages. At the first stage graph positioning on the image is carried out. On the second – comparison of the found graph with the graphs of persons kept in a database is carried out. Graph comparison is made by summation of measures of matching values of corresponding feature vectors received according to the formula (4), on all nodes of the graph.

At the chosen configuration the face graph (an arrangement of graph points) at both stages of recognition automatic system work, a question on a choice of a set of scales of used Gabor wavelets, which provide peak efficiency of algorithms work raises.

In works [6,8,9] for person recognition by face image Gabor functions of five various scales $\nu = \{0,...,4\}$, and eight orientations $\mu = \{0,...,7\}$ were used. Each function was determined by the following characteristic wave vector:

$$\vec{k}_j = \begin{pmatrix} k_\nu \cos\phi_\mu \\ k_\nu \sin\phi_\mu \end{pmatrix}, \quad k_\nu = 2^{-\frac{\nu+2}{2}}\pi, \quad \phi_\mu = \mu\frac{\pi}{8}, \tag{5}$$

where index $j = \mu + 8\nu$. Thus, full wavelet transformation (3) gives 40 complex factors in each image point (5 scales and 8 orientations). The used set of scales of Gabor functions were offered by authors of work [7] who have carried out researches for two values of the spatial factor f determining distance between the next scales of Gabor kernels, and various values of parameter k_{max} determining the maximal scale of used kernels. Researches have shown, that optimum values are $f = \sqrt{2}$ and $k_{max} = \pi/2$. Thus, the used range of spatial frequencies is wide enough (T = 4 – 16 pixels). In work [10] the analysis of used scales of Gabor functions for reduction of number of used kernels has been carried out at preservation of an overall performance in application to a task of localization of person face graph points. Optimum results are received at use of one scale and eight orientations. And the spatial frequencies corresponding to the given scale are small enough (T = 16 – 25 points, depending on the chosen point of face graph and at radius of an eye pupil of 4 pixels).

In the present work, researches similar with described in [10] were carried out with the purpose of reception of an optimum set of Gabor functions for the decision of a recognition task. At carrying out of experiments were used some combinations of scales suggested in work [7] and used in works [6,8,9] described above.

Carried out researches

For researches the database of 160 face images of 10 persons (database of Weizmann Institute of Science∗) was used. All photos have been received at various illuminations. For each of faces key points (the centers of pupils, the centers of eyebrows above pupils, a tip of a nose and the edge of lips) were manually marked in accordance with which face graph was built automatically, and face properties in each graph point as feature vectors received on a basis of (3) and (5) were remembered. Graph configuration described in works [6,8,9] (see fig. 1) was used. The sizes of all face images have been preliminary resulted in one scale. The interpupil distance has been chosen for a scale factor. The reference scale was equal 34 pixels.

Researches were carried out in two directions. First, dependence of efficiency of search in a database on scales of used Gabor kernels was investigated. The given task is characterized by absence of necessity to analyze mistakes connected to face admission as the standard most similar to the input image is searched in a database. Recognition is considered correct if the found image, most similar to the input face image belongs to the same person. Thus, optimization criterion is the percent of correct recognition. As reference accidentally one of 16 images for each person was chosen. In total 16 experiments were carried out. The estimation of efficiency was calculated as the average efficiency received in all experiments. For each of ranges $\nu = \{0,...,4\}$ the results given in table 1 have been received.

The received data testify that the best results for search in a database give scales $\nu = \{0,1,2\}$, and it's better to use all three scales that give the maximal percent of recognition – 87.5%. At use of all five scales efficiency of recognition falls up to 81.4% that's possible to explain by strong dependence of the big scales on conditions of illumination. Use of only 0 and 1 scales reduces efficiency (in comparison with maximal) less than on 1 % at reduction of volume of remembered and analyzed data more than on 30 %. It allows to use only these scales in systems, critical to volume of the remembered data and an operating time of recognition algorithm.

∗ the database is received from a server ftp://eris.wisdom.weizmann.ac.il/pub

Table 1 – Dependence of percent of correct recognition on used scale ranges

Used Gabor functions scales					Recognition
0	1	2	3	4	percent, %
*					74.8
	*				80.8
		*			75.7
			*		64.6
				*	62.0
*	*				86.7
	*	*			84.9
		*	*		76.1
			*	*	70.5
*	*	*			87.5
	*	*	*		82.0
		*	*	*	76.6
*	*	*	*		82.8
	*	*	*	*	80.9
*	*	*	*	*	81.4

The second direction of researches has been connected to access control task. For an estimation of recognition efficiency for each of researched scales sets, functions of density of distribution of graph comparison sizes (see fig. 2) have been calculated. In figure 2, more to the left, function of distribution density, received at comparison of different people faces is shown. More to the right received for face images of one person. The square of crossing area, characterizes the minimal total mistake of recognition, which is achieved at a choice of a threshold of recognition equal abscissa of crossing points of distribution density functions. The data received at a similar choice of recognition threshold, are resulted in table 2.

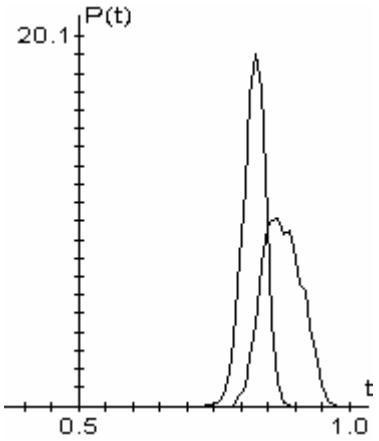


Figure 2 – Distribution density functions of graph comparison results

The analysis of received data, has shown, that the best results by a total mistake are observed at use of the same scales, as in experiments on search in a database. However in a case of access control task the mistakes connected to wrong recognition (it is equivalent to an opportunity of the non-authorized access) have the great importance. Therefore optimum scales are necessary for choosing, mainly being based on the data received for these mistakes.

Table 2 – Dependence of mistakes of 1 and 2 sorts on used Gabor functions scales

Used Gabor functions scales					Face admission	Wrong recognition	Total mistake
0	1	2	3	4			
*					30.83	8.03	38.86
	*				22.67	14.85	37.52
		*			24.00	20.07	44.07
			*		44.75	7.22	51.97
				*	51.25	6.26	57.51
*	*				27.50	6.40	33.90
	*	*			29.08	9.41	38.49
		*	*		32.83	13.20	46.03
			*	*	43.00	7.67	50.67
*	*	*			19.33	14.24	33.57
	*	*	*		27.08	13.62	40.70
		*	*	*	35.83	11.48	47.32
*	*	*	*		25.67	11.19	36.86
	*	*	*	*	29.58	13.72	43.30
*	*	*	*	*	26.33	14.46	40.79

If the percent of allowable mistakes of recognition is rigidly determined, parameter of optimization becomes frequency of recognition, which is determined by mistakes connected to face admission. In table 3 the experimental data received at aprioristic setting of an allowable mistake of recognition are given.

As the results given in table 3 show, use of only 0 and 1 scales is also optimum for a task of access control.

Table 3 – Dependence of mistakes of face admission on used Gabor functions scales and restrictions for a recognition mistake

The used Gabor functions scales					Face admission (%) at an allowable recognition mistake:				
0	1	2	3	4	0 %	1 %	2 %	3 %	4 %
*					75	54	43	43	43
	*				78	53	44	44	44
		*			77	57	57	49	49
			*		79	59	52	52	52
				*	79	63	57	57	57
*	*				69	38	38	38	28
	*	*			72	56	48	48	39
		*	*		69	55	47	47	47
			*	*	75	56	56	50	50
*	*	*			76	50	40	40	29
	*	*	*		67	52	45	45	45
		*	*	*	69	56	51	51	44
*	*	*	*		61	44	44	36	36
	*	*	*	*	67	53	46	46	46
*	*	*	*	*	67	52	45	45	36

Conclusion

The carried out experiments have shown, that reduction of a number of Gabor functions scales from 5 up to 2 (use of only 0 and 1 scales) improves parameters of recognition system work as for task of search in a database, so for access control task. Thus the volume of the remembered and analyzed data is essentially reduced, speed of work of algorithm of face comparison raises.

References

1. C. PJ. Jones, L. Palmer An evaluation of the two-dimensional Gabor-filter model of simple receptive fields in cat striate cortex. // J. Neurophysiol., 1987.-p. 1233-1258.
2. D. Burr, M. Morrone, D. Spinelli Evidence for edge and bar detectors in human vision. // Vision Res., 1989,-p. 419-431.
3. C. Padgett and G. Cottrell. Representing face images for emotion classification. In M. Mozer, M. Jordan, and T. Petsche, editors, Advances in Neural Information Processing Systems, volume 9, Cambridge, MA, 1997. MIT Press.
4. M.S. Bartlett. Face Image Analysis by Unsupervised Learning and Redundancy Reduction. PhD thesis, University of California, San Diego, 1998.
5. V. Bruce. Human face perception and identification. In H. Wechsler, P.J. Phillips, V. Bruce, F. Fogelman-Soulie, and T. Huang, editors, Face Recognition: From Theory to Application, NATO ASI Series F. Springer-Verlag, in press.
6. Wiskott L., Fellous J.M., Krueger N. and von der Malsburg C. Face Recognition and Gender Determination. // Proc. of the Int. Workshop on Automatic Face-and Gesture-Recognition, Zuerich, 1995.-p. 92-97.
7. M. Lades et al. Distortion invariant object recognition in the dynamic link architecture. IEEE Trans. Comput., 42 (3):-p. 300-311,1993.
8. Wiskott, L. Phantom Faces for Face Analysis. // Proc. 7th Intern. Conf. on Computer Analysis of Images and Patterns, CAIP ' 97, Kiel, 1997.-p. 480-487.
9. Wiskott L., Fellous J.M., Krueger N. and von der Malsburg C. Face Recognition by Elastic Bunch Graph Matching. // Proc. 7th Intern. Conf. on Computer Analysis of Images and Patterns, CAIP ' 97, Kiel, 1997.-p. 456-463.
10. 10.Ian R. Fasel, Marian S. Bartlett, Javier R. Movellan, A Comparison of Gabor Filter Methods for Automatic Detection of Facial Landmarks, Proceedings of the 7th Symposium on Neural Computation, 2000,-p. 44-50.

Authors information

Murygin Kirill Valerievich – Institute of Artificial Intelligence, B.Hmelnitsky avenue, 84, Donetsk - 83050, Ukraine; e-mail: kir@iai.donetsk.ua

A PROPOSED STRUCTURE OF KNOWLEDGE BASED HYBRID INTELLIGENT SYSTEMS FOR SOPHISTICATED ENVIRNOMENTS

Agris Nikitenko

***Abstract**: The paper deals with a problem of intelligent system's design for complex environments. There is discussed a possibility to integrate several technologies into one basic structure. One possible structure is proposed in order to form a basis for intelligent system that would be able to operate in complex environments. The basic elements of the proposed structure have found their implemented in software system. This software system is shortly presented in the paper. The most important results of experiments are outlined and discussed at the end of the paper. Some possible directions of further research are sketched.*

***Keywords**: Artificial intelligence, knowledge based intelligent systems, hybrid intelligent systems, autonomous intelligent systems, inductive reasoning, deductive reasoning, case based reasoning, associative reasoning.*

Introduction

The Artificial intelligence is one of the youngest branches of the modern science.

During a short period of time (lasting only several decades) there have been developed a lot of different technologies and approaches to solve various types of problems existing in the field of artificial intelligence. A complexity of those tasks that can be performed by intelligent systems is growing from year to year. In this paper I would like to keep a closer watch on those intelligent systems that would be able to operate autonomously in complex environments that are close to real world. Obviously, if an intelligent system operates autonomously in a complex environment it needs some kind of environment's model. In spite of model's less complexity it is still quite sophisticated. Though, the basic question is, what kind of components are necessary for such an intelligent system in order to maintain and use such a complex environment's model.

Before trying to built a structure of an intelligent system, it is necessary to define the environment in which the system will operate. The basis of such a definition can be found in the assumption that every object (in this case the environment of intelligent system) can be described as a system [Lit.1.] Obviously, a complex environment can be described as a complex system. There are several features that defines a complex system [Lit. 2.]:

- uniqueness – usually complex systems are unique or number of similar systems is unweighted.
- hardly predictable – complex systems are very hard to predict. It means that it is hard to calculate the next state of a complex system if the previous states are known.
- an ability to maintain some progress resisting against some outer influence (including influence of the intelligent system).

Of course, any complex system has every general feature of a system such as a set of elements, a set of relations etc [Lit. 3.]

To built a complete model of an environment (complex system) that corresponds to the listed features is either impossible or very expensive. In this case the intelligent system will have incomplete model of environment or will not have it at all. In complex environments usually it is impossible to describe the environment completely. This is caused by a huge space of possible states of environment (or even infinite), expanses or other reasons. It means that an intelligent system in the great part of cases will be using only an incomplete model of environment during its existence.

A very promising way to deal with a complexity and an incompleteness is to use some kind of learning mechanisms in order to adjust the intelligent system to new conditions.

A closer watch even on the early methods used in the system theory shows that an analysis of a complex system requires three basic types of reasoning: Deductive, Inductive and Associative [Lit.1,2,3.].

It means that an intelligent system also needs to be able to use all of those reasoning techniques in order to be effective enough.

Basic Features of an Intelligent System

In this section the basic features are outlined and explained according to the previous research activities [Lit.4.].

Summarizing the basic features are:

- an ability to generate a new knowledge from the already existing. This ability can be achieved by means of deductive reasoning. In order to increase an effectiveness a case based reasoning can be used [Lit.5.] Under this feature lies an ability to reason logically.
- an ability to learn. This feature can be implemented by means of inductive reasoning. During operation the intelligent system can collect a set of facts through sensing an environment which forms an input for learning.
- an ability to reason associatively. This feature is necessary due to a huge set of possible different situations that the intelligent system can face with. For example, there may be two different situations which can be described by n parameters (n is big enough number) where only k parameters are different (k is small enough number). It is obvious that it is possible to reason about these situations as about similar situations.
- an ability to sense environment. This feature is absolutely necessary for any intelligent system that is built to be more or less autonomous.

- an ability to act. This feature also is necessary for any intelligent system that is designed to do something. If the system (autonomous) is unable to act, it wont be able to achieve its goals.

These features form the basis for an intelligent system that operates in sophisticated environments. According to the features of complex systems that are listed before, any of them can be implemented as it is needed for particular task. The question is how to bind all listed features in one intelligent system.

Obviously, there is a necessity for some kind of integration. There are many good examples of different kinds of integration. For example so called soft computing which combines fuzzy logic with artificial neuron nets [Lit.6.] or Case based reasoning combined with deductive reasoning [Lit.7.].

In order to adjust an intelligent system for some particular tasks different structures can be used [Lit 14].

In this paper I would like to present one of the alternative structures that could be used in order to implement all of the listed features and can form a kernel of an autonomous intelligent system.

Structure of the Intelligent System

According to the list of very basic features there can be outlined the basic modules that correspond to the related reasoning techniques:

As it is shown in figure 1, there are four basic modules.. As it is said before in complex environments there may be a while of unique situations. To extract (or to learn) any rule an intelligent system needs at lest two equal (or similar – the most part of feature (attributes) are equal) situations. It means that in complex environments a while of situations experienced by an intelligent system may remain unused. Obviously, these unique situations (or cases) may be extremely valuable not only for the intelligent system but also for the researcher that uses the system like it is in medicine. The case based reasoning module is involved to process and use these unique situations.

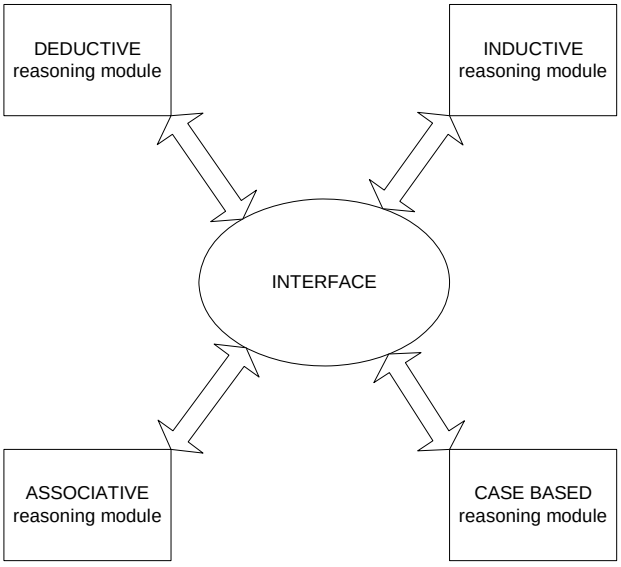


Figure 1. Basic modules

As it is depicted in figure 1, all of the four modules need some interface to communicate with each other.

Of course, the intelligent system needs additional modules that would supply it with a necessary information about the environment and mechanisms to perform some actions. During this research there is developed an alternative structure of the interface that allows combination of four mentioned reasoning techniques. This structure is depicted in the figure 2.

The structure consists of several elements. The fundamental element of whole structure is *object*.

Object. Objects are key elements in the interface structure. They correspond to some kind of entities in the environment (or in the intelligent system). Every object is described with a set of features (attributes). Each feature has some value. As it is depicted in figure 2. objects are linked to each other by associative links. These

links form basis for associative reasoning. Two objects are linked if there is any common feature between them. These links can be weighted. The greater weight becomes the more common features are between objects.

When an intelligent system runs into new situation some subset of objects is activated. These objects map to those entities that the intelligent system currently senses. If there is no rule that can be activated (see below), then the intelligent system may try to activate associated objects where links have some threshold weight. Thus the system can try to reason about objects by using associated rules. The result may be less feasible, but using association between objects the system can run out of the dead end situations. Associative links can spread the "activation" through the whole net of objects if the links among them are strong enough.

A mechanism of associative memory is very useful when the system works with noisy data. This mechanism allows to correct faults of the sensing mechanism [Lit.8.]. For example, if the input vector of the sense which corresponds to some entity has some uncertain or incorrect elements (attributes of object) then the system would not be able to activate any of the objects. In this case associative memory mechanism will activate the closest object [Lit.8.] thus the sensing error will not have significant effect on the reasoning process.

Rules. Rules are any kind of notation that represents causalities. In the practical experimentations were used a well known *if..then* notation. As it is depicted in the figure 2 rules are linked to objects and actions. Each rule may contain a reference to some objects (or its attributes). Therefore if the rule contains such a reference then it is linked to the object. These links help to reduce a searching space and forms a basis for associative reasoning. Each time when the intelligent system receives a new situation, there are activated those objects that are sensed by the system. Linked rules are also activated. The system searches for suitable rules only in the set of activated rules. When system activates objects by using associative links linked rules also are activated thus system can scan also a set of "associated" rules. This ability can significantly improve system's ability to adapt. As it is depicted in figure 2 rules are linked to actions. Rules (for example, those of type *If..Then*) may reference not only to facts but also to actions.

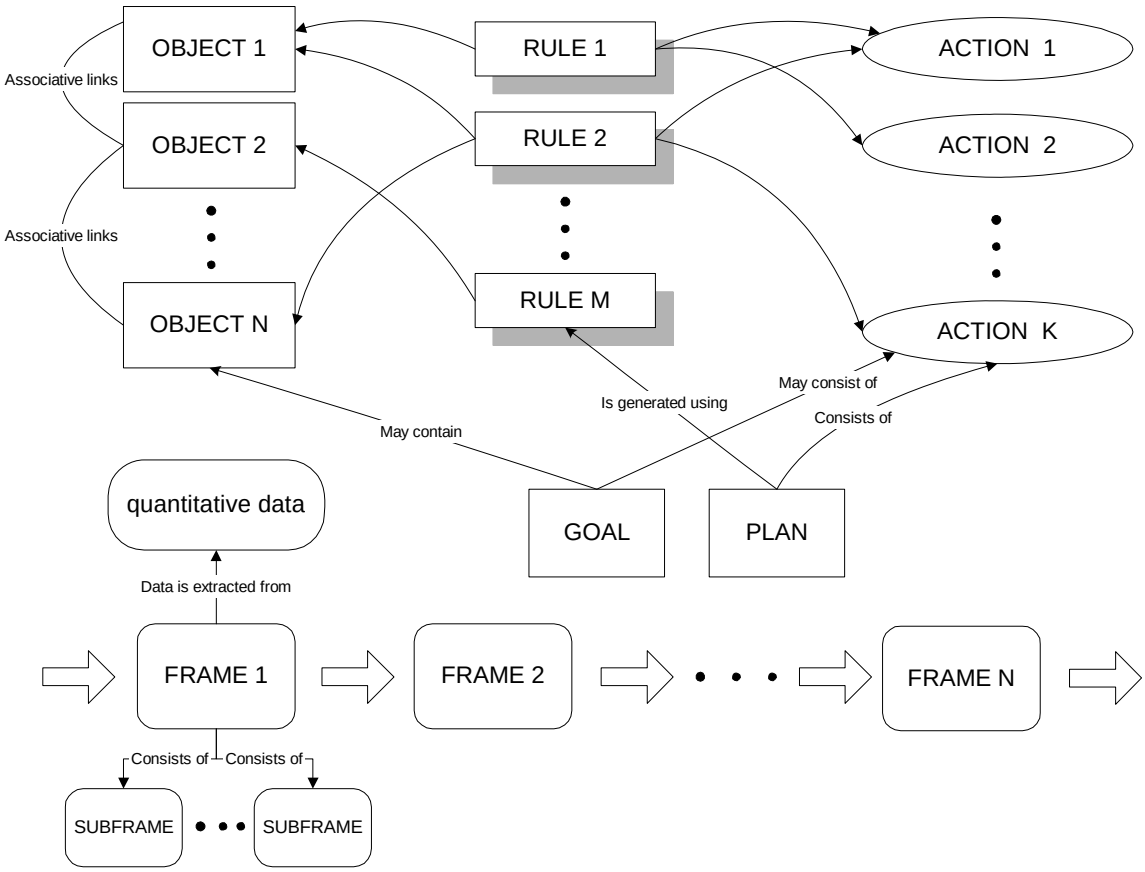


Figure 2. Structure of the interface.

Actions. Actions are some kind of symbolic representation that can be translated by the intelligent system and cause the system to do something. For example "*turn to the right*" causes the system to turn to the right by 90^0.

Actions may be structured in hierarchies. Thus a system can built high level actions that consist of a set of basic actions. For example an action "*open the door*" may consist of two lover level actions: "*unlock the door* " and "*push the door*". High level actions may be formed like scenarios. A scenario can consist of a sequence of lover level actions. It means that rules referencing to the high level actions do not need to reference to each of the basic actions.

The scenarios of actions can be formed as sequence of actions that drive the system to the goal. It means that actions of high level in some way are representation of the positive experience of the system.

Each action consists of three parts: precondition, body and postcondition. Precondition is every factor that should be true before the action is executed. For example, before opening the door it has to be unlocked. Body is a sequence of basic (or lover level) actions. Post conditions are factors that will be true after the execution of action. For example after opening the door, the door will be opened.

2 actions may be sequenced one after another if the following is true:

$$c \in C, C = c' \cup Z ,$$

where c ' – a set of postconditions of the first action
Z – a set of current conditions (known facts)
C – a set of precondition of the second action

Frames. Frames are some kind of data structures that contain the sense array from environment and from the system. It means that frames contain snapshots of the environment's and the system's states.

As it is depicted in figure 2 frames are chained one after another thus forming a historical sequence of the environment's and the system's states. Frames can be structured in hierarchies. Hierarchies help to see values of features that can not be seen in a single snapshot. For example motion trajectories of some object e.c.

Frames are the input for learning (induction module) algorithms. It is obvious that queue of frames can not be infinite due to bounds of hardware equipment. It means that there should be used some "forgetting" mechanism that determines the length of the frame sequence.

Goal. A goal is some kind of a task that has to be done by the system. It can be defined in three ways: as a sequence of actions that should be done, as some particular state that should be achieved or as a combination of actions and states.

Goals can also be structured in hierarchies that determine the priorities of different goals.

Plan. A plan is a sequence of actions that the system is trying to accomplish. It can be formed using both basic and complex actions. After the plan is accomplished it is evaluated depending on whether the goal is achieved or not.

Quantitative data. This element is used to maintain any kind of quantitative data that is needed for the system. For example it can contain certainties about facts or rules, possibilities e.c. A source of quantitative data is the chain of frames.

All of those components together form the interface for the basic modules: Inductive, Deductive, Case based and Associative reasoning.

Fundamental elements of the structure are implemented in experimental software.

Practical Implementation

As it is mentioned above, fundamental elements of the proposed structure are implemented into experimental software.

The implemented elements are: Case based reasoning, Inductive reasoning and Deductive reasoning. Deductive reasoning is implemented as a statement logic module based on rules designed in *if…then* manner. The induction module is implemented using well known algorithm ID3 [Lit.9.] It has its more effective successor C4.5

[Lit.10]. The case based reasoning module is implemented using pairs {situation, action}. Each of such pair has its value which determines how effective it is in particular case. During the planning this value determines which actions are selected if more then one action may be selected.

The environment is implemented as world of rabbits and wolf (domain of pray and hunter). There are defined additional objects “obstacles”. The number of rabbits and obstacles is not specified. The intelligent system is implemented as wolf. Rabbits may be moving or standing at one place. Wolf can catch rabbits. The wolf is moving according to its plan. User can freely change number and place of obstacles and rabbits. The goal can defined and changed at any time.

The intelligent system demonstrates flexibility of the proposed structure. The results of experiments and experience accumulated during the implementation shows that new types of objects can be introduced without changing the proposed structure. The goal can be changed freely.

It means that even being incomplete this structure demonstrates good ability to adapt and to operate

I believe that implementation of the whole structure can give very flexible system that would be able to operate in more complex environments.

Possible Advances and Future Research

Obviously there may be such tasks that can not be done using a single intelligent systems. In other words there may be a task that could be done only by a set of intelligent systems. For example, simulation of real world (battlefields, transport systems e.c.) It means that intelligent system has to be ready to negotiate with other intelligent systems that may be built using different kind of technology. It does not mean that always it is necessary to communicate.

This question is under discussion among researchers working in the field of multiagent systems.

There are different ways to design multiagent system [Lit.11., Lit 12]. In different domains there can be different solutions. Before designing a system for operating in a multiagent environment several questions should be answered.

Some of the basic questions are:

- Will the agents be able to communicate.
- How the agents will communicate to each other.
- What resources they will shared and what resources will stay unshared
- How the conflicts will be solved.
- Of what type the agents will be (competitive, cooperative e.c.)

Only after answering on those questions the intelligent system can be adjusted according to the collected answers.

Referencing to the said above, there may be outlined one of the directions of farther research and experiments – adjustments of the proposed structure in order to allow the intelligent system operate in a heterogeneous communicating multiagent environment. This is the most sophisticated type of multiagent systems [Lit.12.] and the most interesting form the point of view of research. The structure adjusted for such an environment should be able to operate in less complex environments.

One of the most sophisticated problem in such a multiagent environment is communication because every of communication parameter may be variable. It is easy to imagine that two intelligent systems may try to communicate using different knowledge representation schemas, different knowledge, different communication protocols, different type of “conversation” (for example: questioning, answering, argumentation e.c.) or even different physical communication channels (radio frequency, verbal communication e.c.). All said means that it is almost impossible to design an intelligent system that would be able to adjust its communication mechanism to all possible variations. It means that the system should be design for one or few of the possible communication standards. Which one to choose? May be some of the existing standards should be used [Lit.13.]. This question is to be answered only after a deeper analysis. This is the second possible direction of farther research.

In the field of practical research and experimentation the next step is to design an experimental intelligent system that would be an implementation of the whole proposed structure in order to carry out more complex experiments using different environments and different goals.

The other practical experiments that are of my special interest are experiments with some kind of robotic system in order to try the proposed structure in the field of motion control.

Conclusions

Practical experiments show that the proposed structure may be very flexible even in very changing environments with variable goals. It means that it is reasonable to carry out farther research and experiments in order to advance this structure.

Some problems are related with amount of processed data during the reasoning that significantly slows down the whole system.

It means that an optimization of the processes needs to be a part of future activities.

In spite of the quite promising results that are collected during the practical experiments there are still a lot of open questions that should be answered in farther research activities.

References

1. В.Н. Волкова «Теория систем и методы системного анализа вуправлении и связи» Радио и Связь 1983.
2. В.В. Дружинин, Д.С. Конторов «Системотехника» Радио и Связь 1985.
3. А.Д.Цвиркун «Структура сложных систем» Советское радио 1975.
4. A.Nikitenko, J.Grundspenkis “The kernel of hybrid intelligent system based on inductive, deductive and case based reasoning” KDS2001 conference proceedings, St. Petersburg 2001.
5. Bradley J. Rhodes “Margin Notes Building a Contextually Aware Associative Memory”, *The Proceedings of the International Conference on Intelligent User Interfaces* (IUI '00), New Orleans, LA, January 9-12, 2000.
6. M. Pickering “The Soft Option” 1999.A.R.Golding,
7. P.S.Rosenblum “Improving Accuracy by Combining Rule-based and Case based Reasoning” 1996.
8. A.M.Wichert “Associative Computation” der Fakultät der Informatik der Univesität Ulm Wrotslav 2000.
9. J.R. Quinlan “Comparing Connectionist and Symbolic Learning Methods” Basser Department of Computer Science University of Sydney, 1990.
10. J.R. Quinlan “Improved Use of Continuous Attributes in C4.5” Journal of Artificial Intelligence Research 1996.
11. P.Scerri, D.Pynadath, M.Tambe “Adjustable Autonomy in Real-world Multi-Agent Environments” Information Institute and Computer Science Department University of Southern California, 2000
12. P.Stone, M.Veloso “Multiagent systems: A Survey from a Machine Learning Perspective.” IEEE Transactions on Knowledge and Data Engineering. 1996.
13. J.Dale, E.Mamdeni “Open Standards for Interoperating Agent-Based Systems” 2000 (last changes)
14. S. Goonatilake, S. Khebbal “Intelligent Hybrid systems” 2002 john wiley & sons

Author information

Agris Nikitenko - Riga Technical University, Division of Systems Theory, Meza street 1/, Riga, LV-1048, Latvia
phone: (+371) 7901045, cell.: (+371) 9424825; E-mail: agris.nikitenko@rembox.lv

МОДУЛЬНАЯ НЕЙРОННАЯ АССОЦИАТИВНАЯ ПАМЯТЬ ДЛЯ ЗАПОМИНАНИЯ ДАННЫХ БОЛЬШОГО ОБЪЕМА

А.М. Резник, А.К. Дехтяренко

***Аннотация**: Предложена новая архитектура нейронной ассоциативной памяти на основе сети Хопфилда, позволяющая преодолеть ограничения емкости памяти одиночной сети при умеренных вычислительных затратах. Рассмотрено влияние параметров архитектуры на процессы запоминания и чтения данных, определены возможные ошибки при чтении данных и получены оценки зависимости между сложностью организации ассоциативного поиска и его результативностью. Полученные теоретические оценки хорошо согласуются с результатами экспериментов, выполненных на массиве случайных векторов данных.*

***Ключевые слова**: ассоциативная память, нейронные сети, сети Хопфилда.*

Введение

Ассоциативная память (АП) обеспечивает доступ к данным не через значения адреса, как это происходит в адресной памяти, а с использованием самих данных. АП принято делить на авто- и гетероассоциативную. Первая сохраняет лишь ключи и может использоваться в задачах фильтрации и восстановления информации; вторая, сохраняя пары ключ-значение, может быть использована в задачах классификации.

Если АП устойчива по отношению к возможным искажениям входной информации, то она может работать с неполными или неточными входными данными. Такими свойствами обладают модели нейронной АП на основе сети Хопфилда [1,2], представляющей собой мультистабильную систему с обратными связями. В такой сети выход *X*, начиная из состояния определяемого входом сети, изменяется по правилу

$$X_{i+1} = sign(CX_i), \qquad (1)$$

где *C* – весовая (синаптическая) матрица *(n×n)*;

n – размерность сети (количество нейронов);

sign – знаковая функция с областью значений *{-1,1}*.

При условии симметрии *C* процесс изменения состояния сети, называемый конвергенцией, всегда заканчивается в устойчивом состоянии - аттракторе. Если аттракторы совпадают с запомненными данными, то процесс конвергенции сети из заданного входом начального состояния в ближайший аттрактор, соответствует ассоциативному поиску наилучшего соответствия запомненным данным.

При использовании предложенного Д. Хопфилдом алгоритма вычисления синаптической матрицы *C,* количество запоминаемых сетью векторов данных ограничено соотношением $m < 0.14n$. Нарушение этого соотношения приводит к появлению ложных аттракторов и разрушению ассоциативной памяти. Предложенный в [3] псевдоинверсный (проекционный) алгоритм обучения, основанный на точном решении уравнения устойчивости нейронной сети, позволяет поднять эту границу до $m < 0.25n$, что составляет половину от теоретического предела $m = 0.5n$ [4]. Позднее, в работе [5] был предложен метод разнасыщения синаптической матрицы, позволивший отодвинуть этот теоретический предел до $m = n$ и обеспечить работу нейронной АП при $m \le 0.75n$ [6].

Основным недостатком нейронной АП является зависимость максимального объема памяти от размерности запоминаемых векторов. Для увеличения объема АП нередко приходится искусственно увеличивать размерность хранимых данных, что приводит к стремительному возрастанию требований к объемам физической памяти и вычислительных ресурсов (квадратичная зависимость от размерности сети).

Этот недостаток можно преодолеть, заменяя одну большую нейронную сеть набором нейронных сетей меньшего размера и применяя определенный способ распределения данных между ними. Именно этот принцип используется в предлагаемой модульной ассоциативной нейронной сети.

Алгоритмы записи и чтения модульной АП

Модульная ассоциативная нейронная сеть (МАНС) представляет собой множество сетей Хопфилда объединенных в структуру типа двоичное дерево (рис. 1). Для обучения сетей используется псевдоинверсный алгоритм, в котором весовая матрица строится как проекционная для запомненных векторов. Критерием распределения данных между модулями служит значение коэффициента различия *d*, который характеризует модуль ортогональной составляющей поступившего вектора относительно линейного многообразию векторов, запомненных в данном модуле. Для *i*-го модуля и входного вектора *X* эта величина определяется как

$$d_i(X) = \|X - C_i X\|^2 / \|X\|^2 = (X \cdot (I - C_i)X)/n, \qquad (2)$$

где для получения последнего равенства учтены проекционные свойства *C* и биполярные значения компонент *X*. Область значений *d* есть *[0,1]*.

В каждом модуле хранится не более *m* запоминаемых векторов, причем заполнение сети начинается с корневого модуля. Для запоминания нового вектора *X* отыскивается модуль, в который он будет занесен. При этом, начиная с корневого модуля, осуществляется подъем по дереву по следующему правилу

$$i := \begin{cases} 2i, & d_i(X) < t \\ 2i+1, & d_i(X) \geq t \end{cases}, \qquad (3)$$

где *i* – номер рассматриваемого модуля (для корневого модуля *i* = *1*),
t – некоторое фиксированное пороговое значение .

Поиск продолжается до тех пор, пока не будет найден первый не полностью заполненный модуль (в котором количество хранимых векторов менее *m*). В этот модуль и будет запомнен входной вектор *X* .

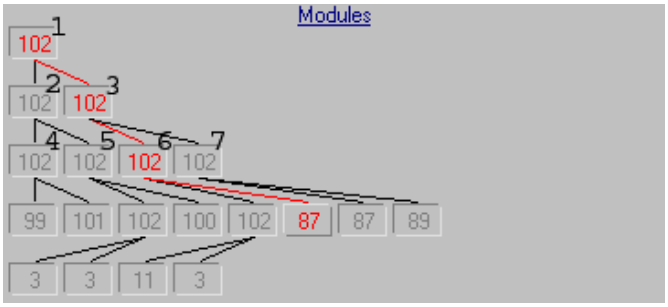


Рис. 1. Структура дерева модулей.

При чтении МАНС в нее поступает входной вектор *X* и требуется определить модуль, который предположительно содержит этот вектор. Для этого строится поддерево поиска аналогично процедуре обучения, однако теперь допускается ветвление – после каждого рассмотренного *i*-го модуля в поддерево включаются один или два модуля следующего уровня:

$$i := \begin{cases} 2i, & d_i(X) < t - \varepsilon \\ 2i+1, & d_i(X) \geq t + \varepsilon \\ \{2i, 2i+1\}, & d_i(X) \in [t-\varepsilon, t+\varepsilon) \end{cases}, \qquad (4)$$

где ε– величина полуинтервала неопределенности.

Модулем, содержащим прототип входного вектора *X*, считается модуль из поддерева поиска, обладающий наименьшим значением коэффициента различия *d*. После того как модуль найден, с ним для входного вектора *X* выполняется обычная процедура извлечения данных из ассоциативной памяти.

Значения параметров *t* и ε оказывают влияние на запоминания данных и чтения АП. Величина *t* определяет, насколько сбалансированным будет сформированное дерево. Слишком малое значение *t* приведет к доминированию правого поддерева относительно любого модуля (включая корневой), слишком большое – левого. Такая ситуация является неблагоприятной, поскольку приводит к протяженным поддеревьям поиска при чтении, что невыгодно с вычислительной точки зрения. Оптимальным значением *t* можно считать медиану распределения плотности вероятности *d*, значение которой существенно зависит от характера запоминаемых данных.

Величина ε определяет интенсивность ветвления поддерева поиска при чтении АП. При $\varepsilon = 0$ ветвление отсутствует, при $\varepsilon = 1$ поддерево поиска совпадает со всем деревом модулей.

Поскольку при чтении АП во входном векторе *X* могут присутствовать искажения, то значения d_i для всех модулей сети могут отличаться от тех, которые были на этапе обучения. Это может привести к выбору неверного модуля для чтения, а значит к ошибочному выходу сети. Неверный выбор может быть обусловлен двумя причинами:

1. ошибка пути, когда построенное поддерево поиска не проходит через модуль, содержащий проверяемый вектор;
2. ошибка принадлежности, когда этот модуль включен в дерево, но не был выбран как модуль с наименьшим значением *d*.

Выбор величины ε влияет на вероятности этих ошибок. Чем больше значение ε, тем больше модулей войдет в поддерево поиска, а значит, тем менее будет вероятность ошибки пути и тем более вероятность ошибки принадлежности.

Распределения значений *d*

Ход процессов записи и чтения данных в МАНС зависит от характера входных данных, и в общем случае его невозможно предсказать a priori. Тем не менее, для определенного типа данных, часто используемого как модельный пример в исследованиях ассоциативной памяти, можно дать некоторые оценки, которые помогут уловить общие закономерности функционирования МАНС.

Пускай запоминаемый массив данных состоит из векторов размерности n с независимыми компонентами, принимающими равновероятные значения {-1,1}. В каждом модуле запоминается m векторов. При псевдоинверсном алгоритме обучения элементы весовой матрицы C имеют распределение, близкое к нормальному [7]. Средние значения диагональных элементов матрицы равны отношению m/n, а недиагональных – нулю. Значения дисперсии элементов C определяется как [6]:

$$D(c_{ij}) = m(m-n)/n^3 . \tag{5}$$

Обозначая G = I – C – проекционная матрица на ортогональное дополнение векторов, хранимых в модуле, оценим первые два момента распределения для величины коэффициента различия:

$$E(d) = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{n} E(x_i g_{ij} x_j) = 1 - m/n ;$$

$$D(d) = \frac{1}{n^2} D\left(\sum_{i=1}^{n}\sum_{j=1}^{n} x_i g_{ij} x_j\right) = D(c_{ij}) = m(n-m)/n^3 . \tag{6}$$

Поскольку в нормальном распределении математическое ожидание и медиана совпадают, то величина E(d) определяет оптимальное значение порога t МАНС, при котором происходит формирование сбалансированного дерева модулей на этапе обучения.

Поведение сети при чтении зависит от характера изменения d под действием шума. Пускай входной вектор X, не содержащийся в данном модуле, искажен случайным шумом интенсивности h, т.е. у h его случайных компонент знак изменен на противоположный. Тогда зашумленный вектор можно представить в виде X + S, где вектор S имеет ровно h компонент с абсолютным значением 2, противоположных по знаку соответствующим компонентам вектора X. Приращение d имеет вид:

$$\Delta d = \left(\|G(X+S)\|^2 - \|GX\|^2\right)/n = \left((S \cdot GS) + 2(X \cdot GS)\right)/n . \tag{7}$$

Распределение Δd имеет условный характер, однако, с целью упрощения оценок, пренебрежем его зависимостью от начального значения d(X).

При равновероятных знаках приращения Δd его распределение должно обладать нулевым средним с дисперсией

$$D(\Delta d) = \frac{1}{n^2}\left(D\left(\sum_{k=1}^{h}\sum_{l=1}^{h} s_{i_k} g_{i_k j_l} s_{j_l}\right) + 4D\left(\sum_{i=1}^{n}\sum_{l=1}^{h} x_i g_{ij_l} s_{j_l}\right)\right) = \frac{16(h^2 + nh)}{n^2} D(c_{ij}), \tag{8}$$

где D(Cij) определяется выражением (5).

Найдем теперь распределение приращения Δd0 для вектора X хранимого в модуле под действием шума интенсивности h. Для такого вектора G X = 0, следовательно:

$$\Delta \mathbf{d}_0 = \left(\mathbf{s} \cdot \mathbf{Gs}\right)/\mathbf{n};$$
$$\mathbf{E}\left(\Delta \mathbf{d}_0\right) = \frac{1}{\mathbf{n}}\mathbf{E}\left(\sum_{\mathbf{k}=1}^{\mathbf{h}}\sum_{\mathbf{l}=1}^{\mathbf{h}} \mathbf{s}_{\mathbf{i_k}}\,\mathbf{g}_{\mathbf{i_k j_l}}\,\mathbf{s}_{\mathbf{j_l}}\right) = \frac{1}{\mathbf{n}}\mathbf{E}\left(\sum_{\mathbf{k}=1}^{\mathbf{h}} \mathbf{s}_{\mathbf{i_k}}^2\,\mathbf{g}_{\mathbf{i_k i_k}}\right) = \frac{4\mathbf{h}}{\mathbf{n}}\left(1-\frac{\mathbf{m}}{\mathbf{n}}\right); \quad (9)$$
$$\mathbf{D}\left(\Delta \mathbf{d}_0\right) = \frac{1}{\mathbf{n}^2}\mathbf{D}\left(\sum_{\mathbf{k}=1}^{\mathbf{h}}\sum_{\mathbf{l}=1}^{\mathbf{h}} \mathbf{s}_{\mathbf{i_k}}\,\mathbf{g}_{\mathbf{i_k j_l}}\,\mathbf{s}_{\mathbf{j_l}}\right) = \frac{16\mathbf{h}^2}{\mathbf{n}^2}\mathbf{D}\left(\mathbf{c}_{\mathbf{ij}}\right).$$

Вероятностные оценки процесса чтения АП

Зная распределения вероятностей для *d*, можно найти вероятности основных событий, играющих важную роль в процессах запоминания данных и чтения МАНС.

Ошибка пути при чтении связана с тем, что хотя бы в одном из модулей поддерева поиска возникло следующее событие:

$$\begin{bmatrix} d(X) < t,\ d(X+S) \geq t+\varepsilon \\ d(X) \geq t,\ d(X+S) < t-\varepsilon \end{bmatrix}$$

Вероятность такого события (прыжка - jump)

$$P_j = \int_0^t f_d(y)\left[\int_{t+\varepsilon-y}^{1} f_{\Delta d}(z)dz\right]dy + \int_t^1 f_d(y)\left[\int_{-1}^{y-(t-\varepsilon)} f_{\Delta d}(z)dz\right]dy = 2\int_0^t f_d(y)\left[\int_{t+\varepsilon-y}^{1} f_{\Delta d}(z)dz\right]dy \quad (10)$$

Пускай теперь поддерево поиска содержит *i*-й модуль с прототипом входного вектора. Ошибка принадлежности (belonging) возникнет, если хотя бы в одном из остальных модулей поддерева коэффициент различия окажется меньшим, чем для *i*-го модуля.

$$P_b = P\left\{d_j(x+s) < d_i(x+s)\right\} = \int_0^1 f_{\Delta d_0}(y)\left[\int_0^y f_d(z)dz\right]dy \quad (11)$$

Если V_i – это множество модулей *i*-го уровня (в дереве, содержащем *l* уровней), то вероятность ошибки пути для произвольного запомненного вектора есть

$$1 - P_{path} = \sum_{i=1}^{l}\left(1-P_j\right)^{i-1} P\left\{x \in V_i\right\} = \frac{1}{2^{l-1}}\sum_{i=1}^{l}\left(1-P_j\right)^{i-1} 2^{i-1} = \frac{\left[2\left(1-P_j\right)\right]^l - 1}{\left(2^l-1\right)\left(1-2P_j\right)} \quad (12)$$

Если же ошибки пути при построении поддерева поиска не произошло, то вероятность ошибки принадлежности при выборе модуля из поддерева будет

$$1 - P_{belonging} = \left(1-P_b\right)^{r-1}, \quad (13)$$

где *r* – количество модулей в поддереве поиска.

Вероятность разветвления (split) поддерева поиска в некотором модуле определяется соотношением

$$P_s = \int_{t-\varepsilon}^{t+\varepsilon} f_d(y)dy, \quad (14)$$

а ожидаемый размер поддерева

$$r = \sum_{i=1}^{l}\left(1+P_s\right)^{i-1} = \frac{\left(1+P_s\right)^l - 1}{P_s}. \quad (15)$$

Эта величина определяет вычислительную сложность процесса чтения АП, поскольку для каждого модуля, кроме листовых, необходимо вычислять величину *d*, что составляет порядка n^2 операций.

Экспериментальные результаты

Полученные соотношения для вероятности ошибок и вычислительной сложности процесса чтения были экспериментально проверены на модельном наборе данных с использованием нейрокомпьютерной программы NeuroLand [8].

В численном эксперименте использовался набор векторов размерности *n = 256*, с независимыми случайными компонентами, принимающими равновероятные значения *{-1,1}*. В каждом модуле

запоминалось *m = 102* вектора, что соответствует 40% насыщению памяти. Величина коэффициента разнасыщения была принята равной 0.1. Уровень шума *h = 33* соответствовал величине полного аттракторного радиуса для одиночной сети, т. е. максимальному уровню искажений, полностью устраняемых сетью в процессе конвергенции. (Заметим, что понятие полного атракторного радиуса отличается от атракторного радиуса, используемого в работах [3,5,6] для обозначения расстояния по Хеммингу, преодолеваемого сетью на последнем шаге конвергенции). Такое значение величины шума позволяет характеризовать качество тестирования сети используя лишь долю правильно отобранных модулей.

Значение порогового параметра *t* определялось исходя из формулы (6). При построении экспериментальных зависимостей по формулам (12) и (15) использовалось значение *l* равное:

$$l = \log_2\left(\frac{M}{m} - 1\right), \tag{16}$$

где *M* – общее количество запомненных в сети векторов.

В первой серии экспериментов определялась вероятность ошибки неправильного выбора модуля при чтении АП в зависимости от количества запомненных векторов при величине полуинтервала неопределенности $\varepsilon = 0.01$. По формулам (10,11,14) были получены следующие величины вероятностей ключевых событий (в скобках указаны экспериментальные значения):

$$P_j = 0.2477\,(0.2275)$$
$$P_b = 1.438 \cdot 10^{-17}\,(0) \tag{17}$$
$$P_s = 0.2557\,(0.1836)$$

На рис. 2 показаны экспериментальная и теоретическая зависимость вероятности ошибки чтения. С ростом заполнения АП теоретическая оценка оказывается несколько завышенной, что вызвано большим теоретическим значениям P_j по сравнению с экспериментальной величиной. Во время эксперимента не было выявлено ошибок принадлежности, что соответствует практически нулевой теоретической величине P_b.

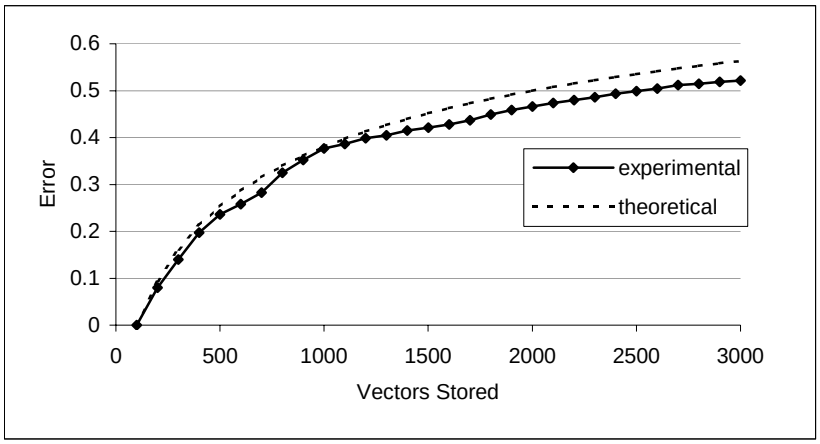


Рис.2 Зависимость вероятности ошибки чтения от заполнения модульной АП

Целью второй серии экспериментов было выяснение зависимости поведения модульной АП от величины полуинтервала неопределенности ε. Рост величины ε вызывает увеличение поддерева поиска. Это может вести к уменьшению количества ошибок пути, однако при этом может возрасти и вероятность ошибки принадлежности. Увеличение ε также сопряжено с увеличением сложности поиска, определяемой отношением средней длины поддерева поиска к его средней длине (*l*) при отсутствии ветвления, т.е. при $\varepsilon = 0$.

Эксперименты проводились на сети, запомнившей *M = 3000* векторов ($l \cong 5$). Чтение данных производилось при различных значениях ε. На рис 3 и 4 показаны экспериментальные и теоретические зависимости для вероятности ошибки и сложности чтения как функции ε. Сопоставление этих зависимостей позволяет выбрать приемлемое значение ε, обеспечивающее компромисс между качеством и сложностью процедуры чтения данных.

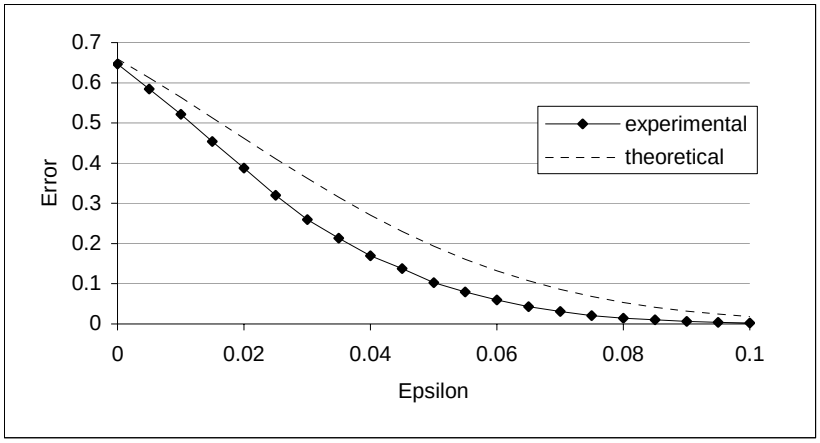


Рис. 3 Зависимость вероятности ошибки чтения от интервала неопределенности

Как и в первой серии экспериментов, ошибок принадлежности не наблюдалось ни для какого значения ε. Теоретическая оценка вероятности ошибки пути также оказались выше экспериментальной зависимости, причем относительное различие между ними возрастает с увеличением интервала неопределенности

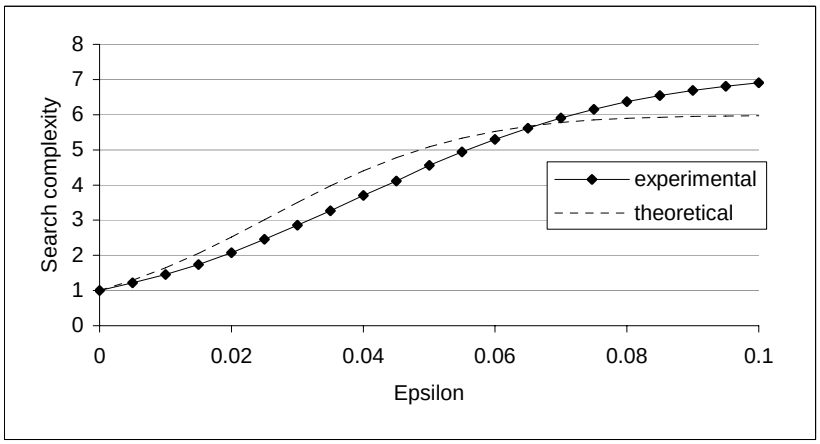


Рис.4. Зависимость сложности чтения от интервала неопределенности

Обсуждение результатов и будущая работа

Рассмотренная модель модульной нейронной ассоциативной сети обеспечивает практически линейную зависимость между количеством запоминаемых векторов данных и необходимым числом ячеек физической памяти, сохраняя при этом основное достоинство сети Хопфилда – способность устранять искажения поступающих данных путем конвергенции в состояния аттракторов. Предложенная нами модель выгодно отличается от известной клеточной нейронной сети, для которой зависимость между количеством связей и размерами сети также близка к линейной [9]. Хотя синаптическая матрица клеточной сети имеет ленточную структуру, напоминающую объединенную матрицу модульной сети, объем ее памяти определяется шириной ленты и не зависит от размеров сети [10]. Поэтому по эффективности ассоциативной памяти клеточная сеть не отличается от обычной сети Хопфилда.

Важным достоинством модульной ассоциативной памяти является возможность применения в ней модулей гетероассоциативного типа. Поскольку в двухслойной гетероассоциативной сети первый слой выполняет функции автоассоциативной памяти, то характер протекания отбора модулей при запоминании данных и при чтении остается неизменным. В то же время полная свобода выбора структуры и функций второго слоя дает возможность сохранять в ассоциативной памяти любые данные, используя для ассоциативного поиска ключи бинарного типа.

Полученные нами соотношения позволяют оценить характер протекания процессов запоминания и чтения данных без их непосредственного осуществления. Это может быть использовано для ускорения подбора параметров сети, оптимального при решении некоторой конкретной задачи. Однако остается неясным насколько эти соотношения, полученные для принятой модели данных, окажутся справедливыми в реальных условиях, при асимметрии и/или значительной корреляции запоминаемых векторов данных.
Очевидно, что предложенная модель имеет большой потенциал для дальнейшего изучения и улучшения. Заслуживает изучения возможность применения других критериев распределения данных, учитывающих характер входных данных. Весьма перспективным является изучение зависимости процесса формирования модульной сети от порядка следования входных данных. Этот процесс, напоминающий самообучение сети Кохонена, может оказаться более эффективным способом кластеризации потока данных.

Библиография

[1] J.J. Hopfield, “Neural networks and physical systems with emergent collective computational abilities,” *in Proc. Nat. Acad. Sci.,* vol. 79, pp. 2554-2558, Apr. 1982.

[2] B. Kosko, “Bi-directional associative memories,” IEEE Trans. Syst., Man, Cybern., vol. 18, no. 1, pp. 49-60, Jan/Feb 1987.

[3] L. Personnaz, I. Guyon, G. Dreyfus, “Collective computational properties of neural networks: New learning mechanisms,” Phys. Rev. A., vol. 34, no. 5, pp. 4217-4228, 1986.

[4] M. Weinfield, “A fully digital integrated CMOS Hopfield network including learning algorithm,” in Proc. Int. Workshop WLSI Art. Intell., Univ. of Oxford, E1-E10, 1988.

[5] A.M. Reznik, D.O. Gorodnichy, A.S. Sitchov, “Regulating feedback bond in neural networks with learning projectional algorithm,” Cybernetics and system analysis, vol. 32, no. 6, pp. 868-875, 1996.

[6] D.O Gorodnichy., A. M. Reznik, “Increasing attraction of pseudo-inverse autoassociative networks,” Neural Processing Letters, vol. 5, no. 2, pp. 123-127, 1997.

[7] Сычев А.С., “Селекция весов в нейронных сетях с псевдоинверсным алгоритмом обучения”, Математические машины и системы, №2, С. 25- 30, 1998.

[8] Резник А.М., Калина Е.А., Сычов А.С., Садовая Е.Г., Дехтяренко А.К., Галинская А.А. “Многофункциональный нейрокомпьютер NeuroLand”, Праці Міжнародної конференції з індуктивного моделювання “МКІМ – 2002” Державний НДІ інформаційної інфраструктури, Львів 20-25 травня 2002, том 2 - секція 4, С. 82-88.

[9] M. Bruccoli, L. Carnimeo, G. Grassi, “Heteroassociative memories via cellular neural networks,” Int. J. Circuit Theory Appl., vol. 26, pp. 231-241, 1998.

[10] А.К. Дехтяренко, Д.В. Новицкий, “Ассоциативная память на базе неполносвязных нейронных сетей”, Труды VIII Всероссийской конференции “Нейрокомпьютеры и их применение” Москва 21-22 марта 2002г.

Информация об авторах

А.М. Резник - Институт проблем математических машин и систем НАН Украины, Киев, ул. Академика Глушкова 42, e-mail: neuro@immsp.kiev.ua

А.К. Дехтяренко - Институт проблем математических машин и систем НАН Украины, e-mail: neuro@immsp.kiev.ua

ЗОНДИРОВАНИЕ ИНТЕЛЛЕКТА НЕЙРОННОЙ СЕТИ ПРИ ОБУЧЕНИИ КЛАССИФИКАЦИИ СЛОЖНЫХ СИГНАЛОВ

Резник А.М., Галинская А.А.

Аннотация: *Выполнено экспериментальное исследование характера информативных признаков, используемых нейронной сетью при обучении классификации сложных сигналов. Применен метод зондирования, с использованием в качестве зонда субоптимального классификатора, отвечающего гауссовой модели обучающей последовательности. Исследовалась сеть прямого распространения с одним скрытым слоем. Эксперименты проведены на массиве данных ультразвуковой локации, используемых для обучения нейронного контроллера системы безопасности пассажиров автомобиля. Показано, что при обучении классификации нейронные сети недоиспользует информацию, отвечающую первым двум моментам распределения вероятностей, однако находят более сложные ассоциативные связи векторов данных, и показывают более высокие результаты, чем субоптимальный классификатор. Параллельное подключение субоптимального классификатора улучшает качество распознавания, тогда как его подключение к входу сети усиливает эффект специализации при обучении.*

Постановка задачи

Как показывает опыт, применяя различные способы комбинирования нейронных сетей в составе многомодульных нейросистем, можно значительно повысить скорость обучения или точность решения сложных задач [1,2]. Выяснению потенциальных возможностей и практическому применению многомодульных нейронных сетей посвящено большое число работ, среди которых преобладают исследования однородных многомодульных структур на основе сетей прямого распространения. Обсуждаются способы преобразования входных данных, и варианты объединения модулей сети [3,4]. Исследуются также многомодульные архитектуры на основе других типов нейронных сетей [4,5,6]. Общий вывод, к которому склоняется большинство исследователей, состоит в том, что многомодульные нейронные сети действуют подобно коллективу экспертов, рассматривающих задачу с различных позиций. Благодаря этому решения, основанные на локальных реакциях отдельных нейронных ансамблей менее подвержены ошибкам.

К сожалению, механический перенос опыта коллективного поведения экспертов - людей на нейронные ансамбли, уровень сложности (и интеллекта) которых пока сопоставим разве что с нервной системой дождевого червя, едва ли правомерен. Конечно, модульная нейронная сеть способна к обучению и принятию самостоятельных решений и можно считать, что она обладает искусственным интеллектом, характеризуемым наличием моделей внешнего мира и языком общения нейронных модулей. Однако такая интерпретация мало способствует выяснению внутренних механизмов формирования и согласования решений, принимаемых нейронными модулями.

В нашем исследовании для выяснения факторов, определяющих поведение обученной нейронной сети, используется метод зондирования с использованием в качестве инструмента статистически оптимального приемника, решающего ту же задачу, что и нейронная сеть. При этом мы исходим из того, что в процессе обучения нейронная сеть стремится к статистически оптимальному поведению при заданной обучающей последовательности и принятом критерии оценки ее ошибок. Располагая статистическими распределениями данных обучающей последовательности, можно создать статистически оптимальное устройство, удовлетворяющее заданному критерию качества работы, а также заранее оценить характеристики поведения, которым будет обладать нейронная сеть после обучения. Объединяя такое оптимальное устройство с нейронной сетью в модульной сети, можно исследовать их взаимодействие при обучении и принятии решений. Варьируя способы объединения, можно выяснить характер информативных признаков исходных данных, используемой нейронной сетью для принятия решений, предложить рациональные способы объединения различных нейронных модулей.

К сожалению, для создания статистически оптимального устройства, необходимо располагать

состоятельными оценками для всевозможных сочетаний значений элементов обучающей последовательности. При больших объемах данных это практически невыполнимо. Однако при слабой статистической зависимости между компонентами векторов данных, обрабатываемых нейронной сетью, можно ожидать выполнения условий центральной предельной теоремы. В этом случае хорошее приближение дает гауссова модель распределения вероятностей обучающей последовательности, учитывающая первые два момента: математические ожидания и ковариации векторов. Получение оценок для этих величин не составляет большой сложности. Используя такие оценки, нетрудно построить и субоптимальное устройство, отражающее основные свойства обучающей последовательности.

Задачей работы является экспериментальное изучение применения данного подхода к анализу поведения реальной нейронной сети. Сравнение поведения субоптимального классификатора и нейронной сети производится на примере решения задачи распознавания сигналов ультразвуковой локации, в системе управления мешками безопасности автомобиля. Анализируя эти сигналы, нейронная сеть оценивает степень безопасности позы пассажира и блокирует работу мешков при наличии угрозы его травмирования. Эксперименты были выполнены при помощи программы MNN CAD [7] с использованием записей сигналов ультразвуковой локации, предоставленных фирмой Automotive Technologies International (ATI Inc., Дэнвилл, Нью-Джерси, США).

Описание исследуемых нейроархитектур

При проведении экспериментов использовались четыре базовые модели классификаторов:

1) Нейронная сеть прямого распространения с одним скрытым слоем нейронов;
2) Простейший перцептрон;
3) Линейный субоптимальный классификатор;
4) Квадратичный субоптимальный классификатор;
5) Объединение субоптимального линейного и квадратичного классификаторов.

Эксперименты выполнялись с помощью комплекса программ MNN CAD [7], позволяющего создавать многомодульные структуры с использованием различных нейронных сетей и включать в их состав дополнительные программные модули. На рис.1 представлена архитектура одного из исследуемых вариантов соединения модулей. В него входит нейронная сеть прямого распространения (модуль B3) и субоптимальный классификатор (B4). Выходы этих модулей поступают в финальный модуль (B5), состоящий из одного нейрона. Модуль B1 служит для нормирования значений компонент входного вектора данных, а модуль B2 – для формирования ожидаемых значений реакции соответствующих модулей при обучении.

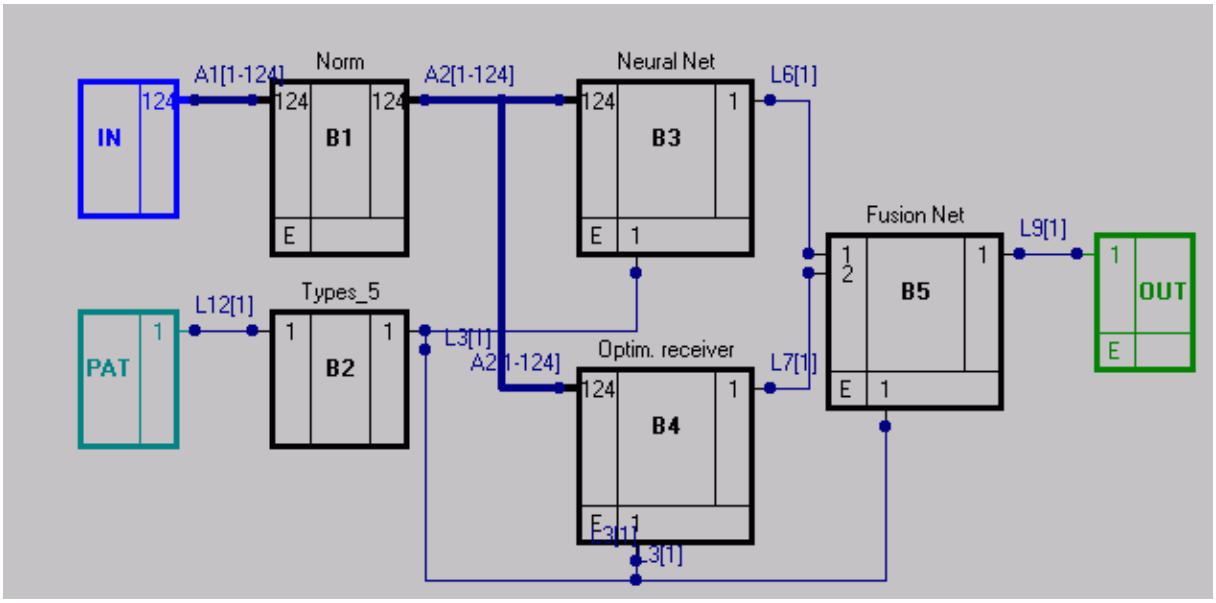


Рис. 1 Архитектура исследуемого многомодульного классификатора (Var2).

В общей сложности было рассмотрено 12 различных вариантов классификатора, включая 5 базовых моделей, объединение сети прямого распространения с перцептроном и 6 вариантов гибридной модульной сети, объединявшей сеть прямого распространения с субоптимальными классификаторами. Были применены два способа построения такой гибридной сети. В первом (Var1) субоптимальные классификаторы подключались к дополнительному входу нейронной сети, т.е. создавалась дополнительная компонента входного вектора представлявшая вещественное значение выхода субоптимального классификатора. Второй способ (Var2) состоял в использовании дополнительного нейрона, на входы которого поступали значения выхода нейронной сети и нормированные значения выхода субоптимального классификатора. При обоих способах обучение нейронной сети выполнялось уже после подключения субоптимального классификатора, т.е. реакция последнего участвовала в формировании решающих правил нейронной сети.

Структура субоптимального классификатора

В рассматриваемой задаче обучающая последовательность содержит векторы *X* двух классов, обозначаемых 0 и 1. Предполагается, что данные имеют нормальное распределение. Плотность распределения вероятностей для *i* - го класса описывается гауссовой функцией:

$$W_i(X) = \frac{1}{\sqrt{(2\pi)^N |\Psi_i|}} exp\left[-\frac{1}{2}(X - A_i)\Psi_i^{-1}(X - A_i)^T \right]. \quad (1)$$

здесь: Ψ_i - ковариационная матрица для *i*-го класса; $|\Psi_i|$ - определитель матрицы;

A_i - математическое ожидание X_i;

X_i^T -транспонированный вектор;

N - размерность A_i, X_i.

Статистически оптимальное решение, обеспечивающее минимальный риск совершения ошибки базируется на пороговой оценке величины *T(X)*, определяющей структуру оптимального классификатора [8]:

$$T(X) = lnW_1(X) - lnW_0(X), \quad (2)$$

Решение в пользу ситуации “0” или “1” принимается по результатам сравнения величины *T(X)* с порогом, зависящим от соотношения между стоимостью потерь при совершении ошибок в ту или иную сторону. Подставив в это выражение значения плотности из (1), найдем:

$$T(X) = \underline{const} + X\left(A_1\Psi_1^{-1} - A_0\Psi_0^{-1}\right)^T - \frac{1}{2}X\left(\Psi_1^{-1} - \Psi_0^{-1}\right)X^T \quad (3)$$

$$\underline{const} = \frac{1}{2}\left(ln|\Psi_0| - ln|\Psi_1| + A_0\Psi_0^{-1}A_0^T - A_1\Psi_1^{-1}A_1^T\right).$$

Зависящие от X последние два члена этого выражения определяют линейную и квадратичную составляющие оптимального классификатора Функцию первой можно представить суммой:

$$S_l = \sum_{i=1}^{N} u_i x_i, \quad (4)$$

где u_i - компоненты вектора оптимального фильтра:

$$U = A_1\Psi_1^{-1} - A_0\Psi_0^{-1}. \quad (5)$$

Квадратичная составляющая определяется выражением:

$$S_s = \frac{1}{2}X\left(\Psi_0^{-1} - \Psi_1^{-1}\right)X^T. \quad (6)$$

Формулы (4)-(6) позволяют синтезировать субоптимальный классификатор, используя выборочные

оценки значений математического ожидания и ковариационных матриц для обучающей последовательности. Если значения ковариационных матриц для обоих ситуаций совпадают, то оптимальный классификатор сохраняет только линейную составляющую. Возможна также ситуация, когда обращается в нуль линейная составляющая, описываемая выражением (5). Тогда оптимальным оказывается квадратичный классификатор (6).

Статистические характеристики обучающей последовательности.

Используемые векторы данных представляют собой последовательности из 124 значений (short integer) сигналов на выходе четырех ультразвуковых датчиков. Данные разбиты на три группы Train – 128000 векторов; Test – 38400 векторов и Valid – 16800 векторов, использовавшихся соответственно для обучения, тестирования в процессе обучения и независимого тестирования (валидации) нейронной сети. Каждый массив содержал примерно поровну векторов класса “0”, соответствующих безопасной позе пассажира и класса “1”, когда необходимо блокировать работу мешков безопасности.

Статистические характеристики векторов, полученные на массиве данных Train, приведены на рис. 2. Четко выделяются диагональные элементы ковариационной матрицы и границы зон датчиков. Разность ковариационных матриц в правом окне представляет наиболее выраженные классификационные признаки сигналов локации.

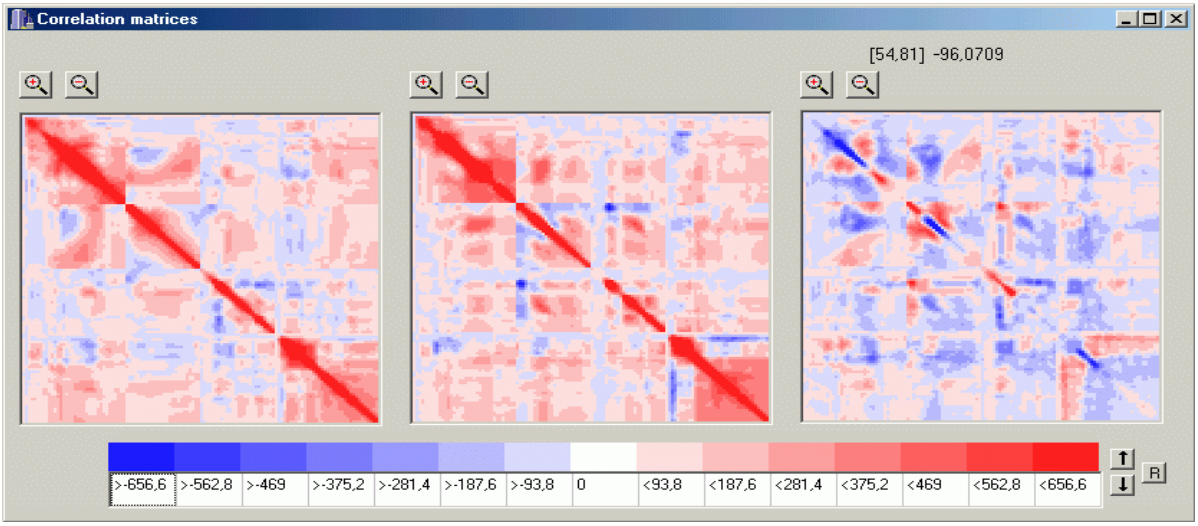


Рис.2 Ковариационные матрицы для классов “0” и “1” их разность (справа).

Приведенные на рис.2 результаты анализа данных массива Train были использованы для расчета компонент оптимального классификатора по формулам (4-6). На рис.3 представлены результаты расчета оптимального линейного фильтра *U*.

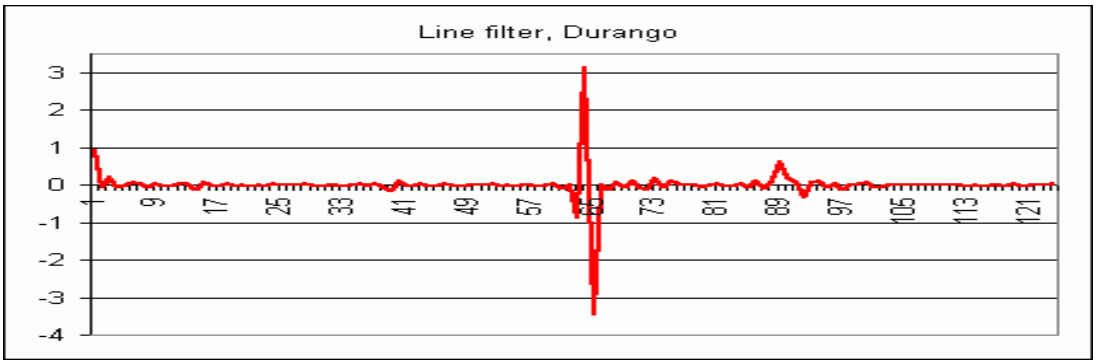


Рис.3. Оптимальный линейный фильтр.

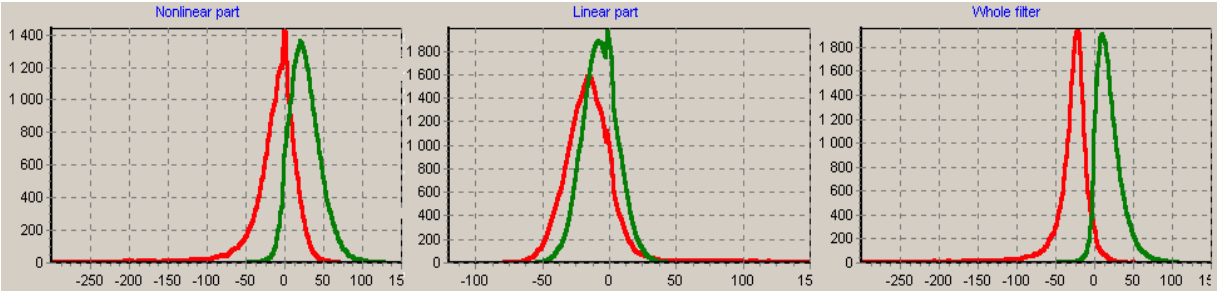
Нейронные модули

Проведению исследования модульной сети с использованием субоптимального классификатора предшествовала серия экспериментов по выбору архитектуры нейронной сети. Для рассматриваемой задачи классификации наилучшие результаты были получены для трехслойной нейронной сети с 15 нейронами в скрытом слое. Использовалась сигмоидная активационная функция нейронов со смещением и алгоритм обучения EDBD [9]. Входные данные предварительно нормировались в диапазоне (0, 1). Нормировка выполнялась по всей обучающей выборке (Train), отдельно для каждого входа сети. Начальные значения веса связей этого модуля задавались датчиком случайных чисел. Для устранения разброса результатов каждый эксперимент с этим модулем повторялся 5 раз и полученные оценки осреднялись.

Модуль перцептрона содержал один нейрон со знаковой (sign) активационной функцией, обучаемый по правилу Хебба. При обучении этого модуля начальные значения веса связей устанавливались равными нулю.

Результаты автономного тестирования модулей

На рис. 4 приведены гистограммы реакции субоптимального классификатора для двух классов векторов данных. Изображения соответствуют (слева направо) квадратичному классификатору, линейному, а также их объединению Под изображениями даны значения порога и средние процентные значения числа ошибок для каждого класса ситуаций. Отметим, что результаты классификации как для линейного, так и для квадратичного вариантов оказываются намного худшими, чем при их объединении.



Class	(threshold = 7)	(threshold = - 5)	(threshold = - 4)
0 / 1	19.29/ 17.81	25.55 / 58.75	5.48 / 4.14
Total %	18.55	42.15	4.81

Рис.4 Гистограммы реакций для квадратичного и линейного субоптимальных классификаторов, а также их объединения на данных массива Train.

Сравнительные результаты тестирования всех базовых моделей классификатора на массивах Train и Valid приведены в табл.1.

Таблица 1. Процент ошибок классификации для базовых модулей

Type	Train %	Valid %
SO/Linear	42.15	37.92
SO/Square	18.25	26.85
SO/Lin. + Sq.	4.81	5.92
Perceptron	6.02	4.83
Neural network	1.98	3.17

Сравнивая эти результаты, можно отметить, что наилучшие показатели дает нейронная сеть прямого распространения. Перцептрон и субоптимальный классификатор даже при объединении линейной и квадратичной составляющих существенно отстают от нейронной сети по количеству ошибок

классификации. На массиве данных Train субоптимальный классификатор обеспечивает более высокие показатели, чем перцептрон, но заметно худшие – на массиве Valid. Из этого можно заключить, что нейронная сеть находит при обучении более сложные ассоциативные связи между составляющими векторов данных, чем те, что содержатся в средних значениях и функциях взаимной корреляции, учитываемых моделью многомерного нормального распределения. В какой-то степени это относится и к перцептрону, решения которого базируются на линейном преобразовании входных данных, аналогичных тем, которые использует линейный субоптимальный классификатор.

Эксперименты с гибридными сетями

Цель экспериментов состояла в выяснении того, насколько существенны для нейронной сети значения тех классификационных признаков, которыми пользуется субоптимальный классификатор. Были проведены испытания гибридных сетей Var1 и Var2, при включении в их состав линейного, квадратичного субоптимального классификатора, а также их объединения. Таким образом, нейронная сеть до начала обучения получала всю полезную информацию, извлекаемую субоптимальным классификатором из входных данных. Также исследовался вариант гибридной сети Var2, при котором вместо субоптимального классификатора подключался заранее обученный перцептрон.

Обучение проводилось на массиве Train по технологии SaveBest с промежуточным тестированием на массиве Test при общем объеме обучающей последовательности до 1,5 млн. векторов. Отобранный наилучший вариант сети затем проходил тестирование на массивах Train, Test и Valid. Каждый эксперимент повторялся 5 раз с различной инициализацией синаптической матрицы. В каждой серии экспериментов фиксировались средние значения (Avg.), а также наименьшие (Min) значения процента ошибок классификации.

Результаты испытаний приведены в Таблице 2. Данные для исходной нейронной сети в нижней строке колонки Var2 получены при автономном тестировании нейронной сети, обученной в составе гибридной сети Var2.

Рассматривая результаты тестирования на массиве Train, можно отметить, что подключение субоптимального классификатора приводит к сокращению числа ошибок классификации для обоих вариантов гибридной сети, причем сокращение наиболее выражено при суммировании линейной и квадратичной компонент классификатора. При испытаниях на массиве Test такого улучшения не наблюдается, а на массиве Valid число ошибок для гибридной сети Var1 даже возрастает. Противоположный результат дает испытание на этом массиве сети Var2 - количество ошибок гибридной сети сокращается почти на треть.

Таблица 2 Сравнительные результаты испытаний двух вариантов гибридной сети

Mode	Test mode	Var1		Var2	
		Avg.%	Min%	Avg.%	Min%
SO/ Linear	Train	2.02	1.78	1.88	1.75
	Test	4.39	4.10	4.34	4.29
	Valid	3.59	3.32	3.53	3.33
SO/ Square	Train	2.13	1.74	1.92	1.79
	Test	4.35	4.17	4.39	4.18
	Valid	3.49	3.18	3.46	3.20
SO/ Lin. + Sq.	Train	1.87	1.64	1.74	1.63
	Test	4.31	4.23	3.98	3.77
	Valid	3.56	3.38	2.97	2.67
Perceptron	Train			1.89	1.76
	Test			4.28	4.22
	Valid			3.26	3.13
Neural network	Train	2.23	1.98	2.09	1.94
	Test	4.31	4.29	4.49	4.15
	Valid	3.48	3.17	3.67	3.28

Подключение перцептрона в гибридной сети Var2 также приводит к сокращению числа ошибок классификации на всех массивах, однако качество классификации на массивах Test и Valid уступает полученному при подключении обеих компонент субоптимального классификатора.

Количественную оценку степени информативности субоптимального классификатора при формировании реакции гибридной нейронной сети дает сравнения значений веса связей на входах дополнительного нейрона обученной гибридной сети Var2. Такие оценки для субоптимального классификатора приведены в табл. 3. Использованы следующие обозначения: W_{NN} - вес связи с выходом нейронной сети, W_{SO} – веса связей с компонентами субоптимального классификатора, $k=W_{SO}/(W_{NN}+W_{SO})$ – степень информативности соответствующей компоненты. Для линейной составляющей она оказывается наименьшей – около 15%. Для квадратичной компоненты, учитывающей различие ковариационных матриц векторов данных, она значительно выше – более 25%. Наивысшей, как и следовало ожидать, оказывается информативность суммы компонент - более 37%. Достаточно неожиданной оказалась оценка информативности перцептрона– ниже 3%.

Таблица 3 Оценка информативности гауссовых компонент в гибридной нейросети

Mode	W_{NN}	W_{SO}	$k = W_{SO}/ (W_{NN}+ W_{SO})$
SO/ Lin.	4.92	0.91	0.156
SO/ Sq.	4.74	1.63	0.256
SO/ Lin. + Sq.	4.15	2.49	0.376
Perceptron	3.95	0.11	0.027

Обсуждение результатов экспериментов

Использованный нами экспериментальный материал достаточно хорошо отвечает модели многомерного нормального распределения, и можно было ожидать, что нейронная сеть будет использовать, в основном те же классификационные признаки, что и субоптимальный классификатор, реализующий эмпирическую статистику обучающей выборки. Оказывается, что это далеко не так. Нейронная сеть в процессе обучения находит сложные ассоциативные связи между элементами данных, недоиспользуя более простые корреляционные зависимости, на которые опирается субоптимальный классификатор. Это прослеживается на данных табл. 2., где отчетливо видно, что подключение субоптимального классификатора всегда приводит к снижению числа ошибок нейронной сети. Улучшение классификации особенно заметно при подключении к нейрону, формирующему выход сети (Var2), причем в этом случае этот эффект наиболее выражен при тестировании на данных массива Valid. Это очень важный результат, свидетельствующий об улучшение способности к обобщению запоминаемых данных, и указывающий на относительную независимость критериев, используемых нейронной сетью и субоптимальным классификатором при принятии решений.

При подключении субоптимального классификатора к входу нейронной сети (Var1) улучшение классификации происходит только на данных массива Train, тогда как на массивах Test и Valid оно почти незаметно. Это свидетельствует об усилении эффекта специализации сети при подключении субоптимального классификатора. Подобный эффект наблюдается впервые и пока не имеет объяснения. Можно лишь предположить, что слой скрытых нейронов сети блокирует поступление информации от субоптимального классификатора на ее выход. Природа этого явления пока неясна, однако практические выводы можно сделать уже сейчас: при конструировании гибридных сетей высокоинформативные источники информации целесообразно подключать поближе к выходу нейронной сети.

Весьма неожиданной оказалась крайне низкая оценка влияния перцептрона при его параллельном подключении к выходу нейронной сети. Его вес оказался ниже 3%, что намного ниже линейного и квадратичного субоптимальных классификаторов (15% и 25%) и полностью противоречит результатам табл. 2. Можно предположить, что в данном случае информативные признаки, формируемые нейронной сетью при обучении, имеют тот же характер, что и у перцептрона, но обладают значительно большей мощностью. Поэтому вклад перцептрона в окончательное решение оказывается незначительным.

Библиография

1. Amanda J.C. Sharkey, On combining artificial neural nets.
2. Amanda J.C. Sharkey, Noel J. Sharkey, Combining diverse neural networks
3. Giorgio Giacinto, Fabio Roli, Design of effective neural network ensembles for image classification purposes
4. Agnes Crepet, Helen Paugam-Moisy, Emanuelle Reynaud, Didier Puzenat, A modular neural model for binding several modalities
5. Bin Tang, Malcolm I. Heywood, Michael Shepherd, Input Partitioning to Mixture of Experts
6. Bart L.M. Happel, Jakob M.J. Murre, The design and evolution of modular neural network architecture.
7. А.М. Резник, М.Э. Куссуль, А.С. Сычов, Е.Г. Садовая, Е.А. Калина Система проектирования модульных нейронных сетей САПР МНС. – Труды 8-й Всероссийской конференции «Нейрокомпьютеры и их применение», Москва, 21-22 Марта 2002
8. Д. Миддлтон Введение в статистическую теорию связи. - т2.- М.: Сов. Радио. 1962.- 831с.
9. R.D. Reed and R.J. Marks, Neural Smithing. MIT Press. - 1999. - 346с.

Информация об авторах

А.М. Резник – Институт Проблем Математических Машин и Систем НАН Украины, пр. Глушкова 42, Киев, Украина, neuro@immsp.kiev.ua

А.А. Галинская – Институт Проблем Математических Машин и Систем НАН Украины, пр. Глушкова 42, Киев, Украина, neuro@immsp.kiev.ua

НЕЙРОСЕТЕВАЯ МОДЕЛЬ РАНЖИРОВАНИЯ ПРЕДИКТОРОВ ПРИ КРАТКОСРОЧНОМ ПРОГНОЗИРОВАНИИ ПАВОДКОВ

А.М. Резник, К.М. Кужель.

***Аннотация**: Применение нейронных сетей для прогнозирования основано на том, что при обучении на данных предистории прогнозируемого процесса нейронная сеть приобретает свойства модели этого процесса. Это позволяет использовать искусственные нейронные сети, строение которых обычно известно, для изучения внутренних механизмов сложных процессов, прямое исследование которых часто затруднено или даже невозможно. В работе проводится такое исследование на примере ранговой оценки предикторов при краткосрочном прогнозировании наводнений в регионе Карпат с использованием нейронных сетей прямого распространения.*

1. Введение

Обучение нейронной сети решению некоторой задачи иногда рассматривают как формирование модели запоминаемых примеров, в форме разложения исходных данных задачи в нейросетевом базисе [1,2]. На уровне отдельного нейрона такое разложение предусматривает две элементарные операции – вычисление постсинаптического потенциала и формирование реакции нейрона путем нелинейного преобразования величины этого потенциала. Первая операция является линейной и задается значением матрицы веса связей, вычисляемой в процессе обучения нейронной сети. Обучение обычно выполняется по методу обратного распространения ошибки, позволяющему с необходимой точностью приблизить значения реакции сети к значениям, заданным в примерах обучающей последовательности. Полученные при обучении значения веса связей отражают информацию о наблюдаемом объекте, содержащуюся в

обучающей последовательности. Зная структуру связей между нейронами сети и характер линейных и нелинейных преобразований в нейронах, можно представить эту информацию в более удобной для интерпретации форме и использовать для прогнозирования поведения объекта в конкретных условиях.
Эти соображения позволяют рассматривать обучаемую нейронную сеть как средство для исследования сложных систем без активного вмешательства в их поведение. Такой способ изучения может быть особенно эффективным для приложений, связанных с моделированием и управлением в сложных экологических, экономических и технических системах, где последствия вмешательства могут быть непредсказуемы. В подобных задачах часто важна не столько интерпретация поведения всей системы, сколько оценка взаимного влияния отдельных наблюдаемых факторов. Такие оценки можно получить путем анализа значений весов связей нейронной сети, обученной на всей имеющейся совокупности данных наблюдений поведения исследуемого объекта. Более полную информацию о поведении наблюдаемой системы можно получить при варьировании параметров нейронной сети в процессе обучения и анализе изменений значения веса связей. Исключая связи, веса которых близки к нулю, можно упростить формируемую нейронной сетью модель, выявить главные факторы, определяющие поведение наблюдаемой системы. Подобный метод селекции связей достаточно часто применяют для оптимизации параметров нейронной сети при решении прикладных задач [3,4]. В работах [5,6] аналогичный метод использован для поиска информативных признаков при распознавании образов.
В нашей работе рассматривается применение нейронной сети для прогнозирования наводнений на горных реках региона Карпат. Предварительные данные о разработанной системе краткосрочного прогнозирования содержатся в работе [7]. В настоящей работе, продолжающей эти исследования, приводятся новые результаты нейросетевого моделирования, связанные с оценкой значимости предикторов - компонент гидрометеорологических данных.

2. Метод нейросетевого моделирования

Создаваемую при обучении нейронной сети модель обучающего массива данных можно представить как функцию *Y= F(W, X),* где *X* и *Y* - значения векторов на входе и выходе сети, а *W* – набор значений весовых коэффициентов для входов нейронов сети. Для нейронной сети прямого распространения с одним скрытым слоем нейронов такая модель имеет вид:

$$y_i = f_2[\sum_{j=1}^{n_1} w_{i,j}^2 f_1(\sum_{k=1}^{n_0} w_{j,k}^1 x_k - b_1) - b_2], \quad (1)$$

где: $f_l(\ )$ – активационная функция нейронов *l*- го слоя сети;
y_i – значение реакции на *i*-м выходе нейронной сети;
x_k– значение *k*- го входа сети;
$w_{i,j}^l$ - вес связи для *j* –го входа *i*- го нейрона *l*- го слоя сети;
b_l – значение порога для нейронов *l*- го слоя сети;
n_0, n_1- соответственно число входов и число нейронов скрытого слоя.

Активационная функция обычно имеет вид сигмоиды. При малых значениях аргумента реакция нейрона почти линейно зависит от воздействия на его входе. При больших значениях аргумента выход нейрона практически перестает зависеть от изменения значений воздействий на его входах. Используя разложение выражения (1) в степенной ряд, можно изучать поведение сети в окрестности любого заданного значения входного вектора X, используя линейное приближение:

$$\Delta Y \approx [\partial F(W, X) / \partial X]\Delta X,$$

или

$$\delta y_i \approx \sum_{k=1}^{n_0} a_{ik} x_k; \quad (2)$$

где:

$$a_{ik} = f_2^{(1)}[\sum_{j=1}^{n_1} w_{i,j}^2 f_1(\sum_{k=1}^{n_0} w_{j,k}^1 x_k - b_1) - b_2]\sum_{j=1}^{n_1} f_1^{(1)}(\sum_{k=1}^{n_0} w_{j,k}^1 x_k - b_1) w_{i,j}^2 w_{j,k}^1; \quad (3)$$

$f_l^{(1)}(\ldots)$ - значение производной активационной функции *l*- го слоя.

Описываемая выражением (1) нейронная сеть является статической в том смысле, что значения векторов *X* и *Y* не содержат компонент, явно зависящих от времени. Это создает определенные неудобства при моделировании динамических объектов, в частности, решении задач прогнозирования. В таких случаях

динамику процесса учитывают путем представления последовательности наблюдаемых величин в форме компонент векторов *X* и *Y*. Интервал наблюдения процесса задают с помощью скользящего окна. Иногда для представления динамики вместо наблюдаемых значений процесса на входы нейронной сети подают разностные оценки текущих значений его производных. Этот способ обычно используют в нейроконтроллерах [8].

Как следует из (3), вклад каждого входа в формирование поведения сети определяется значениями произведений коэффициентов $w^2_{i,j}w^1_{j,k}$ для связей, образующих путь к соответствующему выходу сети.

Степень влияния для каждой пары (*k,i*) зависит от текущего состояния сети, определяемого значением вектора *X*, причем, из-за нелинейности производных активационных функций эта зависимость имеет достаточно сложный характер. Однако, учитывая наличие большого числа путей от каждого входа к выходу сети, а также то, что веса связей от скрытых нейронов к выходу сети формируются под влиянием большого числа входов, можно ожидать, что суммарный эффект влияния входов определяется, в основном, значениями веса их связей со скрытыми нейронами. Исходя из этих соображений, для укрупненной оценки степени влияния входа на реакцию сети можно использовать значения квадрата модуля вектора весовых коэффициентов:

$$c_k = \sum_{j=1}^{n_1} (w^1_{j,k})^2, \qquad (4)$$

Эти величины нельзя непосредственно применить в модели поведения наблюдаемой системы, поскольку для построения такой модели необходимы экспериментальные оценки значений коэффициентов выражения (3). Однако, ранжируя полученные значения c_k,, можно выстроить приоритетный ряд факторов, влияющих на поведение системы, что во многих случаях оказывается достаточным для его интерпретации.

Для оценки устойчивости результатов ранжирования входов нейронной сети, при проведении экспериментов производилось также вычисление сумм значений парных произведений весовых коэффициентов:

$$g_{i,k} = \sum_{j=1}^{n_1} (w^2_{i,j}w^1_{j,k})^2. \qquad (5)$$

3. Описание эксперимента

В рассматриваемой задаче краткосрочного прогнозирования паводков была использована нейронная сеть прямого распространения с одним скрытым слоем нейронов, и одним линейным нейроном на выходе. Величина постсинаптического потенциала этого нейрона указывала прогнозируемое значение уровня воды в заданном пункте. На входы нейронной сети поступали нормированные значения гидрометеорологических показателей в пунктах наблюдения за предшествующий период наблюдения.

Исходные данные для проведения экспериментов представляли собой результаты замеров среднесуточных значений температуры (T, ^{0}C), осадков (P, mm), расхода (Q, m^3/s) и уровня воды (H, cm) за 5- летний период с 1 января 1996р. до 31 декабря 2000р для 14 станций Укргидромета. Из них 5 находились в бассейне р. Уж: Ужгород (T, P, Q, H), Жорнава (P, Q, H), Заричиве (P, Q, H), Великий Березний (T, P) и Симер (P, Q, H). Остальные 9 станций находились в бассейне р. Латорица: Нижние Ворота (T, P), Подполозье (P, H, Q), Свалява (P, H, Q), Мукачево (H, Q), Чоп (P, H, Q), Нелипино (H, Q), Зняцево (P, H, Q), Межгорье (T, P), Плай (T, P). Расположение станций приведено на карте рис.1.

На рис 2 приведен график уровня воды в районе Ужгорода за весь период наблюдений. Максимумы наводнения наблюдались 5 ноября 1998г. (+337 cm), 6 марта 1999р. (+210 cm) и 10 марта 2000р. (+203 cm). Паводок 1998 р. был наибольшим и имел катастрофические последствия для всего Закарпатья.

Набор измерений по всем пунктам наблюдения за каждые сутки (всего 37 чисел) составлял вектор данных, поступавший на входы нейронной сети. Массив, включавший 1813 векторов данных, был разбит на две части, одна из которых, охватывавшая период с 1 января 1996г. по 31 декабря 1999г., использовалась только для обучения нейронной сети, а вторая (с 1 января по 31 декабря 2000г.) для прогнозирования. Часть экспериментов была выполнена с использованием укороченных векторов данных, охватывавших только бассейн р. Уж (15 чисел). Во всех случаях прогнозирование уровня воды выполнялось для района г. Ужгород.

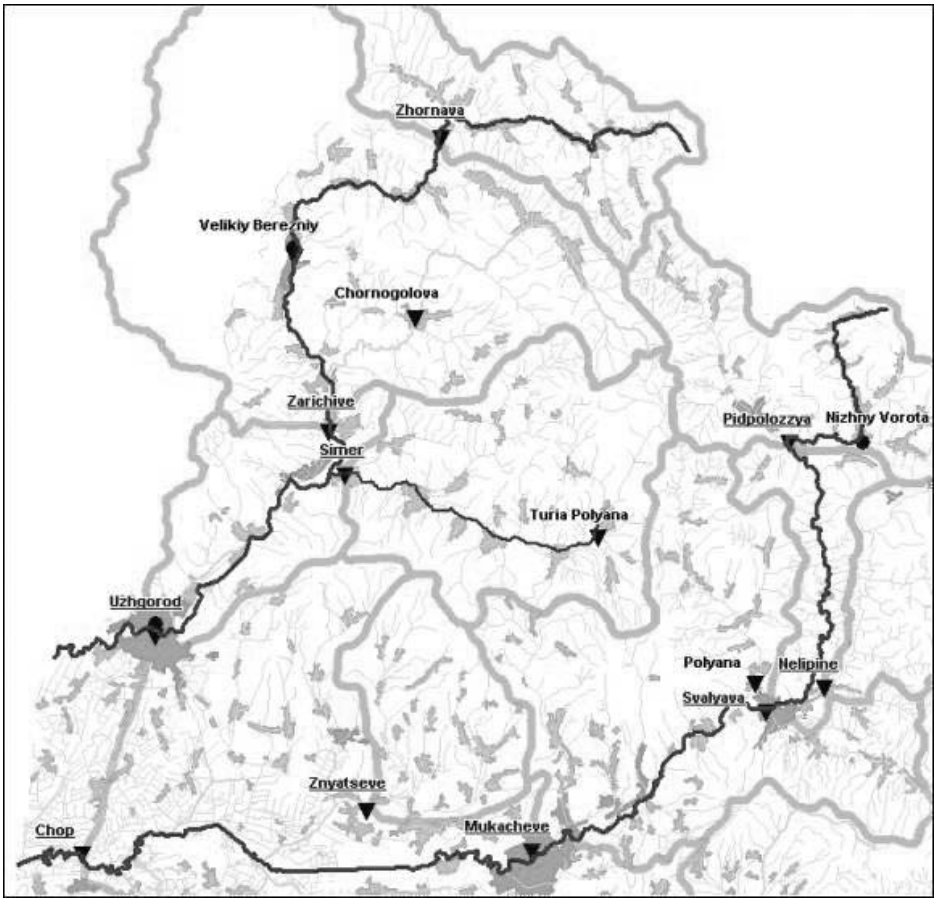


Рис.1 Бассейны рек Уж и Латорица.

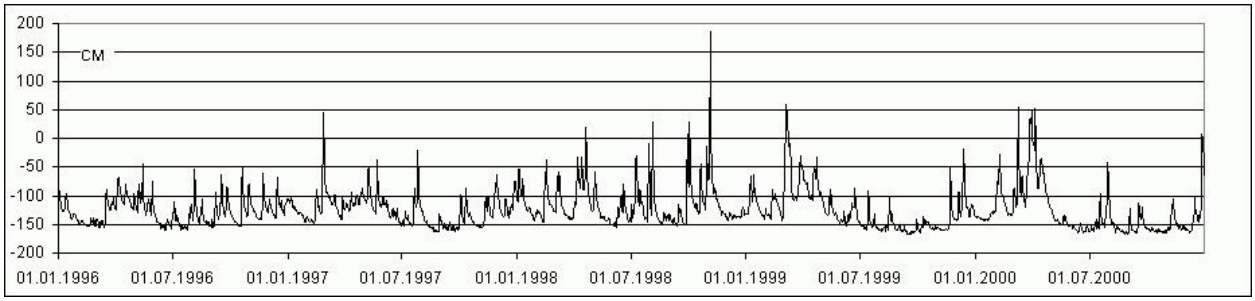


Рис.2 Уровень воды р. Уж в районе Ужгорода с 01.01.1996 по 31.12.2000.

Эксперименты проводились с помощью нейрокомпьютера MNN CAD [9], на котором была реализована нейронная сеть и средства препроцессинга, обеспечивавшие нормирование и формирование последовательностей векторов данных. Для обучения нейронной сети применен алгоритм обратного распространения ошибки EDBD с индивидуальной настройкой и постепенным сокращением параметров скорости и момента [10]. Использовался некумулятивный режим обучения с контролируемым уровнем среднеквадратичной ошибки 1%.

4. Результаты прогнозирования

При обучении на входы сети поступали векторы, описывающие гидрометеорологическое состояние за период наблюдения (3-15 дней), а на ее выходы – значения уровня воды в 4 пунктах наблюдения на последующий день. Такими пунктами были: Ужгород, Заричиве, Симер и Жорнава. При тестировании обученной сети результаты прогнозирования в этих пунктах сравнивались с данными фактических

измерений и вычислялись частоты превышения абсолютной величиной ошибки заданных пороговых значений. Полученные вариационные ряды позволяли оценить надежность прогнозирования. Сравнение полученных результатов показало, что наименьшую ошибку дает прогнозирование на основании гидрологических данных за последние три дня, при использовании всего массива данных, собираемых в бассейнах обоих рек. На рис.3 даны результаты прогнозирования весеннего паводка 2002г. в районе Ужгорода, основанные на использовании данных только по бассейну р. Уж и для случая, когда использовались данные из обоих бассейнов (Уж + Латорица).

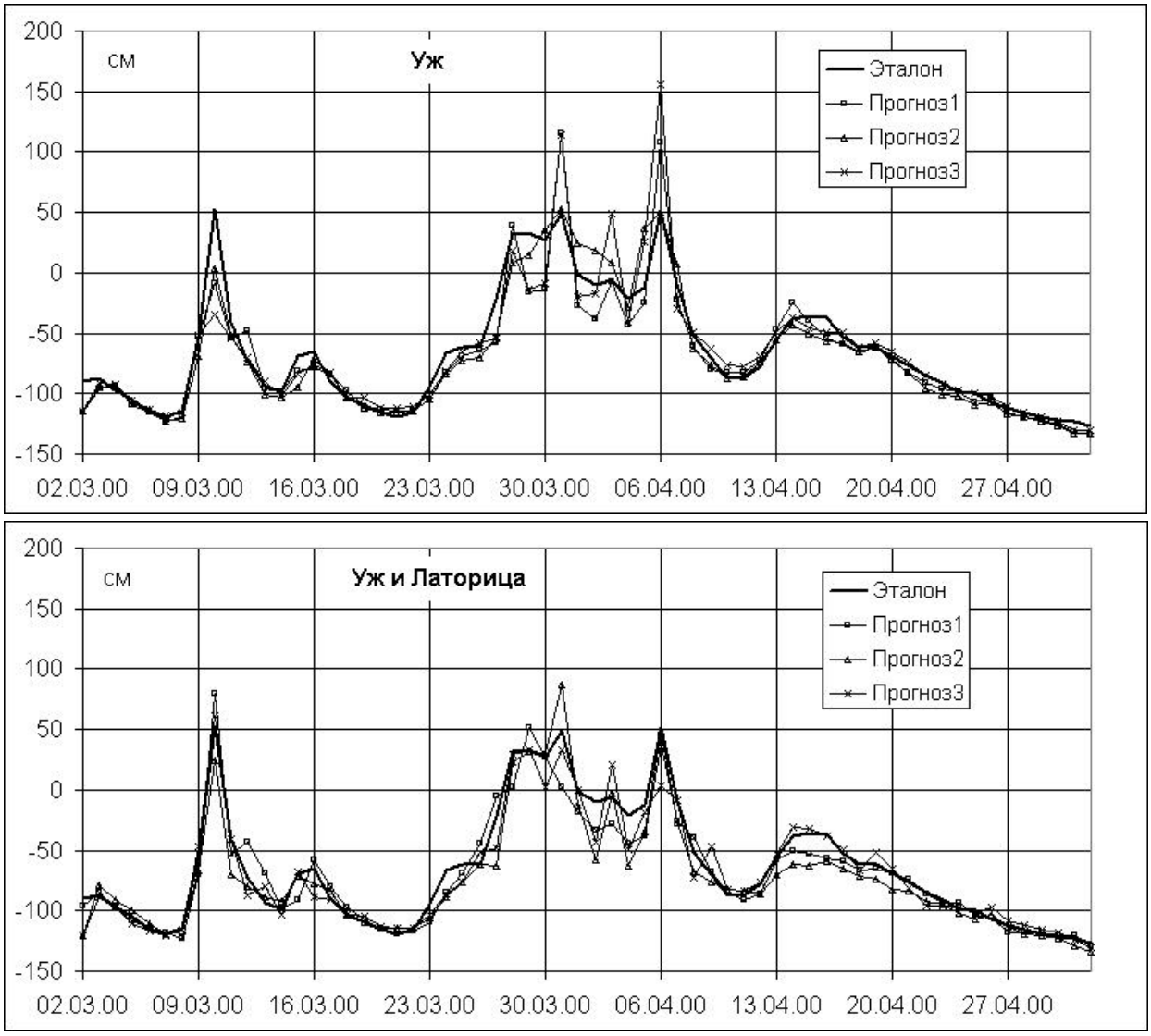


Рис.3 Данные прогноза весеннего паводка 2000г в г. Ужгород.

Приведенные результаты прогноза содержат по три реализации, соответствующие различной начальной инициализации нейронной сети при обучении. Можно отметить, что прогноз, основанный на использовании наблюдений в обоих бассейнах, дает более точные оценки, причем наиболее точным оказывается прогнозирование уровней высокой воды. Последнее находит подтверждение при анализе вариационного ряда абсолютной величины ошибок прогнозирования. На рис. 4 приведены зависимости между величиной доверительного интервала и вероятностью того, что абсолютная величина ошибки прогнозирования не выйдет за его пределы, полученные по экспериментальным данным прогноза за 2000-й год. Из них следует, например, что при прогнозировании по гидрометеорологическим данным для бассейна р.Уж в 99% случаев абсолютная величина ошибки прогноза не превысит 58 см.. Если же использовать для прогноза также и данные бассейна Латорицы, то эта величина снижается до 40 см. При снижении требований к надежности прогноза ниже 95% оба результата оказываются практически одинаковыми.

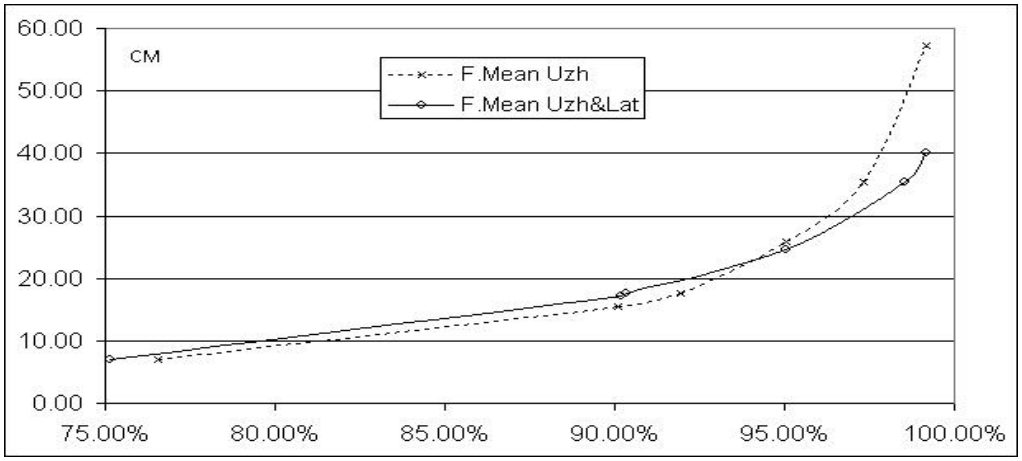


Рис. 4 Величина доверительного интервала ошибки прогнозирования паводка в районе г. Ужгород.

5. Оценка значимости предикторов

Для сравнительной оценки значимости различных компонент исходных данных при прогнозировании использовались соотношения (4) и (5). Для каждого входа сети вычислялись значения коэффициентов c_i и g_i и производилось их ранжирование. В табл.1 и 2 приведены списки 12 первых членов ряда соответственно для c_i и g_i, полученных при прогнозировании по данным за предшествующие 3 или 7 дней для одного и двух речных бассейнов. Обозначения предикторов следующие: первая буква обозначает измеряемый параметр (H –высота воды; P- уровень осадков; Q – расход воды) следующее сочетание из двух букв - пункт наблюдения (uz –Ужгород, zh –Заричиве, za –Жорнава, si -Симер, mi – Межгорье, pi – Подполозье, vb- Великий Березний). В скобках указан день замера. В колонках даны значения параметра c_i для данного предиктора. В двух последних колонках в скобках указаны ранги предикторов.

Анализируя данные таблиц, можно заметить, что общий состав предикторов в верхней части ранжированного списка сохраняется во всех вариантах, но их роли заметно перераспределяются. Оба критерия c_i и g_i дают примерно одинаковый список приоритетов, в котором преобладают предикторы, характеризующие уровень воды накануне в пунктах наблюдения выше по течению реки, а также уровень осадков в верховье реки. Это вполне объяснимо, поскольку время пробегания волны наводнения от верховья р. Уж составляет менее 10 часов. Результаты оценок g_i, приведенные в табл.2 позволили объяснить причины зависимости между гидрологическими режимами речных бассейнов р.р. Уж и Латорица, изолированных друг от друга горным хребтом. Попадание предикторов Pmi и Ppi, характеризующих уровни осадков в долине р. Латорица в числе приоритетных для прогноза уровня воды в г. Ужгород указывает на то, что эта зависимость обусловлена влиянием осадков, выпадающих в обоих бассейнах практически одновременно. Интересно отметить, что те же предикторы в списке приоритетов для коэффициента c_i оказались заметно ниже – на 16 –м (Pmi) и 20 -м (Ppi) местах.

Табл. 1 Распределение значений коэффициента c_i для 12 старших предикторов

	Уж+Латорица 3 дня	Уж 3 дня	Уж+Латорица 7 дней
Hzh -(0)	608	(2) 476	(1) 495
Hsi –(0)	374	(1) 500	(2) 235
Hza –(0)	286	(3) 289	(3) 221
Qzh- (0)	173	(5) 172	(4) 112
Pzh –(0)	143	(8) 114	(8) 44
Hzh -(-1)	123	(9) 112	(9) 44
Huz –(0)	113	(4) 213	(5) 71
Hsi -(-1)	109	(10) 111	(6) 66
Hzh -(-2)	86	(7) 127	(13) 25
Hza - (-1)	75	(13) 91	(10) 40
Tvb- (0)	66	(12) 93	
Hsi -(-2)	59	(14) 59	(12) 30

Табл.2 Распределение значений коэффициента *g* для 12 старших предикторов

	Уж+Латорица, 3 дня	Уж 3дня	Уж+Латорица, 7 дней
Huz-(0)	621	(1) 966	(3) 172
Pzh-(0)	507	(4) 597	(6) 89
Hzh-(0)	385	(2) 731	(1) 368
Qzh-(0)	226	(6) 435	(4) 145
Hza (0)	182	(5) 514	(2) 353
Hzh-(-1)	166	(8) 294	(10) 52
Hzh-(-2)	163	(9) 251	(12) 30
Hsi -(0)	156	(3) 655	(9) 59
Huz-(-1)	150	(10) 179	(8) 60
Pmi- (0)	121		(11) 43
Hza-(-1)	113	(7) 359	(7) 85
Ppi –(0)	94		(5) 98

Обсуждение результатов

Полученные с помощью нейронной сети ранговые оценки влияния гидрометеорологических данных при прогнозировании паводков достаточно хорошо совпадают с интуитивными представлениями и мнениями экспертов о характере происходящих при этом процессов. Этот результат, как и то, что точность нейропрогнозирования не уступает лучшим результатам, полученным другими методами, позволяет утверждать, что построенная нейронной сетью модель паводков близка к той, которую способны создать специалисты при проведении соответствующих исследований. Поэтому особого внимания заслуживают те результаты нейропрогнозирования, которые для специалистов оказались неожиданными. Прежде всего, необъяснимой остается причина того, что учет дополнительных данных по бассейну р. Латорица привел к значительному улучшению точности прогнозирования максимумов паводка в долине р. Уж, но практически не повлиял на качество прогноза при низкой воде. Загадочна также причина довольно высокого рейтинга показателя температуры воздуха Tvb- (0), поскольку в обозреваемом периоде причиной высоких паводков были осадки, а не таяние снегов, вызванное повышением температуры воздуха. Пока неясно, насколько стабильны эти аномалии и для их подтверждения необходимы дальнейшие наблюдения. Если хотя бы одна из них подтвердится, то можно будет говорить об открытии нейронной сетью нового важного природного явления, исследование которого потребует участия человека.

Библиография

[1] Галушкин А.И. Нейрокомпьютеры / кн 3,-М.: ИПРЖР,-2000.- 528с.
[2] Комарцова Л.Г., Максимов А.В. Нейрокомпьютеры / -М. Из-во МГТУ им. Баумана, -2002. – 320с.
[3] Сычев А. С. Селекция связей в нейронных сетях с псевдоинверсным алгоритмом обучения // Математичиские машины и системы № 2 1998. с.25-30
[4] Boger Z. Who is afraid Big Bad ANN? // The 2002 IEEE World Congress on Computational Intelligence, Honolulu May 12-17 2002. p.2000—2004
[5] L.M. Belue, K.W. Bauer Determining input features for multilayer perceptrons // Neurocomputing 7, 1995, - p.111-121.
[6] Verikas A, Bacauskiene M., Halmqvist K. Selection Features for Neural Network Committees // The 2002 IEEE World Congress on Computational Intelligence, Honolulu May 12-17 2002. p.215-221
[7] Різник О.М., Железняк М.Й., Новицький Д.О., Кужель К.М., Дончиць Г.В. Використання штучної нейронної мережі для короткотермінового прогнозування повеней. МНС // Праці Міжнародної конференції з індуктивного моделювання “МКІМ – 2002” Державний НДІ інформаційної інфраструктури, Львів 20-25 травня. том 2 - С. 96-102
[8] Сигеру Омату, Марзуки Халид, Рубия Юсуф Нейроуправление и его приложения / - М.: ИПРЖР, 2001. - 324с.
[9] Резник А.М.,. Куссуль М.Э., Сычов А.С., Садовая Е.Г, Калина Е.А. Система проектирования модульных нейронных сетей САПР МНС. //– Труды 8-й Всероссийской конференции «Нейрокомпьютеры и их применение», Москва, 21-22 Марта 2002.
[10] Reed R.D. and. Marks R.J, Neural Smithing. / MIT Press. - 1999. - 346с

Информация об авторах

А.М. Резник - Институт проблем математических машин и систем НАН Украины, пр. Академика Глушкова, 42, Киев 187, 03187, Украина, e-mail: neuro@immsp.kiev.ua

К.М. Кужель - Институт проблем математических машин и систем НАН Украины, пр. Академика Глушкова, 42, Киев 187, 03187, Украина, e-mail: neuro@immsp.kiev.ua

МОДЕЛИРОВАНИЕ ОСНОВНЫХ ПСИХОЛОГИЧЕСКИХ ФУНКЦИЙ

А.Шевченко, В.Ященко

Abstract*: In work the opportunity of modeling of the basic psychological functions is considered. For modeling psychological functions the new class of neural networks – neural-like growing networks is used. The definitions of concepts " artificial intelligence ", " artificial subconsciousness " and " artificial consciousness " are submitted.*

Keywords: *Neural-like networks, artificial intelligence, artificial subconsciousness, artificial consciousness.*

Вступление

В настоящее время большинство ученых, занимающихся проблемой разработки интеллектуальных вычислительных систем и роботов, понимают, что эти средства должны обладать развитым интеллектом и возможностью общения с человеком при помощи органов чувств.

В связи с этим возникает проблема создания вычислительных систем и роботов новой генерации с нетрадиционной архитектурой, позволяющей осуществлять распознавание и обработку символьной информации, выполнять анализ и классификацию, делать логический вывод и другие операции, характерные для систем искусственного интеллекта, так же эффективно, как и вычислительные операции.

В Институте искусственного интеллекта в результате проведенных исследований теоретически доказана возможность создания новой формации машин (машинокомпьютеров, роботокомпьютеров, полуроботов и пр.), которые, аналогично человеку, способны к восприятию и переработке разнообразной информации и самостоятельному выполнению механических действий.

В институте проблем математических машин и систем разработана новая технология обработки информации - новый класс нейронных сетей (нейроподобные растущие сети). Доклады о новом классе нейронных сетей опубликованы в трудах международных конференций [1-4]. Новая технология апробирована и получила широкое применение в Институте искусственного интеллекта в системах распознавания образов и интеллектуальных роботах.

На Первой международной научно-практической конференции «Искусственный интеллект – 2000», которая проходила с 11 по 16 сентября 2000 года в п. Кацивели, Крым, Украина, усилиями многих ученых под эгидой и непосредственном участии Института искусственного интеллекта принята концепция формирования искусственного интеллекта на основе создания искусственного сознания. В качестве рабочих сформированы определения понятий интеллекта и искусственного интеллекта.

«*Интеллект*» - совокупность универсальных процедур, которая позволяет на сознательном уровне строить конкретные алгоритмы решения частных творческих задач. Созданные алгоритмы определяют интеллект.

Таким образом, интеллект проявляется как реализованный алгоритм решения задач, сформированный сознанием. При этом «*естественное сознание*» определяется как высшая форма деятельности живого организма, владеющая определенным количеством информации о себе и своем окружении, способная получать информацию, формировать знания, определять смысл и цель своего существования.

«*Искусственное сознание*» - высшая управляющая система машины, владеющая определенными знаниями о себе и своем окружении, способная получать информацию, формировать знания, определять свое предназначение в соответствии с целью и задачами, поставленными создателем или пользователем.

«*Искусственный интеллект*» - реализованный алгоритм решения задач, сформированный искусственным сознанием.

Вопросы возможности обладания технических систем сознанием обсуждаются учеными давно. В 50-60-х годах Г.Фрэнк, Бэнсе и др. относили сознание к теме кибернетических исследований [5-7].

К Штейнбух в работе [8] высказывает следующее предположение: «Каждое субъективное переживание соответствует определенному физически описываемому состоянию организма и прежде всего состоянию нервной системы, а также отчасти гуморальной системы и других органов.

Неважно, что в данный момент в большинстве случаев еще не известны закономерности взаимосвязи между содержанием сознания и физической ситуацией. И то, что вследствие действия принципа неопределенности физическая ситуация не может быть описана с любой точностью, здесь также не столь важно. Ведь, во-первых, процессы, протекающие в нервной системе, подчиняются в целом законам макрофизики, а во-вторых, рассматриваемые стохастические явления могут быть описаны методами кибернетики. Из сделанного выше предположения неизбежно вытекает, что искусственно созданные технические системы могут обладать сознанием».

В 1957г. модельную трактовку основных психологических понятий (*мышление, мысль, сознание, подсознание*) дал Н. Амосов на основе разработанной им гипотезы о сетевом разуме с системой усиления-торможения (СУТ) [9]. Эти понятия выглядят следующим образом:

1. *Мышление* - взаимодействие моделей, направляемое чувствами и СУТ.

2. *Мысль* - модель, усиленная СУТ в данный момент.

3. *Сознание* – движение активности по значимым моделям, усиленным СУТ, отражающим важнейшие отношения в системе субъект – среда.

4. *Подсознание* – взаимодействие моделей ослабленных СУТ. Оно обеспечивает подготовку моделей для сознания, распознавание заученных образов, и выполнение привычных действий.

Н. Амосов рассматривал два противоположных подхода к моделированию психологических функций и интеллекта в целом. Эти подходы были названы сетевым и алгоритмическим. Соответственно он различал и два типа моделей – сетевой и алгоритмический интеллекты.

Рассмотрим возможность моделирования основных психологических функций (*мышление, мысль, сознание, подсознание*) на нейроподобных растущих сетях, т.е. возможность моделирования сетевого интеллекта.

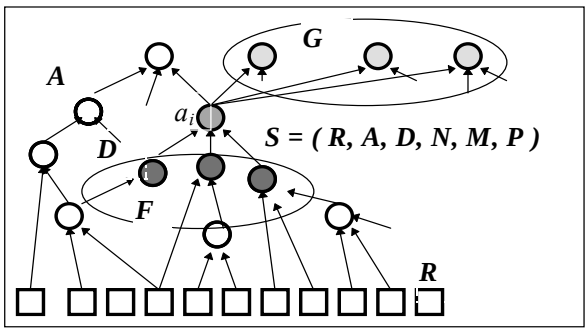


Рис.1. Нейроподобная растущая сеть

Нейроподобные растущие сети являются динамической структурой, которая способна воспринимать информацию, формировать и накапливать знания, вырабатывать управляющие воздействия в соответствии с целью и задачами. Структура сети изменяется в зависимости от значения и времени поступления информации на рецепторы, а также предыдущего состояния сети. В ней информация об объектах представляется ансамблями возбужденных вершин и связями между ними. Запоминание описаний объектов и ситуаций сопровождается вводом в сеть новых вершин и дуг при переходе, какой либо группы рецепторов и ранее образованных вершин в состояние возбуждения. Процесс возбуждения волнообразно распространяется по сети. Эти сети представляют собой иерархическую структуру, позволяющую запоминать и выделять общие признаки изучаемых понятий, формировать многоуровневую структуру сети в соответствии с описываемыми знаниями о знаниях предметной области.

Нейроподобные растущие сети

Формально одномерные нейроподобные растущие сети задаются шестеркой: $S = \{ R, A, D, P, N, M \}$, где $R = \{r_1 , r_2 , r_3 ,..., r_n \}$ -- конечное множество рецепторов которые составляют порождающее множество сети; $A = \{a_1 , a_2 , a_3 ,..., a_k \}$ -- конечное множество концепторов, которые соответствуют сочетаниям признаков, определяющих конъюктивные связи обьектов;

$D = \{d_1 ,d_2 ,d_3 ,...,d_s \}$ -- конечное множество дуг связывающих рецепторы с концепторами и концепторы между собой; M - множество весовых коэффициентов дуг; N - конечное множество переменных коэффициентов связности; Переменный коэффициент связности, определяет минимально допустимое число дуг заходящих на концептор и позволяет варьировать числом дуг приходящих на концепторы и числом концепторов в сети. Так при $N = k$ минимальное число дуг приходящих на формируемый концептор равно k и если k достаточно велико то в сети увеличивается число дуг приходящих на концептор и уменьшается число концепторов соответствующих пересечениям описаний объектов или ситуаций проблемной области. Если k настолько велико, что в сети не выделяются концепторы соответствующие этим пересечениям, то образуется однослойная растущая сеть. $P = \{p_1 ,p_2 ,p_3 ,...,p_g \}$ - конечное множество порогов возбуждения вершин множества A, $P_i = f(m_i) > P^0$, P^0 -- минимально допустимый порог возбуждения при условии, что множеству дуг D' приходящих на вершину a_i соответствует множество весовых коэффициентов $M'=\{m_1 , m_2 , m_3 , ... , m_w \}$, причем m_i может принимать как положительные, так и отрицательные значения.

Вышеперечисленные множества определяют структуру сети. Такое представление удобно для формального описания, но не обладает наглядностью, поэтому нейроподобные растущие сети также представляются ориентированным графом (рис.1), в котором вершины не имеющие заходящих дуг, называются рецепторами, остальные концепторами.

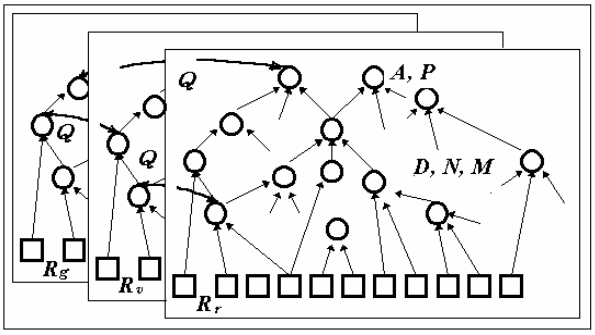


Рис.2. Многомерная нейроподобная растущая сеть

Многомерные нейроподобные растущие сети (рис.2) представляются следующим образом: $S = \{ R, A, D, P, N, M, D_i\}$, где $R \supset R_l , R_r , ... , R_z$, $A \supset A_l , A_r ,..., A_z$, $D \supset D_l , D_r ,..., D_z$, $N \supset N_l , N_r,..., N_z$, $M \supset M_l, M_r ,..., M_z$, $R_l = \{r_1 ,r_2 ,r_3 ,...,r_l \}$; $R_r = \{r_1 ,r_2 ,r_3 ,...,r_r \}$; $R_z = \{r_1 ,r_2 ,r_3 ,...,r_z\}$ конечные подмножества рецепторов принадлежащих различным информационным пространствам, например, лингвистическому, речевому и зрительному; $A_l = \{a_1 ,a_2 ,a_3 ,...,a_l \}$; $A_r = \{a_1 ,a_2 ,a_3 ,...,a_r \}$; $A_z = \{a_1 ,a_2 ,a_3 ,...,a_z \}$ -- конечные подмножества концепторов, принадлежащих различным информационным пространствам, отражающих понятия или объекты, например, в лингвистической, речевой и зрительной областях ; $D_l = \{d_1, d_2, d_3, ..., d_l \}$; $D_r = \{d_1 ,d_2 ,d_3 ,...,d_r \}$; $D_z=\{d_1 ,d_2 ,d_3 ,...,d_z \}$ — конечное подмножество дуг, принадлежащих различным информационным пространствам, отражающих связи между понятиями или объектами, например, в лингвистической, речевой и зрительной областях; D_l, D_r, D_z — конечное подмножество дуг, связывающих концепторы подмножеств A_l , A_r , A_z. $P_l = \{p_1 ,p_2 ,p_3 ,...,p_l \}$ --конечное множество порогов возбуждения вершин множества A_l ; $P_r = \{p_1 , p_2 , p_3,...,p_r \}$ --конечное множество порогов возбуждения вершин множества A_r; $P_z = \{p_1, p_2, p_3 ,...,p_z \}$ --конечное множество порогов возбуждения вершин множества A_z ; M - конечное множество весовых коэффициентов дуг; N - конечное множество переменных коэффициентов связности. D_i - конечное множество дуг, связывающих концепторы различных информационных пространств.

Уже на приведенных н-РС могут моделироваться простые виды из указанных классов психофизиологических функций. Однако для моделирования процесса мышления, как осознаваемого взаимодействия с внешней средой целесообразно использовать разработанные в этих целях рецепторно-эффекторные нейроподобные растущие сети (рэн-РС) [10,11].

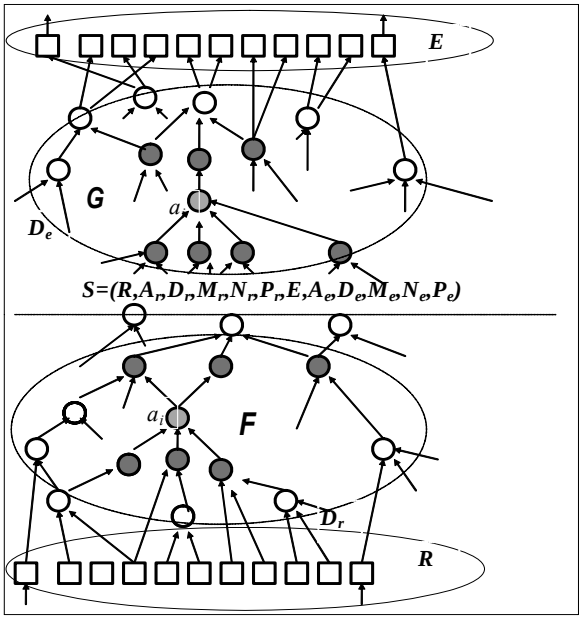


Рис.3. Рецепторно - эффекторная нейроподобная растущая сеть

В рэн-РС выделяются рецепторные ***R*** и эффекторные ***E*** поля, а также рецепторные ***F*** и эффекторные ***G*** зоны (рис.3). Сети рэн-РС формально задаются следующим образом:

S = (R, A_r , D_r , P_r , M_r , N_r , D_i^r , E, A_e , D_e , P_e , M_e , N_e, D_i^e), где ***R*** $=\{ r_i \}$, $i=1, n,$- конечное множество рецепторов, ***A_r*** $=\{ a_i^r \}$, $i = 1, k,$ - конечное множество концепторов рецепторной зоны, ***D_r*** $=\{ d_i^r \}$, $i = 1, e,$ - конечное множество дуг рецепторной зоны, ***M_r*** - конечное множество весовых коэффициентов дуг рецепторной зоны; ***N_r*** - конечное множество переменных коэффициентов связности рецепторной зоны; ***D_i^r*** - конечное множество дуг, связывающих концепторы различных информационных пространств рецепторной зоны; ***P_r*** $=\{ P_i^r \}$, $i = 1, k$, где P_i^r - порог возбуждения вершины a_i^r, $P_i^r = f(m_i^r) > P_i^0$ (P_i^0 - минимально допустимый порог возбуждения) при условии, что множеству дуг ***D_r*** , приходящих на вершину a_i^r , соответствует множество весовых коэффициентов ***M_r*** $=\{m_i^r\}$, $i = 1,w,$ причем m_i^r может принимать как положительные, так и отрицательные значения.

E $=\{ e_i \}, i = 1, f,$ - конечное множество эффекторов, ***A_e*** $=\{ a_i^e \}$, $i = 1, k$, - конечное множество концепторов эффекторной зоны, ***D_e*** $=\{ d_i^e \}$, $i = 1, e,$ - конечное множество дуг эффекторной зоны, ***M_e*** - конечное множество весовых коэффициентов дуг рецепторной зоны; ***N_e*** - конечное множество переменных коэффициентов связности рецепторной зоны; ***D_i^e*** - конечное множество дуг, связывающих концепторы различных информационных пространств эффекторной зоны. ***P_e*** $=\{ P_i^e \}$, $i = 1, k,$ где P_i^e - порог возбуждения вершины a_i^e , $P_i^e = f(m_i^e) > P_i^0$ (P_i^0 - минимально допустимый порог возбуждения) при условии, что множеству дуг ***D_e*** , приходящих на вершину a_i^e , соответствует множество весовых коэффициентов ***M_e*** $=\{m_i^e\}$, $i = 1,w,$ причем m_i^e может принимать как положительные, так и отрицательные значения.

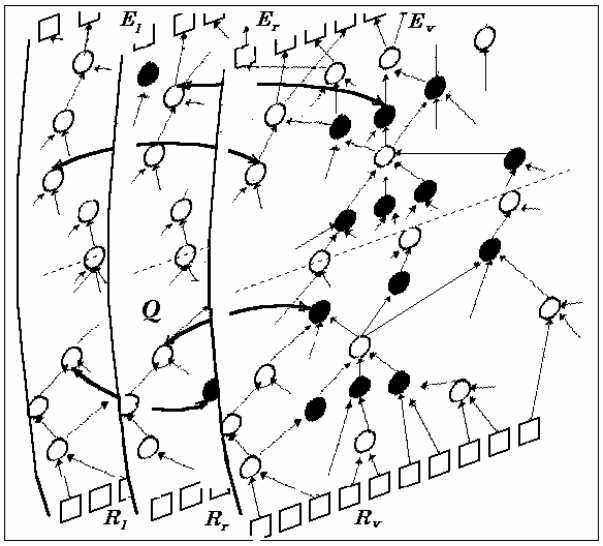


Рис.4. Многомерные рецепторно-эффекторные нейроподобные растущие сети

Многомерные рецепторно-эффекторные нейроподобные растущие сети (мрэн-РС) представляется графом (рис.4) и формально задаются следующим образом:

S = (R, A_r, D_r, P_r, M_r , N_r, E, A_e, D_e, P_e, M_e, N_e); $R \supset R_v$, R_s, R_t ; $A_r \supset A_v$, A_s, A_t ; $D_r \supset D_v$, D_s, D_t ; $P_r \supset P_v$, P_s , P_t ; $M_r \supset M_v$, M_s, M_t ; $N_r \supset N_v$, N_s, N_t ; $E \supset E_r$, E_d, E_d,; $A_e \supset A_r$, A_{d1}, A_{d2} ; $D_e \supset D_r$, D_{d1}, D_{d2} ; $P_e \supset P_r$, P_{d1}, P_{d2} ; $M_e \supset M_r$, M_{d1}, M_{d2} ; $N_e \supset N_r$, N_{d1}, N_{d2} , здесь ***R_v , R_s, R_t,*** - конечное подмножество рецепторов визуальной, слуховой и других (например, тактильных, вкусовых) областей,

A_v, A_s, A_t - конечное подмножество нейроподобных элементов; ***D_v, D_s, D_t*** - конечное подмножество дуг; ***P_v, P_s, P_t*** - конечное множество порогов возбуждения нейроподобных элементов рецепторной зоны, принадлежащих, например, визуальному, слуховому, тактильному информационным пространствам; ***N_r*** - конечное множество переменных коэффициентов связности рецепторной зоны; ***E_r, E_{d1}, E_{d2}*** - конечное подмножество эффекторов речевой области и различных областей действий; ***A_r, A_{d1}, A_{d2}*** - конечное

подмножество нейроподобных элементов; D_r, D_{d1}, D_{d2} - конечное подмножество дуг эффекторной зоны; P_r, P_{d1}, P_{d2} - конечное множество порогов возбуждения нейроподобных элементов эффекторной зоны, принадлежащих, например, речевому информационному пространству и пространству действий; N_e - конечное множество переменных коэффициентов связности эффекторной зоны.

Моделирование основных психологических функций на многомерных рецепторно-эффекторных нейроподобных растущих сетях

В структуре рэн-РС выделяются области безусловных и условных рефлексов, а также области накопления знаний и мотиваций. Однако такое деление структуры рэн-РС чисто условно, так как элементы указанных областей распределены в случайном порядке по всей сети. Безусловные рефлексы относят к врожденным, а условные - к приобретенным.

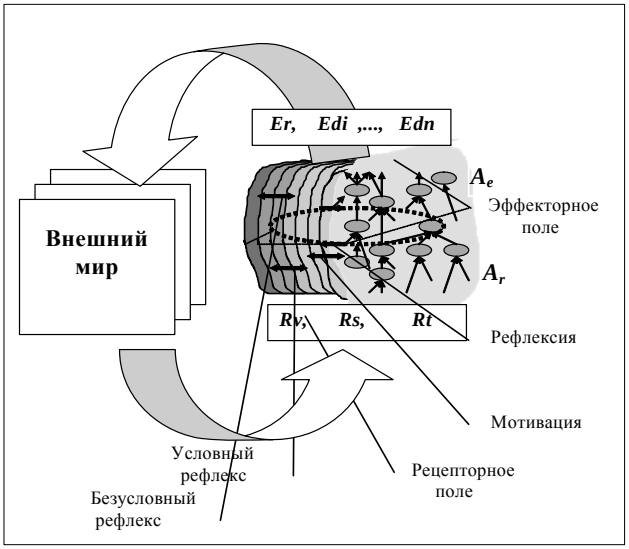


Рис.5. Условная схема моделирования психологических функций

В нашем случае моделирования основных психологических функций (рис.5) к безусловным рефлексам мы относим ту часть рэн-РС, которая формируется при создании системы например для обеспечения ее внутренних потребностей, таких как поддержание оптимального напряжения питания, включение резервного источника питания в случае отключения основного, контроль работоспособности отдельных узлов и их сочетаний, а также потребности ее развития, например, непрерывное поступление информации (любознательность), возможность общения с подобными системами, человеком и др. К условным рефлексам отнесем ту часть сети, которая формируется в процессе обучения. Хотя это разделение довольно условно так как, например, при обучении вождению автомобиля на начальных этапах обучения стажер выполняет операции управления осознанно, обдумывая каждое движение и тем самым, вырабатывая условные рефлексы управления. Но опытный водитель управляет им не задумываясь, выполняя операции управления автоматически, на уровне безусловных рефлексов. Значит, выработанные условные рефлексы управления автомобилем превратились в безусловные. В рэн-РС во время обучения (формирование весовых коэффициентов связей и порогов возбуждения вершин сети) формируется область условных рефлексов, а на уровне умения (установления, фиксации коэффициентов связей и порогов возбуждения) эта область переходит в область безусловных рефлексов.

Восприятие - информация, поступающая из внешнего мира на рецепторное поле и далее, в рецепторной зоне в соответствии с правилами формирования сети запоминается, анализируется, классифицируется, обобщается и при каждом новом поступлении усиливается (подтверждается) или ослабляется (увеличивая или ослабляя весовые коэффициенты связей и изменяя пороги возбуждения узлов) и таким образом накапливается, осуществляя ее фильтрацию в соответствии с принципом ассоциативного восприятия, выражающегося в том, что классифицируется только та информация, которая сочетается с предыдущей информацией, запомненной в сети.

Действие - в соответствии с информацией (условием), формируемой в рецепторной зоне, в эффекторной зоне (в соответствии с правилами, описанными выше) целевая ситуация анализируется, классифицируется, обобщается, усиливается, если цель достигается или ослабляется - в противном случае, вырабатывая в эффекторном поле сигналы управления органами воздействия на внешний мир.

Рефлексия - информация циркулирует в замкнутом контуре рецепторной и эффекторной зон.

Неосознанная реакция - внешняя информация через область безусловных рефлексов воздействует на внешний мир.

Осознанная реакция - внешняя информация через область условных рефлексов и знаний воздействует на внешний мир.

Заключение

Таким образом, в соответствии с определениями основных психологических понятий, предложенных Н.Амосовым, в структуре рэн-РС эти определения можно сформулировать следующим образом:

1. *Мышление* - взаимодействие возбужденных ансамблей нейроподобных элементов, направляемое ансамблями возбужденных нейроподобных элементов, соответствующих мотивационных областей.
2. *Мысль* - ансамбль нейроподобных элементов, возбужденных в данный момент.
3. *Сознание* - движение активности по ансамблям нейроподобных элементов, направляемое нейроподобными элементами, мотивационных областей, отражающее важнейшие отношения в системе субъект - среда.
4. *Подсознание* - взаимодействие ансамблей нейроподобных элементов, обеспечивающих поиск целевых ситуаций, не передавая возбуждение в эффекторную зону. Оно обеспечивает подготовку моделей для сознания, распознавание заученных образов и выполнение привычных движений.

В связи с этим, определение понятия искусственный интеллект для сетевой модели интеллекта можно сформулировать следующим образом: «*Искусственный интеллект*» определяется как реализованный алгоритм решения задач, сформированный искусственными сознанием и подсознанием. Под «*искусственным сознанием*» понимается движение активности (направляемое нейроподобными элементами, мотивационных областей, отражающее важнейшие отношения в системе субъект – среда) по сети нейроподобных элементов, в которой накоплены определенные знания о себе и окружающем мире, которая способна воспринимать информацию, формировать и накапливать знания, вырабатывать управляющие воздействия в соответствии с целью и задачами, поставленными создателем или пользователем. Под «*искусственным подсознанием*» понимается взаимодействие ансамблей нейроподобных элементов, обеспечивающих поиск целевых ситуаций, не передавая возбуждение в эффекторную зону, которое обеспечивает подготовку моделей для искусственного сознания, распознавание заученных образов и выполнение привычных движений.

Литература

[1] Yashchenko V. Neural-like growing networks - new class of the neural networks. Proceedings of the International Conference on Neural Networks and Brain Proceeding, pages 455 -458, Beijing, China, Oct.27-30'98

[2] Yashchenko V. Neural growing network in solving problem of computerisation of natural languages. First international workshop computerisation of natural languages, sept. 3-7, 1999, Varna, St. Konstantin, Bulgaria, pp. 129-136

[3] Yashchenko V. Receptor-effector neural-like growing network. YIII Международная конференция KDS-99 Знания-Диалог-Решения, Сентябрь 13-18, 1999, Крым, Кацевели, С. 144-152

[4] Yashchenko V. Receptor-effector neural-like growing network - an efficient tool for building intelligence systems. Proceedings of the second internatiional conference on information fusion, July 6-8, 1999, Sunnyvale Hilton Inn, Sunnyvale, California, USA, Vol.II, pp. 1113-1118.

[5] Frank, H.: Kybernetische Grundlagen der Padagogik. Baden-Baden:AgisVerlag 1962.

[6] Bense, M.: Bewuβtseinstheorie. Grundlagenstudien 2 (1961) H. 3, S. 65-73.

[7] Gunther, G.: Das Bewuβtsein der Maschinen. Krefeld und Baden-Baden:AgisVerlag 1957.

[8] Steinbuch, K.: Automat und mensch. Kybernetische Tatsachen und Hypothesen. Berlin: Springer – Verlag 1965.

[9] Амосов, Н.: Алгоритмы разума. К.: Наукова думка 1979.

[10] Ященко В.А. Рецепторно-эффекторные нейроподобные растущие сети эффективное средство моделирования интеллекта.I // Кибернетика и систем. анализ. - 1995. - 4. - С. 3-8.

[11] Ященко В.А. Рецепторно-эффекторные нейроподобные растущие сети эффективное средство моделирования интеллекта.II // Кибернетика и систем. анализ. - 1995. - 5. - С. 14-18.

Сведения об авторах

Шевченко А.И. - Донецкий государственный институт искусственного интеллекта. E-mail: aishe@iai.donetsk.ua

Ященко В.А. - Донецкий государственный институт искусственного интеллекта. E-mail: mis@immsp.kiev.ua

АСПЕКТЫ БИОТЕХНИЧЕСКИХ ИССЛЕДОВАНИЙ ЗРИТЕЛЬНОГО ВОСПРИЯТИЯ ЧЕЛОВЕКОМ

Шульга Е.Ю.

Abstract: *In given clause the questions of use of resources of biotechnical systems are considered at research of process of touch perception by the man of the visual verbal information.*

Keywords: *biotechnical system, visual verbal stimulus, visual alphabet, handicap (flare).*

Введение

Современные требования синтеза различных систем искусственного интеллекта предопределяют необходимость исследований психофизических реакций человека на различные виды воспринимаемой им сенсорной информации. Реализация этих особенностей в технических системах с различным уровнем интеллектуализации требует методики, инвариантной как относительно комплекса технических средств, обеспечивающих взаимосвязь объекта управления с внешней средой, так и относительно специфики самого объекта управления. То есть, биотехнические системы призваны обеспечивать наработку параметров для настройки интеллектуальной системы на особенности информационных потоков, поступающих от измерительных систем и алгоритмов восполнения информации о характеристиках объекта управления. Согласно вышеизложенному биотехнические системы - это комплекс, позволяющий синтезировать адаптивные алгоритмы для систем с неполной информацией об объекте управления. Создание таких комплексов, в первую очередь, нуждается в разработке методических положений, опирающихся на особенности восприятия окружающего мира человеком.

В отличие от современных ЭВМ, где операции обработки сенсорной информации выполняются последовательно, зрительная система, очевидно, способна выполнять эти операций независимо и параллельно. Деление зрительной системы на две подсистемы, которые находятся в левом и правом полушариях, способствует осуществлению параллельных процессов.

Структурно *биотехническая система* (БТС) включает в себя биологический объект исследования или управления, человека-исследователя и ЭВМ, которые объединены единым алгоритмом функционирования для максимального достижения поставленной задачи в данной ситуации.

Исходя из этого определения особенностей, рассмотрим ключевые свойства БТС:

- БТС формируется итоговой целью ее функционирования, то есть цель функционирования БТС является ее системно- образующим фактором.
- Структура БТС зависит не только от цели, но и от метода, положенного в основу ее функционирования, причем метод всегда опирается на определенные свойства объекта исследования и подчиненный итоговой цели;
- БТС обладает гомеостатическими свойствами, так как в ее состав всегда входит биологический объект [1].

В свою очередь, асимметрия функций больших полушарий мозга общепризнанна, однако, конкретные принципы, которые обуславливают не равноценность полушарий, продолжают интенсивно обсуждаться [1,2].

В частности, представляют собой интерес такие аспекты проблемы, как:

- сравнительная характеристика работы левого и правого полушарий мозга человека в зависимости *от типа* зрительной информации при *наличии помехи.*
- изучение физиологической значимости *механизмов* межполушарного взаимодействия в процессе анализа предложенных сигналов (символов) для понимания принципов обработки информации на уровне целого мозга.

Исследование последних двух десятилетий продемонстрировали расхождения специализации полушарий в зависимости от *вербальных/невербальных* характеристик предлагаемых символов, их эмоциональности, а также от способов переработки информации. Большинство исследователей [1,2] показывают, что левое полушарие, конечно, имеет перевес при восприятии вербального материала (букв, слов, изображений, которые легко вербализируются). Такой результат объясняли тем, что вербальная информация в этом случае имеет более прямой доступ к специализированным языковым центрам, расположенным в левом полушарии (Kimura,1961).

Правое полушарие лучшее (более быстрее, точнее) обрабатывает невербальный материал, который тяжело вербализируется. Также оно лучше левого узнает незнакомые лица и преобладает при решении ряда задач, связанных с оценкой пространственных свойств символа, оценке ориентации, лиц, восприятии глубины.

Имеющиеся методики исследований, которые используются на данный момент касаются, в основном, выявления отдельных проявлений функционирования зрительного анализатора и зрительной коры, не имеют универсализм, который разрешал бы установить единый механизм, который лежит в основе динамики процесса распознавания. В ряде принципов, выдвинутых и обоснованных в данных методиках, подчеркивается детерминированный характер функционирования зрительного анализатора.

В связи с этим, методологическая ценность экспериментального изучения процесса распознавания с помощью применения ресурсов биотехнической системы исследования, в конечном итоге состоит в возможности выявления и изучения свойств и принципов функционирование структурных элементов ЦНС разного уровня иерархии. Способ экспериментального исследования ЦНС иерархического уровня предусматривает измерение адекватной исходной реакции, в частности, мозга при определенного рода влияниях на него через зрительный анализатор.

Результаты

Экспериментальные данные, полученные в ходе верификации данной БТС, составили информационное поле первичных знаний, осознание которых через логико-эвристическое, информационно– структурное и математическое моделирования функционирования центральной нервной системы (ЦНС) позволило обнаружить некоторые общие для всех уровней и специфические для процесса распознавания принципы.

Наши исследования [3,4], во-первых, подтвердили, что при предъявлении символа сначала правое полушарие с помощью дедуктивного метода (от общего к частному, от синтеза к анализу) быстро первично оценивает ситуацию, опираясь на твердые элементы системы парного мозга и их врожденные механизмы. Потом левое полушарие, на основе индуктивного метода (от частного к общего, от анализа к синтезу) уже вторично формирует представление об общей закономерности и разрабатывает соответствующую стратегию обращения, используя не только твердые, но и гибкие элементы системы в ходе обучения. Также выявили, что время, необходимое для распознавания изображения, зависит от информационного содержания предложенных изображений и осуществляется выбором зрительных образов в определенном алфавите, то есть, определяется всей совокупностью изображений, ожидаемых наблюдателем в данной ситуации.

В виде зрительного алфавита в нашем исследовании были выбраны отдельные буквы и трех буквенные слова русского алфавита. Комплекс технических средств БТС предъявлял их группе испытуемых в правое и левое поля сетчатки глаза в нормальных условиях видения и под влиянием вспышки. Обнаружили интересный факт. Он состоял в том, что в нормальных условиях видения, как и предполагалось (исходя из известных фактов) процесс распознавания букв и слов шел лучше в правом поле зрения сетчатки глаза. То есть в левом полушарии. Ожидали, что под влиянием помехи (впышки) процесс распознавания ухудшится (увеличится общее время распознавания и уменьшится число правильных ответов. Но как показали экспериментальные данные под влиянием вспышки, несмотря на действительно увеличившееся время распознавания, число правильных ответов увеличилось.

Таким образом, мы сделали вывод о том, что вспышка выступала не только как помеха, но и как некий стимулирующий фактор для процесса зрительного опознания. Все эти факты тем самым говорят нам о необходимости проведения дальнейших исследований в данном направлении.

В свою очередь, получение этих закономерностей не стало бы возможным без использования соответствующего функционального программного обеспечения синтезированной нами биотехнической системы. Ее внешний вид приведен на рис. 1

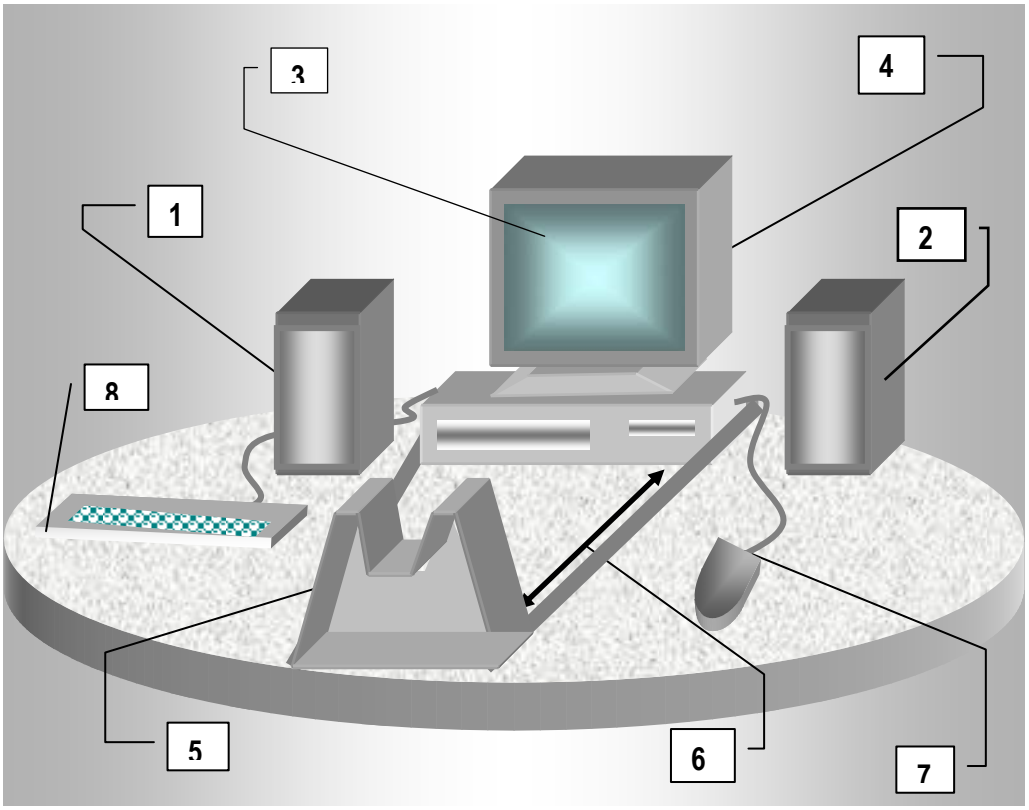


Рис. 3.1. Внешний вид экспериментальной установки БТС в опыте по звуковому опознанию сигналов.

Обозначения на рисунке:

1. Левый звуковой источник (левая акустическая система);
2. Правый звуковой источник (правая акустическая система);
3. Вспышка (помеха) на экране монитора БТС (x_3, табл. 2.3);
4. Монитор;
5. Подголовник экспериментальной установки;
6. Расстояние до источника звукового сигнала (x_5, табл. 2.3) ;
7. «Мышь» ЭВМ;
8. Клавиатура ЭВМ (фиксация решений ЛПР

Заключение

1) На основе анализа современных исследований выделена проблематика биотехнических систем и ее место при синтезе и функционировании интеллектуальной системы.
2) Согласно проблематики, сформирована структура биотехнической системы, функционально предназначенной для формирования информационных массивов, моделей реакций, и их взаимосвязях с фиксацией данных в исследуемых и синтезируемых моделях в разделах базы данных.
3) Определены особенности функционирования БТС в составе интеллектуальных систем в предпосылках режимов обучения- адаптации интеллектуальных систем через информационную обработку реакций испытуемых.

4) Методологические аспекты, выделенные в структуре БТС, позволяют ставить и решать задачи статистического анализа и моделирования психических реакций человека на сенсорную информацию, представленную зрительными образами, с формированием информационных массивов и синтезом математических моделей в БТС, как начального приближения оценивания состояний интеллектуальной системы.

Литература

1. Кроль В.М. Специфика работы зрительных механизмов правого и левого полушарий мозга человека //Ж. Высш. нервн. деят.-1995.- Т.45, №6.- С.1075-1083.
2. Ващук Ф.Г. Введение в проблему информационно-структурного моделирования процесса исследования и системы формирования психосоциальной сферы.- Ужгород, 1995.-107 с.
3. Шульга Е.Ю. Апробация работы автоматизированной системы «Flash-Words», при исследовании зрительного опознания в связи с асимметрией мозга //Искусственный интеллект» № 3, 2001, стр. 382-386.
4. Шульга Е.Ю. Основні логіко-формальні моделі, отримані у ході верифікації біотехнічної системи дослідження процесу впізнання зорових вербальних стимулів // Математичні системи і машини», м. Київ №1, 2002, стр. 123.

Информация об авторе

Шульга Екатерина Юрьевна - ассистент кафедры математики Донецкого государственного института искусственного интеллекта, e-mail: seu@iai.dn.ua

ON NEURON MECHANISMS USED TO RESOLVE MENTAL PROBLEMS OF IDENTIFICATION AND LEARNING IN SENSORIUM

G.S. Voronkov, Z.L. Rabinovich

Abstract: *The paper considers some possible neuron mechanisms that do not contradict biological data. They are represented in terms of the notion of an elementary sensorium discussed in the previous authors' works. Such mechanisms resolve problems of two large classes: when identification mechanisms are used and when sensory learning mechanisms are applied along with identification.*

Keywords: *sensory learning, identification, task solution, elementary sensorium.*

Introduction

The main component of thinking quite probably consists in the problem (task) resolution. The question of how to simulate the brain function of thinking becomes more and more urgent.

The paper represents some viewpoints related to arrangement of neural mechanisms implementing two large mental problem classes, viz. how the mental identification and learning tasks are solved.

The above viewpoints follow from the general description of sensory systems and of the sensorium in all. This general description is based on the model paradigm [1]. According to this paradigm, the sensorium is the neuron model of the familiar sensory environment. The model paradigm notion is actually the projects of the ideas, mostly known and taken from various spheres of knowledge about the brain and information processes, onto the unique hierarchical neural network. This network comprises the elementary sensory system [2], the memory medium [4] and the intelligent medium [4]. It is proposed as the generalized result obtained when data about

structural and functional arrangement of sensory systems are analyzed [2, 5]. The paper collects and develops the fragments of the model paradigm notion, related to thinking.

Elementary sensorium: some basic notions

1.The long-term memory (LTM) is the very model from the neurons (the very sensorium; see the right part of Fig. 1), it reflects the potential familiar sensory environment (about unfamiliar environment, see below, # # 1 and 2). In the typical structure (TS; see the left part of Fig. 1**),** the complex familiar potential object is submitted as a whole by symbolic neuron and its elementary properties – by the receptors and quasi-symbolic neurons. In the sensorium, the complex familiar potential object is submitted as a whole by the neuron on the upper most synaptic level, here the different modal key properties of this object are submitted too**.** At the lower levels, its complex subproperties and their traits are submitted. The most elementary indecomposable properties are submitted in the very first sensory systems TS of the sensorium.

Being influencing, stimulating, the object is submitted in the model by the same neurons, but already in the excited state. The characteristics specific only to the stimulating object (force, duration, importance) are submitted in functional parameters of neurons (force and duration of excitation), the memory of them is kept for some time as a changed excitability of the formerly activated neurons and the changed conductivity of the ways to them, in other words – in the changed synaptic weight. It is **the short-term memory - STM**. Its mechanisms are rather investigated neuro-physiological mechanisms of the plasticity.

The presentation of **unfamiliar object** activates the neurons of low synaptic levels, only that of the neurons matching the familiar properties and subproperties of this unfamiliar object. They are remembered for some time in STM with the help of mechanisms described above. As it is possible to see, the storing in STM is not accompanied by formation of the new connections. The storing in LTM is the arrangement of the symbolic neurone to match the new object (as a whole), the new symbolic neuron and formation of its TS connections.

The mutual positive connections inside TS (between symbolic and quasi-symbolic neurons) provide rhythmization of activity, its quantumization, amplification and contrasting. The latter is provided by the lateral inhibition characteristic for biological neuro-networks.

The rhythmic process, in the separate TS, develops, lasts for some time and attenuates. The attenuation occurs due to the collecting recurrent inhibition. All these processes occur in the olfactory bulb, TS prototype, and the bulb computer model [6]. The descending connections in the elementary sensorium provide synchronization of rhythmic activity of appropriate TSs. The activity of the uppermost TS plays a leading role in this.

Task and solution: general notions

2. The task here is a complex object (situation, picture, phenomenon) of the sensory environment, shown to the elementary sensorium for identification; the solution is the identification of this object. The absence of identification is absence of solution. **The familiar object** can be unidentified due to several circumstances: or due to it is insufficiently distinct among the environmental objects, or it is shown insufficiently full (fragmentary), or due to combination of these circumstances. We also believe, that any presentation (actualization) of the environment is "presentation with the purpose of identification" since identification is the basic function of the sensory systems. The identification process is initiated by influence of adequate stimulus on the receptors, then the process is directed by the network architecture, the neuron connections, their present excitability state and others.

The stated question can also be a part of the task condition (if the task is also shown in the verbal form), it is the important component of the task condition, in fact, the major "help" making the object distinct in the background.

Note, that the examined elementary sensorium (Fig. 1) is isolated from motivation, emotional and other systems of the brain. So, by its example, we actually analyze only the own possibilities of this simple circuit (in terms of the tasks solution) and only occasionally we take account of its communications with other brain structures. In this analysis we imagine that this circuit works by the known neuro-physiologic mechanisms. This analysis is limited by the use of only biological mechanisms and model paradigm framework**.**

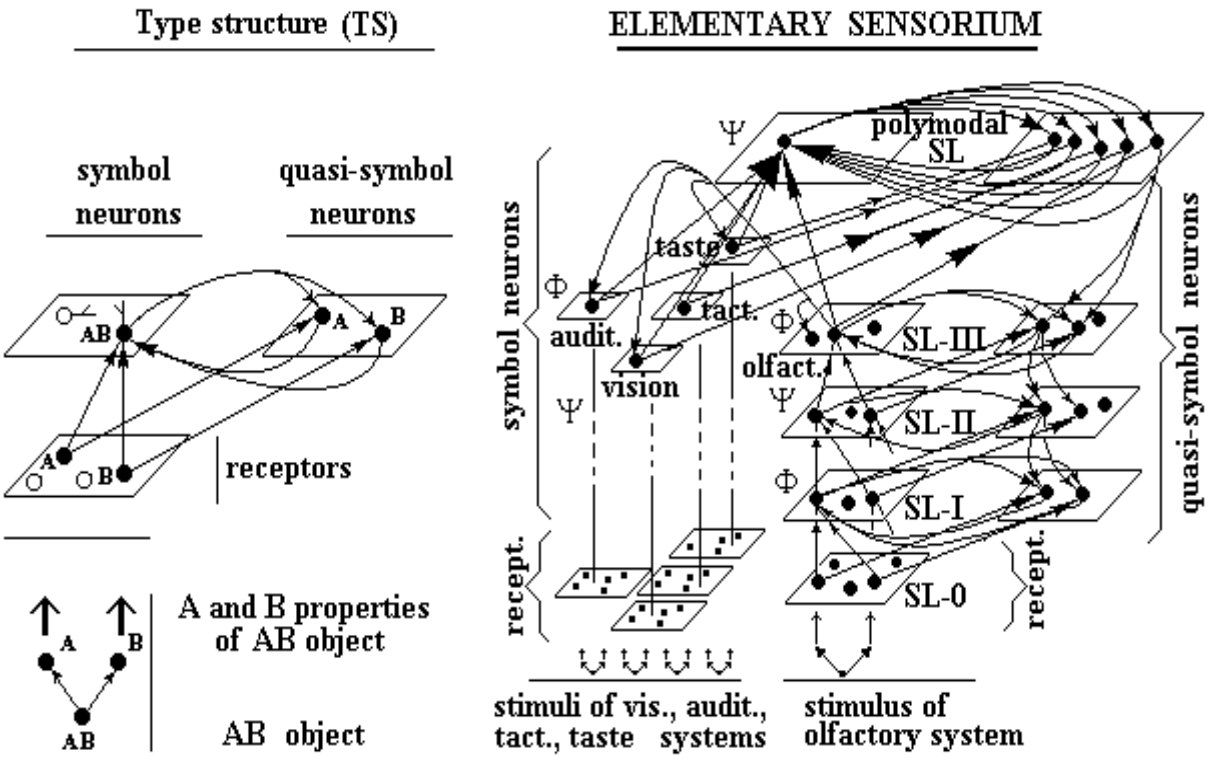


Fig. 1.

The shown **unfamiliar (new) object** cannot be identified in principle, since it is not submitted in the sensorium as a whole. Identification of this object in the sensorium is possible only after formation of its neuron model (see below, # 6).

This understanding of the "task" and "task solution" is followed by classification of the tasks, being the large share of all mental tasks in all, to two classes. The first class involves the tasks, solved by the identification mechanisms only. This is the class of "identification tasks ". The second class involves the tasks, solved, by the mechanisms forming new "concepts" in addition to the identification mechanisms. The concept-forming mechanisms are similar to or are the sensory learning mechanisms. It is the class of "learning tasks". Note here, that the identification and sensory learning differ from the comparative process, the latter is not considered in the work.

3. The presented identification task, if it is not solvable at once, is submitted in the brain by the activated sensorium part. This part consists of the activated neurons, matching the background objects and their properties including the task object and its properties. However, the latter (viz. the neurons matching the task object) do not reach the condition of activity, appropriate to identification (see below # 7). Thus activated part of the sensorium is **the initial situation**.

To the solution of identification task there corresponds **the final situation** (identification state). It is a steady, for some time, special dynamic activity state of neurons matching the task object only. These activated neurons represent simultaneously both the task object as a whole and its major properties. The feature of the dynamic state consists in the fact that the activity of these neurons is amplified and rhythmical, and their rhythms are synchronized (see below, # 8). In the period, when the final situation takes place, the neuron rhythmic activity dominates the activity of others neurons of the elementary sensorium. The neuron domination is expressed in priority influence of their activity on the effector and other brain systems. In the stated representation, the domination state is correlated with the phenomenon called by the standard term "attention".

4. The process of transition from the initial situation to the final situation is **the process of task solution**.

By definition, the identification task assumes, that the final situation neurons are submitted in the sensorium a priori; they are the part of the neuron model (long-term memory), representing the potential environment in the brain. Thus, the process of the solution consists in selective activation of the final situation neurons. In addition to

"the matching realization mechanism" in the direct sensory pathway, during selective activation, other elementary sensorium mechanisms (see below**,** # # 7, 8) act; the set of the selective activation mechanisms is possible to designate as **mechanisms of identification**.

The moment, when the final situation is reached, correlates, according to the stated representation, to the subjective emotional sensation, insight (eureka). Here, it is possible to note, that, since the problem of sensation remains a principle “blank space" in the brain problem, this subjective display of biological sensorium work, used here in interpretation of the elementary sensorium work, is only formal.

If the task is not solved (i.e. the final situation is not reached, the insight is not shown), for continuation of the solution process, it is necessary to expand the task conditions by entering the additional fragments or different-sort helps making object distinct in the background.

5. There may be the following identification tasks, for instance. 1) A game for children. It is necessary here to look for and identify a known contour within a web of lines (branches). Task statements are: a represented picture proper (e.g. branches of a bush); a question asked in some form; other prompts. The solution is vision and identification of a desired (known) contour. 2) Examination school tasks. The task solution means here to identify a task type (scheme) since the solution of each task type (description or model in the most generalized form at upper levels) is already present in the sensorium (or the linguistic system) of a person being examined on the basis of a passed program. 3) Tasks solved by analogy. As in the case of school tasks, gist of solution consists in identification of the general scheme characterizing both an initial situation and another but the known one, i.e, it is necessary to identify a scheme already represented in the sensorium. 4) In essence, solution syllogisms (i.e. deduction process) also possibly comes to identification.

It is obvious, that being similar in principal, the task solution processes in the given examples 1) -4) should differ among themselves with TS activation sequence.

6. For the learning task, the final situation cannot, in principal, be reached with the help of only the identification mechanisms since the absence of the uppermost (uniting) TS and other TSs, which potentially would represent the task object (and its properties), does not allow the solution process to reach the identification state (see # 2).

To solve this task it is necessary to add the missing TSs into the initial situation. This can be carried out only by formation of new TSs. At the first synaptic level, the new TS formation is initiated by influence of a new stimulus of on the receptors (sensory learning). To initiate TS formation at a high level, it is necessary to activate TS symbolic neurons of the previous level. The latter can proceed in two ways: through the sensory learning and/or through the influence of activating brain systems on the sensorium symbolic neuron fields. Evidently, the neuron model formation in the second way largely proceeds from the "internal resources" of the sensorium, from LTM. Note here, that formation of TS symbolic neurons in the elementary sensorium has some formal similarity with formation of new tops in pyramidal networks [7].

The model formation proper (new TSs formation)) can be considered as a really creative process since it singles out a new object (or property) in sensory environment (or in its neuron model) and models this object (viz., matches the new TS symbolic neuron with the new object). In linguistic terms, the singling-out of the new object in the environment (and the formation of the new "object" on the basis of knowledge) is formation (birth) of the new concept.

Neuron mechanisms of identification and sensory learning in the elementary sensorium

7. In the elementary sensorium (Fig. 1), matching the potential sensory environment, the initial situation is represented by the activated neurons of typical structures (TS), paired with the presented task (see # 3).

Thus if any of the key stimulus appears presented," the matching realization mechanism" of the sensory direct pathway selectively activates the symbolic neuron of the uppermost TS (the neuron input is organized as disjunction). This neuron matches the task object as a whole.

It activates the matching quasi-symbolic neurons. The latter match the single key properties of the task object. By descending projections, these quasi-symbolic neurons activate the lower TS quasi-symbolic neurons matching the subproperties of the key properties.

The neuron activation of all lower TSs can be facilitated by the activities of the ascending pathways. By associative (signification) projections, the TS symbolic neurons of the initial situation sub-activate (or activate) the

symbolic neurons of other TSs.These neurons play the role of the context. As a result, the quantity of the activated neurons (and TSs) involved increases. The quantity of the involved TSs determines the depth and detail of identification.

8. Every TS is rhythmically active because of an arrangement of neuron bonds in TSs (Fig. 1, [2], [5]). *Initial situation* TSs are not yet united by the uppermost TS and, therefore, their operation is autonomous to a significant extent. Their rhythms may also be not synchronized at this moment. If the uppermost TS is activated, it synchronizes an operation of every final situation TS according to the above bonds. The synchronization in neuron rhythmic activity of all TSs is reached for some time period. And here is the "completed pyramid" making up and corresponds the final situation, viz. task solution. This period of the most intensive neuron rhythmic activity of the completed pyramid runs along with the highest neuron inhibition of all the neighbouring TSs not matching the final situation. The above activity domination is thus provided**.** The above inhibition occurs due to lateral inhibition bonds. This period runs for some time. It is limited by a recurrent summed inhibition accumulated inside the TS (see # 3). Therefore, the *task solution* is the synchronized rhythmic neuron activity of the complete TSs pyramid, stable for some time. The synchronized rhythmic activity is, so to say, the dynamic attractor where the rhythmic activity of the pyramid vertex TS acts as the order parameter. According to the above idea, this very state is correlated with sensation of identification (understanding) of the complete picture as whole and simultaneously with modal perception (vision, hearing) of the picture details.

9. What is a possible mechanism of new TSs generation (their symbol and quasi-symbol neurons, bonds between these neurons and their bonds with other TSs) when the learning task is being solved? One can make the assumptions close to the truth so far, proceeding, in particular, from the following. 1). The data, derived from the sensory learning in ontogenesis, show that these mechanisms really exist. This can be exemplified by generation of detectors of vertical lines for kittens grown up in the environment with vertical lines and the absence of the detectors when kittens are grown up in the "horizontal" environment (Hubel, Wisel, 1962). 2). So called stem cells are shown in nervous system. They are generated into new neurons. 3). The constant "searching" growth of the neuron processes is illustrated. The growth is terminated when a contact with a target has been established. 4). Some hypotheses about the targeted growth mechanisms (e.g., it may be the chemical affinity principle) and about the selective bond formation principles (the Hebb principle) are advanced. 5). A host of neuron mechanisms have been studied that provide the selectivity of neurons and plasticity of the neuron inputs. 6). The targeted axon projections, including convergence, on neurons of sequence synapse levels is shown.

10. Consider the hypothetical elementary process of generation of the very first sensorium TS (Fig**.** 1). A set of various-type receptors is formed according to the genetic program. The sensory learning for an object supposes a reiterated activation of a certain set of different-type receptors. This activation initiates the growth of axons to the next level neurons. The first contact of any axon out of a group of growing axons with one of the neurons makes this neuron a target for other simultaneously growing axons due to the Hebb principle, for instance. This is a possible way of the symbol neuron generation. The symbol neuron corresponds to a joint array of certain properties when a neuron input is generated according to conjunction, or to a class of certain properties when the input is generated according to disjunction. Possibly, the targeted 1:1-projection of receptors onto quasi-symbol neurons is provided by the chemical affinity factor. The generation of positive mutual communications between a symbol neuron and corresponding quasi-symbol neurons may likewise be based either on chemical affinity or on the Hebb principle. Supposedly, the latter is also the basis for generation of significative communications between any symbol neuron and other particular symbol neurons. The same principles may also provide the generation of descending bonds of quasi-symbol neurons. Since the symbol neuron fields act as receptor fields for TSs of the next synaptic levels, the above process can provide the generation of the second-, third- and higher-level TSs. The TSs generation process is initiated by activation stimuli: the sensory stimuli occur at sensory learning, and, the stimuli from activation brain systems occur at learning based on remembering. The process is directed by genetically provided growth, the target selection mechanisms and by the network architecture.

It is clear that the new TS generation in the neuron model, proceeding from only inner sensorium resources**,** calls for additional mechanisms to involved, in particular, the short memory mechanisms for identified fragments of the initial situation as well as their periodic recurrence (withdrawal from the memory). Likewise, the tasks may differ by the significance of the initial situation fragments: the higher the level of TSs, being the fragments of the initial situation, and the more fragments are involved, the greater is the possibility for the sensorium to solve the problem. This argument is true for the problems of both classes.

It is evident from the above description that the learning problem solution process needs more energy and more time than the solution of identification problems.

Conclusion: the role of tutor and language system in identification and sensory learning

11. The described way, where the concepts are derived in the elementary sensorium, is possibly basic providing the emergence of the concepts in fauna evolution and in the creative human thinking. Likewise, the sensorium concepts in ontogenesis in animals are generated more efficiently, namely, via learning by tutor. In fact, the tutor-involving method consists of the sensory learning supplemented with the important prompt into the initial situation. The prompt helps single out the object as a whole from the background. When higher animals bring up their progeny, the behaviour of mature animals acts as the prompts. Humans also have this element, but prompts are implemented largely verbally through the language system associated with the sensorium [2,8]. The language system matures in ontogenesis faster than the sensorium. Thus, in addition to the genetic factor, the volume of the formed sensorium (LTM) is incomparably larger than that in animals; the human sensorium is actually of another quality. It may well be, the role of the language system was as much important in development of the sensorium (genetically fixed) due to their communication function in the unusually rapid evolution of humans.

Likewise, there occurred the evolution of the language system proper. In addition to the communication and simulation functions, the intelligent function (task solution) was also developed in it. There is a ground to assume that the above description fragments of thinking neuron mechanism in the sensorium are also applicable for the language system.

In addition, the interaction between the sensorium and the language system is a new factor in thinking, including the creative one, and adds new additional mechanisms there. This makes the qualitative level of thinking even higher.

It is evident that, the working-up of the aspects, in addition to the whole number of other ones not considered in this paper can make an essential contribution into the development of the model paradigm notion about thinking mechanisms.

Bibliography

[1] Voronkov G.S. Brain and Information. Problems of Neuro-Cybernetics - Proc. Jubilee International Conference on Neuro-Cybernetics, Rostov-on-Don, 2002, vol. 1, p. 15-18. (In Russian); http://www.biolog.ru/vnd

[2] Voronkov G.S. Sensory System - A neuron Semiotic Model of an Adequate Medium. In: Comparative Physiology of a Higher Nervous Activity of Men and Animals. Moscow, Nauka, 1990, p. 9-21. (In Russian)

[3] Voronkov G.S., Rabinovich Z.L. A Natural Medium of Memory and Thinking and Its Model Representation. KDS-2001 - Knowledge-Dialog-Solution: Proc. Int. Scientific and Practical Conference, St. Petersburg, 2001, vol. 1, p. 110-115. (In Russian)

[4] Voronkov G.S., Chechkin A.V. Neuron Semiotic Systems as Intelligent Media. KII-96 - Artificial Intelligence - 96: Proc. 5th National Conference with International Participation, Kazan, 1996, vol. 1, p. 26-35. (In Russian)

[5] Voronkov G.S. Analyzing Principles of Coding in Sensory Systems // Biological Sciences, Dep. VINITI, No. 2144 - V88, 1998. 39p. (In Russian)

[6] Voronkov G.S., Izotov V.A. A computer Model for Neuron Mechanisms of Information Processing at the First Synaptic Level of Olfactory System // Intelligent Systems, 1998, vol. 3, No. 1-2, p. 87-108 (In Russian)

[7] Gladun V.P. Partnership with Computer. Kiev, Port-Royal, 2000, 119p. (In Russian)

[8] Voronkov G.S., Rabinovich Z.L. Sensory and Language Systems as two Forms of Knowledge Representation // News of Artificial Intelligence, 1993, No. 2, p. 116-124. (In Russian).

Author information

Voronkov Gennadi Sergeevich – The M.V. Lomonosov Moscow State University, Leninskie Gory, Moscow, 119899, Russia; e-mail: gsv@gvoronkov.home.bio.msu.ru

Rabinovich Zinovi Lvovich – V.M. Glushkov Institute of Cybernetics, Prospect Acad. Glushkova, 40

03680, Kiev-187, Ukraine; e-mail: eco@public.icyb.kiev.ua

ИСКУССТВЕННАЯ МУДРОСТЬ

Загоруйко Н.Г.

***Аннотация**. Мудрость является составной частью разума и проявляется в умении выдвигать цели, ставить проблемы и формулировать задачи. Рассматривается возможность построения программ искусственной мудрости, имитирующих человеческую способность к выбору целей.*

1. Введение

В соответствие с теорией целеустремленных систем [1] свойства индивидуума определяются его откликами на различные стимулы или воздействия. Содержание отклика зависит от трех составляющих: набора рефлексов и навыков, доведенных до автоматизма, базы знаний, позволяющих сознательно управлять выбором действий, и системы целевых установок и ценностей. Эта онтологическая схема хорошо согласуется с представлениями русской классической философии [2] о разуме, как о единстве трех начал: мудрости, ума и воли. Мудрость – это способность выбирать цели, формировать систему ценностей. Ум или интеллект – основанная на знаниях способность строить эффективные планы достижения выдвинутых целей. Воля – способность организовать деятельность, направленную на выполнение плана достижения выдвинутых целей. Разумным считается такое поведение, в котором гармонично проявляются все эти три способности.

При разработке искусственного разума этап создания искусственной воли состоит в организации процессов выполнения программ, реализующих заданный алгоритм. Достижения этого направления широко известны. Продолжаются интенсивные исследования архитектуры вычислительных машин, систем и сред, влияющих на этот процесс, совершенствуются методы программирования.

Направление искусственного интеллекта, занятое разработкой алгоритмов (планов) решения творческих задач, еще не достигло таких признанных результатов, как предыдущее, но на его развитии сосредоточены большие усилия многочисленных коллективов исследователей.

Что же касается систем искусственной мудрости, то целенаправленная задача их создания, фактически, еще не ставилась. «Машина может решить любую задачу, но никогда не сможет поставить ни одной из них». Это высказывание А. Эйнштейна, сделанное на заре развития вычислительной техники, многими считается бесспорным и до сих пор. Между тем, философские и психологические исследования в области гносеологии постоянно затрагивают проблему возникновения и развития разума и такой его уникальной и определяющей характеристики, как целенаправленность. Кроме того, многие работы в области искусственного интеллекта затрагивают проблему целеполагания, структуру иерархии «суперцель-цель-подцель». Так что, нельзя сказать, что мы ни сколько не продвинулись в вопросе понимания, что такое цель и в чем состоит процесс выбора цели или постановки задачи.

Ниже приводятся описание одной из возможных систем взглядов на эту проблему.

2. Цель

Состояние любой системы определяется текущими значениями (X0) ее внутренних характеристик и характеристик влияющего на нее окружения. Цель представляет собой описание желательного для системы состояния в заданный будущий момент или промежуток времени (Xt). Для внешнего наблюдателя истинная цель системы всегда остается неизвестной. Он может лишь строить модели поведения системы с разными вариантами целей и выбирать ту модель (и следовательно, ту цель), с которой наилучшим образом согласуются наблюдаемые действия системы. Строго говоря, даже собственные цели оказываются иногда результатом самообмана. Однако в дальнейшем мы с учетом этих оговорок будем рассматривать идеализированную ситуацию, при которой цель системы известна и однозначно определяется желательным состоянием ее характеристик Xt.

В качестве наблюдаемой системы будем рассматривать некоторую популяцию живых особей, поведение которых определяется индивидуальными целями. Относительно происхождения этих целей можно исходить из двух гипотез: а) цель возникает стихийно, развивается и закрепляется механизмом естественного отбора; б) цель в завершенном виде задается внешними силами.

Особи зарождаются своими родителями, наследуя некоторую комбинацию их свойств, живут определенный промежуток времени, порождают какое-то количество потомков и отмирают. Гипотезу стихийного происхождения целей можно обосновать, если ориентироваться на такую наблюдаемую характеристику, как продолжительность жизни особи и популяции в целом. Случайное сочетание свойств некоторой особи может оказаться благоприятным для сохранения ее жизни в условиях постоянно существующих разрушающих воздействий внешней среды. Особи с другим сочетанием свойств могут оказаться хуже приспособленными к сопротивлению энтропийным процессам и скорее исчезнут из популяции. В результате такого естественного отбора в популяции станут преобладать особи, свойства которых лучше приспособлены к условиям существования. Они обладают более высоким уровнем жизненного потенциала: дольше живут и порождают больше потомков, чем их менее удачливые соседи.

Внешний наблюдатель вполне обоснованно придет к заключению, что особи данной популяции ведут себя так, как если бы они сознательно преследовали цель самосохранения и повышения уровня своего жизненного потенциала.

Рассмотрим гипотезу задания цели внешними силами. Цели могут быть самыми разными, но главной их характеристикой будет их направленность в системе координат «самосохранение» - «саморазрушение». Особи, заданная цель которых ориентирована на саморазрушение, быстро достигнут эту цель и прекратят свое существование. Выживут, и будут развиваться те особи, цель которых направлена на сохранение и повышение уровня жизненного потенциала.

Так что происхождение цели не играет существенной роли. Задается ли она мудростью высших сил в готовом виде, или возникает и закрепляется стихийно, цель особи в живой популяции состоит в повышении уровня жизненного потенциала.

Очевидно, что цель не обязательно должна явно формулироваться и осознаваться ее носителями. Не требуется даже, чтобы она ощущалась ими инстинктивно. Достаточно того, чтобы действовал механизм обратной связи: чтобы действия особи в согласии с этой целью повышали вероятность самосохранения, а отклонение от цели отрицательно сказывалось на этой вероятности. Тогда общее направление развития популяции будет тем же, как и в случае, когда все особи осознают свои цели. Однако скорость развития таких популяций будет различной.

3. Иерархия целей

Если особи обладают способностями осознавать свои цели, то их движение к цели может совершаться не методом случайного блуждания, а более экономично, по плану, который разрабатывает их интеллект. Этот план может содержать ряд промежуточных этапов, которые являются подцелями заданной главной цели (задача разбивается на подзадачи). По поводу каждой подцели можно задать вопрос «Для чего это делается?», и содержательный ответ на него будет состоять в том, что достижение данной подцели служит необходимым условием достижения более высокой цели, вплоть до главной. Но тот же вопрос правомочен и по отношению к этой главной цели: «А для чего делается это?». В конце концов, цепочка таких рассуждений приводит нас к главному вопросу: «Для чего делается все то, что делает система?». Применительно к живым системам этот вопрос эквивалентен вечному вопросу о смысле жизни.

Из предыдущего следует естественный вывод: главная цель (суперцель) большинства особей состоит в повышении уровня своего жизненного потенциала. «Главная цель жизни – жить!» - говорил Л.Н. Толстой.

Перейдем теперь от суперцели особи к суперцели популяции. Мы пришли к заключению, что в живой популяции доля особей, преследующих цель самосохранения, будет всегда больше доли особей, ориентированных на самоуничтожение. Суммарный вектор целей всех особей будет направлен на сохранение большинства особей, и, следовательно, популяции в целом.

Однако такой механически получаемый вектор не является результатом индивидуальных усилий, специально направленных на самосохранение популяции, и потому не может считаться суперцелью ни каждого индивидуума, ни популяции.

На каждую особь постоянно действует разрушающее сопротивление среды. Если условия жизни особи становятся более трудными (возрастает воздействие внешней среды, возникает дефицит жизненного ресурса вследствие роста численности популяции и т.д.), то в поисках способов выживания особи могут случайно обнаружить отдельные проявления синергии. Оказывается, что если часть усилий, затрачиваемых на производство ресурсов, переориентировать с индивидуального производства на коллективное, то возрастает как суммарный, так и индивидуальный объем производимого ресурса. Продолжительность жизни особей, входящих в такие коллективы («семьи»), будет увеличиваться в большей степени, чем особей, продолжающих бороться за жизнь индивидуально. Каждый член коллектива таким способом повышает уровень своего жизненного потенциала. И, преследуя свою прежнюю цель, он теперь будет стремиться к сохранению не только себя, но и «семьи». Система отношений между особями усложняется, но эта более сложная организация способствует повышению жизнестойкости, как особей, так и семьи в целом.

Тем же путем может обнаружиться, что объединение усилий разных коллективов позволяет решать проблему жизнеобеспечения еще более эффективно. На этом этапе возникает сотрудничество в рамках более крупной части популяции («рода»). Теперь у особи есть цели трех уровней, согласованных между собой: цель самосохранения особи, семьи и рода. Действия по достижению этих целей, скоординированные в рамках более сложной организации, ведут к повышению уровня жизненного потенциала этих трех элементов популяции.

Дальнейшее вовлечение в кооперацию все большей части популяции завершается возникновением самой высокой цели – суперцели популяции. Она состоит в постоянном повышении уровня жизненного потенциала всех ее элементов и самой себя.

И здесь справедливо предположение о независимости результата от происхождения целей. Они могут возникнуть стихийно или быть привнесены в популяцию извне. В том и другом случае в популяции приживется только цель, ориентированная на повышение уровня жизненного потенциала всех ее элементов, и, в первую очередь, элементов самого нижнего уровня – особей. Если какая-то иерархическая структура не способствует самосохранению особей, то у особей не будет стимула содержать такую ресурсно-затратную структуру. Иерархические общественные и организационные надстройки возникают и сохраняются только в том случае, если они приносят пользу особям.

4. Зарождение промежуточных целей.

Выше говорилось о том, что план достижения суперцели расчленяется на составные этапы – подцели. Каков механизм их формирования?

Чем выше иерархический уровень управляющей структуры, тем сложнее понять ее реальное влияние на благополучие особей. Время между началом некоторого действия и его результатом может быть сравнимым с продолжительностью жизни особи. По этой причине многие объективно положительные действия могут восприниматься индивидуумом как отрицательные («не популярные решения») и вызывать инстинктивные ответные действия, препятствующие достижению целей высокого уровня. Во избежание таких конфликтов в популяции должны иметься механизмы, помогающие особи понять, отвечают ли ее действия высшим, а, следовательно, и собственным целям или нет.

В общем случае должна существовать система реакций, которая поощряла бы особь за действия, согласованные с суперцелью, и наказывала бы в противном случае. Правильная система таких ориентиров будет поощрять инициативную деятельность, направленную на достижение целей популяции, и уменьшать деятельность противоположного характера. В идеале особь от положительной деятельности должна получать удовольствие, а от отрицательной – страдание. Именно эту мысль сформулировал Демокрит [3], отвечая на вопрос «В чем смысл жизни?»: «Цель всего живого – избегать страданий и стремиться к удовольствиям».

Многие авторы подчеркивают, что главным стимулом деятельной жизни являются отрицательные ощущения. Сытое животное может предаваться сну или играм, а голодное начинает искать или добывать пищу. Ощущение неблагополучия заставляет человека действовать с целью устранения причины этого неблагополучия. Отсюда можно сделать следующий вывод: цель (проблема, задача) возникает из

ощущения неблагополучия. Локальное неблагополучие порождает цель малого масштаба, глобальные неприятности заставляют ставить и решение глобальные проблемы.

Причины отрицательных ощущений и методы их устранения могут быть очевидными: вышла из строя нужная вещь - почини или купи новую; программа требует слишком большого машинного времени - оптимизируй ее наиболее трудоемкие части; бессонница – уплати налоги.

В менее очевидных ситуациях источником задачи может служить нарушение привычного хода событий, появление факта, не согласующегося с ранее принятой моделью наблюдаемого процесса. В этом случае возникает потребность понять причину такого явления, оценить ее влияние на достижение преследуемой цели и, если нужно, устранить эту причину.

Близкой к этой ситуации является ощущение нарушенной гармонии. Можно представить себе, что развитие некоторой сложной системы состоит из совершенствования ее отдельных частей. Если все части развиваются с приблизительно одинаковым успехом, то ситуация не вызывает беспокойства. Если же обнаруживается подсистема, развитие которой отстает от общего фронта движения и снижает качество системы в целом, то именно ее развитие становится целью особых усилий.

Если машинная программа, имитирующая процесс развития такой системы, содержит в описании суперцели требование гармоничности этого развития, получает информацию о текущем состоянии всех подсистем и имеет критерии для сравнения относительной успешности развития каждой части, то она способна обнаруживать локальные нарушения гармонии и выработать цель (ставить задачу), направленную на устранение этого неблагополучия.

Заметим, что программы такого рода уже существуют. К ним можно отнести системы автоматического управления сложным технологическим процессом. Требование гармоничности в суперцели системы представлено заданием поддерживать параметры процесса в определенных границах. Если параметры приближаются к границам допустимых значений, то система воспринимает это в качестве сигнала неблагополучия, ставит задачу устранить его и, решая эту задачу, вводит параметры в нужные пределы. Если же собственных управляющих ресурсов системе не хватает для устранения этих отклонений, то она сообщает об этом, т.е. ставит задачу перед обслуживающим персоналом.

Алгоритм ZET [4] предназначен для обнаружения грубых ошибок и заполнения пробелов в таблицах данных. В режиме обнаружения ошибок (редактирования таблиц) он имеет своей суперцелью обнаруживать в таблице элементы, которые не согласуются с закономерностями, характерными для этой таблицы. Алгоритм имеет средства для обнаружения скрытых в таблице закономерностей, а также средства предсказания того значения проверяемого элемента, которое имело бы место, если бы закономерность полностью выполнялась. Сравнение предсказанного элемента с фактически записанным в таблице позволяет обнаруживать отклонения от закономерных значений (локальные неблагополучия) и сообщать о них хозяину таблицы с формулировкой возникающих при этом задач: «Проверить данный элемент. Если в таблице указано верное значение, подтвердить его. Если оно ошибочно, ввести истинное значение. Если истинное значение не известно и требуется указать закономерное, то подтвердить приемлемость предсказанного значения».

Если суперцель состоит в постоянном улучшении базы данных, содержащей некомплектные или сильно зашумленные данные, то при поступлении каждой новой порции информации алгоритм ZET будет использовать ее для обнаружения ошибок и заполнения имеющихся пробелов. Оценивать близость к суперцели можно по сумме разностей между значениями элементов, записанными в таблицах, и значениями этих же элементов, предсказанными алгоритмом ZET.

Эти примеры показывают, что проблема автоматического выдвижения целей или постановки задач не является неразрешимой. Обычное возражение против этого утверждения сводятся к тому, что суперцель задает машине человек. Это верно, по отношению к машине человек выступает внешней силой, привносящей суперцель. Однако это ничем не отличается от условий, в которых и сам человек решает проблему постановки новых задач. Как было показано выше, все живые, в том числе мыслящие особи, имеют суперцель, которая родилась не по их инициативе, а в ходе эволюции популяции или кем-то сформулирована и внесена в популяцию в готовом виде. И особь действует в согласии с кем-то заданной суперцелью, самостоятельно выдвигая лишь цели, которые являются подцелями этой суперцели.

Можно заметить, что помимо суперцели в машинную программу человек привносит еще и многие элементы, необходимые для ее достижения. Это тоже верно. Но и человек при движении к суперцели

использует большой объем знаний и навыков, выработанных и переданных ему предшествующими поколениями. Другую часть полезных навыков человек приобретает в ходе своего жизненного опыта в режиме самообучения. Точно так же, интеллектуальная программа часть операций выполняет по строго заданной инструкции, а некоторые элементы ее поведения формируются в результате самообучения (строятся решающие функции по обучающей выборке, выбирается наиболее информативное подпространство характеристик, в котором затем принимаются решения и т.д.).

Наконец, сторонники гипотезы о появлении разума в ходе развития популяции говорят, что человек свои знания и навыки приобрел естественным эволюционным путем, а машина получает это сразу в готовом виде от человека. И это противопоставление справедливо. Но оно не имеет отношения к обсуждаемой проблеме. Нас не интересуют различия в происхождении двух систем – человека и машины. Для нас важно лишь то, что машина, располагая, как и человек, извне заданной суперцелью и некоторыми извне полученными знаниями, в процессе движения к суперцели может обнаруживать локальные неблагополучия и выдвигать промежуточные цели (ставить задачи), направленные на устранение этих неблагополучий.

5. Суперцель и устойчивое развитие ноосферы

Описывая эволюционный вариант формирования суперцели, мы исходили из предположения о том, что увеличение функциональных возможностей системы, в частности возможности ее самосохранения, неизбежно связаны с ростом сложности ее организации [5]. На всякую природную систему действуют разрушающие силы, причем, чем сложнее устроена система, тем выше риск ее разрушения. Если некая система оказывается в состоянии сохранять себя или даже усложнять свою структуру, это свидетельствует о том, что система имеет средства, которые могут противостоять естественным разрушительным процессам.

Такие средства реализуют функции типа "внешнее воздействие - реакция - адекватный отклик" и известны как механизм отражения или сознание, простейшие проявления которого присутствовали на самых ранних стадиях развития органического мира. Простейшие микроорганизмы противостоят энтропийным процессам с помощью высокопродуктивных способов размножения. В животном мире наблюдаются механизмы адаптации к изменениям среды обитания в виде пассивного гомеостатического приспособления к изменяющейся среде, рефлексов уклонения от угрозы и некоторых более сложных рефлексов. Для сохранения более высокого уровня организации материи возможности этих механизмов были бы недостаточными. Требовалось появление способности материи к активному и упреждающему противодействию энтропии.

Эта способность получила свое воплощение в Разуме, с помощью которого носители разумной жизни могут не только приспосабливаться к среде обитания, но и изменять ее в благоприятном для себя направлении. Разум должен развиваться одновременно с развитием (усложнением) жизненных систем. Если развитие Разума отстает от роста сложности системы, то начинают преобладать силы ее энтропийного разрушения.

С развитием Разума у индивидуумов и по мере их социализации начинают проявляться элементы "Коллективного Разума" в виде постановок общих целей, коллективно вырабатываемых планов их достижения и организации совместных действий, направленных на реализацию этих планов. В результате, изменения природы Земли и ближнего космоса, вызываемые деятельностью людей, стали по своим масштабам сравнимыми с изменениями чисто природного характера. Эта веха в истории Земли стала осознаваться в качестве перехода к образованию вслед за геосферой и биосферой новой сферы ее развития - ноосферы.

Так же, как для отдельного человека главным средством самосохранения является его Разум, так для человечества и среды его обитания в эпоху ноосферы главным средством самосохранения является Коллективный Разум. Влияние несовершенного нарождающегося Коллективного Разума может порождать многочисленные процессы в природных, производственных, социальных или духовных областях, одни из которых объективно ведут к росту, а другие к ослаблению жизненного потенциала человечества. Их суммарный результат может носить нестабильный характер и в каждый момент времени проявляться в качестве процесса деградации, стагнации или развития. Выделим ту часть ресурсов коллективного

Разума, которая порождает процессы развития. Этому объекту наиболее близко соответствует используемый в русской философии термин "Соборный Разум" [2].

Таким образом, важнейшим средством самосохранения и развития ноосферы является не просто та часть интеллектуального и биоэнергетического потенциала человечества, которая объединяется в Коллективный Разум, но та, которая образует Соборный Разум, ведущий к сохранению и развитию ноосферы. Следовательно, пути устойчивого развития ноосферы совпадают с путями устойчивого роста мощности Соборного Разума. В этом состоит суперцель современной цивилизации, которая с большим трудом начинает осознаваться человечеством [6 ,7].

6. Заключение

1. Вне зависимости от того, как возникает или кем извне привносится суперцель, она сохраняется в популяции только в том случае, если ориентирована на повышение уровня жизненного потенциала особей этой популяции.
2. Разум является одновременно средством и суперцелью сохранения и устойчивого развития ноосферы. Способность обнаруживать ситуации, препятствующие движению к суперцели и формулировать цели, направленные на преодоление этих препятствий - мудрость, является важнейшей характеристикой всякой разумной системы.
3. Цель порождается ощущением локального неблагополучия. Если обнаруживаются факты, нарушающие гармонию развития, ожидаемый ход событий или известную закономерность, и при этом оцениваются, как негативно влияющие на суперцель, то возникает потребность (цель, задача) устранить причины появления таких фактов.
4. Главная задача создания искусственной мудрости состоит в разработке методов автоматического обнаружения неблагоприятных ситуаций в потоке наблюдаемых событий, критериев для оценки степени влияния (важности) локального неблагополучия на достижение заданной суперцели, и формулировки требований к устранению обнаруженных неблагополучий с указанием их важности.
5. Искусственный разум будет служить мощным средством усиления и расширения возможностей естественного разума. Важным локальным неблагополучием на пути к достижению этой цели является слабая разработанность проблемы автоматической формулировки задач. Этим обосновывается большая актуальность исследований, направленных на создание искусственной мудрости.

Литература:

1. Акофф Р., Эмери Ф. О целеустремленных системах. – М.: Советское радио. 1974.
2. Федоров Н.Ф. Собрание сочинений в 4-х томах. – М.: Прогресс, 1995.
3. Материалисты древней Греции. М.: Мир, 1957.
4. Загоруйко Н.Г. Прикладные методы анализа данных и знаний. Изд. ИМ СО РАН, Новосибирск, 1999.
5. Загоруйко Н.Г. Исследование проблем, связанных с моделированием процессов устойчивого развития ноосферы.// В сб. "Искусственный интеллект и экспертные системы" (Вычислительные системы вып.160) Новосибирск,1997, с.3-17.
6. Вернадский В.И. Научная мысль как планетарное явление. М.: Наука, 1991.
7. Коптюг В.А. Конференция ООН по окружающей среде и развитию (Рио-де-Жанейро, июнь 1992 г.) Информационный обзор. – Новосибирск: наука СО РАН, 1992.

Сведения об авторе

Загоруйко Н.Г. - Институт Математики СО РАН, Новосибирск, Россия; e-mail: zag@math.nsc.ru

Section 2: Decision Making

МЕТОДОЛОГИЯ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА В МНОГОУРОВНЕВОМ УПРАВЛЕНИИ

А.В.Босик

Аннотация: *В данной статье представлена методология искусственного интеллекта в многоуровневой системе управления.*

Ключевые слова: *многоуровневое управление.*

Переход к рыночным отношениям кардинально меняет взгляд на организацию производства, создающую условия для наилучшего использования техники и людей в процессе производства и тем самым повышает его эффективность.

Рыночные отношения на первый план выдвигают новые цели производства, рассматривающие его как гибкий механизм, способный в любой момент перестроиться на изготовление других видов продукции при изменении спроса, как оптимальное производство, функционирующее с наименьшими затратами и выпускающее точно в срок высококачественную и конкурентоспособную продукцию.

Традиционные методы управления не обеспечивают требуемой эффективности формирования управления в условиях:

- недостаточности априорной информации о внешней среде функционирования;
- большого количества трудно учитываемых факторов нестационарности и субъективного их характера.

Объектом управления выступает предприятие. Предприятие представляет собой систему большой размерности.

Для принятия решений в системе большой размерности предлагается иерархическая многоуровневая система. Описание архитектуры системы включает в себя описание условий обработки сигналов, перемешивания данных системы и распознавание образов, экспертной системы. Система распознавания образов моделируется нейронными сетями в совокупности с нечетким контролем.

Управление предприятием, как и иным любым субъектом хозяйственной деятельности невозможно без наличия ресурса производственных мощностей, сырьевых запасов, рабочей силы. Эти ресурсы в многоуровневом управлении рассматриваются как ресурсы управления, в рамках которых можно менять характер объекта управления.

Многоуровневое управление предприятием структурно подчиняется иерархии подразделений, существующих на предприятии (рис.1).

Под управлением в общем случае понимается воздействие как на систему, так и на подсистемы для достижения общей цели и локальных целей для подсистем.

В приведенной схеме управляющее воздействие имеет наивысший приоритет на первом уровне, наименьший – на шестом. Таким образом, управляющее подчинение идет снизу-вверх, а контроль исполнения управляющего воздействия сверху-вниз (обратная связь).

Для формализации структуры многоуровневой системы введем следующие обозначения. Пусть $P = \{p_i\}$, $i = \overline{1..n_1}$ – множество предприятий, $PR = \{pr_j\}$, $j = \overline{1..n_2}$ – множество производств,

$C = \{c_k\}$, $k = \overline{1..n_3}$ – множество цехов, $PD = \{pd_l\}$, $l = \overline{1..n_4}$ – множество переделов, $U = \{u_t\}$, $t = \overline{1..n_5}$ – множество участков, $A = \{a_s\}$, $s = \overline{1..n_6}$ – множество агрегатов.

рис.1 Схема многоуровневой системы управления

Формализуем функции состояния объектов многоуровневой системы. Пусть $FA(a_s)$ – функция состояния агрегатов, которая определяется следующим образом:

$$FA(a_s) = \begin{cases} 0, \textit{если на агрегате произошла авария}, \\ 1, \textit{если агрегат находится в рабочем состоянии}, \\ 2, \textit{если агрегат находится в текущем ремонте}, \\ 3, \textit{если агрегат находится в резерве}. \end{cases} \quad (1)$$

Также введем функцию состояния работы участков

$$FU(u_t) = \begin{cases} 0, \textit{если на участке произошла авария}, \\ 1, \textit{если участок находится в рабочем состоянии}, \\ 2, \textit{если участок находится в текущем ремонте}, \\ 3, \textit{если участок находится в резерве}. \end{cases} \quad (2)$$

Состояние переделов представлено следующим образом:

$$FPD(pd_l) = \begin{cases} 0, \textit{если на переделе произошла авария}, \\ 1, \textit{если передел находится в рабочем состоянии}, \\ 2, \textit{если передел находится в текущем ремонте}, \\ 3, \textit{если передел находится в резерве}. \end{cases} \quad (3)$$

Функция состояния цехов имеет вид:

$$FC(c_k) = \begin{cases} 0, \textit{если в цехе произошла авария}, \\ 1, \textit{если цех находится в рабочем состоянии}, \\ 2, \textit{если цех находится в текущем ремонте}, \\ 3, \textit{если цех находится в резерве}. \end{cases} \quad (4)$$

Состояние производства характеризуется следующей функцией:

$$FPR(pr_j) = \begin{cases} 0, \textit{производство стоит}, \\ 1, \textit{производство работает}. \end{cases} \quad (5)$$

Функция состояния предприятия представлена следующей функцией:

$$FP(p_i) = \begin{cases} 0, \textit{предприятие не работает}, \\ 1, \textit{производство работает}. \end{cases} \quad (6)$$

Для формализации соответствия между предприятиями, производствами, цехами, переделами, участками, агрегатами введем отображения множеств.

Между предприятиями и производствами введем следующее соответствие: если p_i – предприятие, то $PRP(p_i)$ – множество производств данного предприятия и наоборот, если pr_j – производство, то $PPR(pr_j)$ – множество предприятий, включающих данное производство.

Между производствами и цехами также установим соответствие. Если pr_j – производство, то $CPR(pr_j)$ – множество цехов данного производства. Если c_k – цех, то $PRC(c_k)$ – множество производств, включающих данный цех.

Введем отображение множеств цехов и переделов. Если c_k – цех, то $PDC(c_k)$ – множество переделов данного цеха. Если pd_l – передел, то $CPD(pd_l)$ – множество цехов, включающих данный передел.

Между переделами и участками установим следующее соответствие. Если pd_l – передел, то $UPD(pd_l)$ – множество участков данного передела и наоборот, если u_t – участок, то $PDU(u_t)$ – множество переделов, включающих данный участок.

Введем отображение множеств участков и агрегатов. Если u_t – участок, то $AU(u_t)$ – множество агрегатов данного участка. Если a_s – агрегат, то $UA(a_s)$ – множество участков, содержащих данный агрегат.

Состояние системы, определяемое функциями состояния FA, FU, FPD, FC, FPR, FP, изменяется по мере необходимости, а именно: при возникновении и ликвидации аварийной ситуации или при проведении планового ремонта на агрегатах, участках, переделах, в цехах.

Таким образом, состояние системы может быть описано следующим набором признаков:

$$\{FP(p_i), FPR(pr_j), FC(c_k), FPD(pd_l), FU(u_t), FA(a_s)\}, \quad (7)$$

$$i = \overline{1..n_1}, j = \overline{1..n_2}, k = \overline{1..n_3}, l = \overline{1..n_4}, t = \overline{1..n_5}, s = \overline{1..n_6}$$

n_1 – количество предприятий, n_2 – количество производств, n_3 – количество цехов,

n_4 – количество переделов, n_5 – количество участков, n_6 – количество агрегатов.

При возникновении аварии на агрегате a_s значение признака $FA(a_s)$ в соответствии с (1) изменяется с 1 на 0. При проведении ремонтных работ на аварийном агрегате значение признака $FA(a_s)$ изменяется с 0 на 2. При завершении ремонтных работ агрегат переходит в класс резервных и соответствующий признак изменяется с 2 на 3. И, наконец, при включении агрегата в работу значение признака изменяется с 3 на 1.

Аналогичная ситуация происходит при работе участков, переделов, цехов.

Предприятие может находиться в одном из двух состояний: работает или нет. Аналогичная ситуация происходит в работе производства.

На сегодняшний момент существуют три вида заказов: а)госзаказ; б)заказы по договорам с традиционными потребителями; 3)маркетинговые (биржевые) заказы.

Совокупность заказов, формируемых на предприятии, образует портфель заказов, которые представляются в виде нечетких множеств. Портфель заказов формируется в целом и по подразделениям. На каждом уровне работа подразделения нечеткая из-за сложившихся условий.

Для формализации существующих видов заказа введем следующие обозначения. Пусть $X_1 = \{x_{1u}^{\alpha}\}$, $u = \overline{1..N_1}$ – множество госзаказов, $X_2 = \{x_{2v}^{\alpha}\}$, $v = \overline{1..N_2}$ – множество договорных заказов, $X_3 = \{x_{3w}^{\alpha}\}$, $w = \overline{1..N_3}$ – множество маркетинговых заказов.

Согласно существующей системе формирования портфеля заказов по каждому виду ассортимента общее количество продуктов по ассортименту x^{α} имеет вид (11).

$$x_1^{\alpha} = \sum_{i=1}^{N_1} x_{1i}^{\alpha} \qquad (8)$$

$$x_2^{\alpha} = \sum_{i=1}^{N_2} x_{2i}^{\alpha} \qquad (9)$$

$$x_3^{\alpha} = \sum_{i=1}^{N_3} x_{3i}^{\alpha} \qquad (10)$$

$$x^{\alpha} = \sum_{i=1}^{3} x_i^{\alpha} \qquad (11)$$

Использование алгебры нечетких множеств, булевой алгебры, теории систем управления позволяет синтезировать интеллектуальную систему

Основополагающей частью этой системы является динамическая детерминированная модель $R(x)$ прогноза и распознавания решений при управлении предприятием, представленная системой дифференциальных уравнений первого порядка.

Формально каждое уравнение системы может быть представлено в виде

$$\frac{d\bar{y}}{dx} = \bar{f}(x,t,\beta), \qquad (12)$$

где $\bar{y}$ – прогнозируемая величина:

- расходов, возникающих при осуществлении предприятием основного вида своей деятельности;
- объемов запасов производственного сырья;
- валовых расходов предприятия, связанных с покупкой сырья и основных средств;
- расходов по каждому цеху на предприятии;
- прибыли предприятия;

$f_i(x,t,\beta)$ – нелинейная правая часть уравнения, зависящая кроме всего прочего и от параметров модели, которые настраиваются в процессе идентификации.

В качестве критерия оценки эффективности выбранного портфеля заказов используется система коэффициентов (эффективность применимости той или иной стратегии, которая определяется скоростью, точностью и прибылью), вычисляемых на множестве $R\{(y,\mu_r(y))\}$, полученном по модели $R(x)$ [Кофман, 1982]. Критерий сравнения в каждом отдельном случае выбирается по-разному.

На рис.2 представлена обобщенная схема управления на каждом подразделении. В качестве примера рассмотрен нижний уровень иерархии.

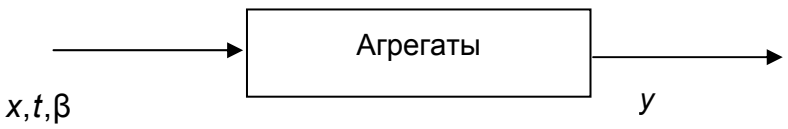


рис.2 Обобщенная схема управления на каждом подразделении

Конечным пользователем данной системы является дирекция предприятия.

Сложное поведение в многоуровневом управлении складывается из цепочек «распознавание ситуации – принятие решений – действие».

Распознавание ситуации происходит на основе теории нечетких множеств. При решении задачи распознавания приходится иметь дело с проявлением нечеткости при рассмотрении признаков, учитываемых в процедуре классификации.

Каждый плановый цикл представляет собой образ. Вследствие этого формируется база знаний, содержащая формальное описание заказов, ассоциативной связи «распознавание ситуации – принятие решений – действие», которые накапливаются в процессе работы интеллектуальной системы ситуационного прогноза. При формировании портфеля заказов система работает в режиме поиска ассоциативных связей. Из базы знаний выбираются ассоциативные связи, соответствующие данному портфелю заказов. Если на данный портфель заказов найдено несколько ассоциативных связей, то включается система экспертного оценивания и выбирается оптимальный вариант. Критериями оптимальности служит время выполнения заказа, получение прибыли от заказа.

При отсутствии ассоциативных связей на данный портфель заказов система переключается в режим формирования новых ассоциативных связей.

В соответствие с иерархией в управлении выделяют высший, средний и оперативный уровни. Высший уровень принимает стратегические решения: определяет цели управления, внешнюю политику, объем материальных, трудовых и финансовых ресурсов, разрабатывает долгосрочные планы и стратегии их исполнения.

На среднем уровне принимаются тактические решения, связанные с состоянием тактических планов, контролем за их выполнением, наблюдением за объемами всех ресурсов, разработкой управленческих решений для перевода предприятия на необходимый уровень, спрогнозированный планами.

На оперативном уровне принимаются оперативные решения, связанные с реализацией планов. Основное задание оперативного управления состоит в согласовании всех элементов управляющего процесса в пространстве и во времени с необходимой степенью децентрализации.

Заключение

В данной статье рассмотрена методология искусственного интеллекта в многоуровневом управлении. Формально представлено понятие многоуровневой производственной системы управления. В общем виде предложена динамическая детерминированная модель.

Библиография

1. Кофман Арнольд. Введение в теорию нечетких множеств. – М.: Радио и связь, 1982 г.
2. Нечеткие множества в моделях управления и искусственного интеллекта /Под ред. Д.А.Поспелова. – М.: Наука, 1986 г.

Авторская информация

Босик Александр Валентинович – аспирант Донецкого государственного института искусственного интеллекта, ул. Р.Люксембург, д.12, к.1101., г.Донецк - 55, 83055, Украина, e-mail: stolyarenko@rambler.ru

A REAL-TIME DECISION SUPPORT SYSTEM PROTOTYPE FOR MANAGEMENT OF A POWER BLOCK USING COGNITIVE GRAPHICS

A.P. Eremeev, V.N. Vagin

Abstract: *This report describes the basic tools of cognitive graphics for a real-time decision support system of a semiotic type on the example of the prototype for management and monitoring of a nuclear power block implemented on the basis of the tool complex G2+GDA. This work was supported by RFBR (project 02-07-90042).*

Introduction

Real time decision support systems (RTDSS) are hardware-software complexes, intended for the help to the decision making persons (DMP) at the management of complex objects and processes of a various nature in conditions of rigid temporary restrictions. When searching the decisions, expert models constructed on the basis of expert knowledge and heuristic methods of decision search are used. According to a modern classification of software, RTDSS are a class of integrated intelligent (expert) systems of a logic-linguistic type, combining strict mathematical decision search methods with non-strict, heuristic methods, based on expert knowledge [1-2].

The necessity of creating RTDSS is caused by continuously growing complexity of controlled objects and processes with simultaneous time reduction, yielded by DMP on problem situation analysis and acceptance of necessary managing actions.

Conceptually joining the approaches and methods of the decision support theory, theory of information systems, artificial intelligence and using the objective and subjective information, RTDSS provides DMP with the analysis of a soluble problem and directs him during decision search for increasing of a decision efficiency.

One of the basic problems at designing RTDSS is a choice of the suitable formal apparatus for a description of the decision support process and a construction on its base adequate (correct) decision making model (DMM). As such apparatus the production systems are usually used. However, available expert system design tools are guided on static problem domains, i.e. on situations, not requiring corrections of DMM and decision support strategies during decision search.

The peculiarities of problems, solved by RTDSS, are:

- the necessity of the temporary factor account at the problem situation description and during decision search;
- the necessity of decision making in conditions of temporary restrictions determined by a real controlled process;
- the impossibility of obtaining all objective information, necessary for the decision, and in this connection use of subjective, expert information;
- the complexity of a search, the necessity of an active participation of DMP;
- the presence of nondeterminism, the necessity of a correction and an introduction of additional information during decision search.

A basic purpose of RTDSS is to help to DMP at the control of complex objects and processes, revealing and preventioning of dangers, development of the recommendations, i.e. to help in the sanction of problem situations before they will become irreversible.

The main design principles of RTDSS are:

- openness and dynamics;
- adaptivity and learning;
- semioticity;
- distributivity and parallelism in information processing;
- application of a highperformed computer technique and efficient tools (complete environments) of the type G2 + GDA (G2 Diagnostic Assistant);
- application of cognitive graphics and a hypertext in information mapping.

The generalized architecture of RTDSS is given in fig. 1. In contrast with traditional expert systems, in RTDSS it is necessary to include the additional modeling block, and the forecasting one for analysis, an estimation of accepted decision consequences and a choice of the best recommendations. These blocks are implemented on the basis of the imitative modeling system G2+GDA. The choice of the tool complex G2 for implementing RTDSS is caused by integration basic high-effective technologies of complex program product development: object-oriented programming; open system technology and client-server technology; the active object graphics; a structured natural language and a hypertext for the information representation; decision search, based on production rules, procedures, dynamic (imitative) models; parallel fulfillment in real time of independent processes; the friendly interface with various types of the users (DMP, system manager, expert, knowledge engineer, programmer); a combination of technology of intelligent (expert) systems based on knowledge, with the technology of traditional programming.

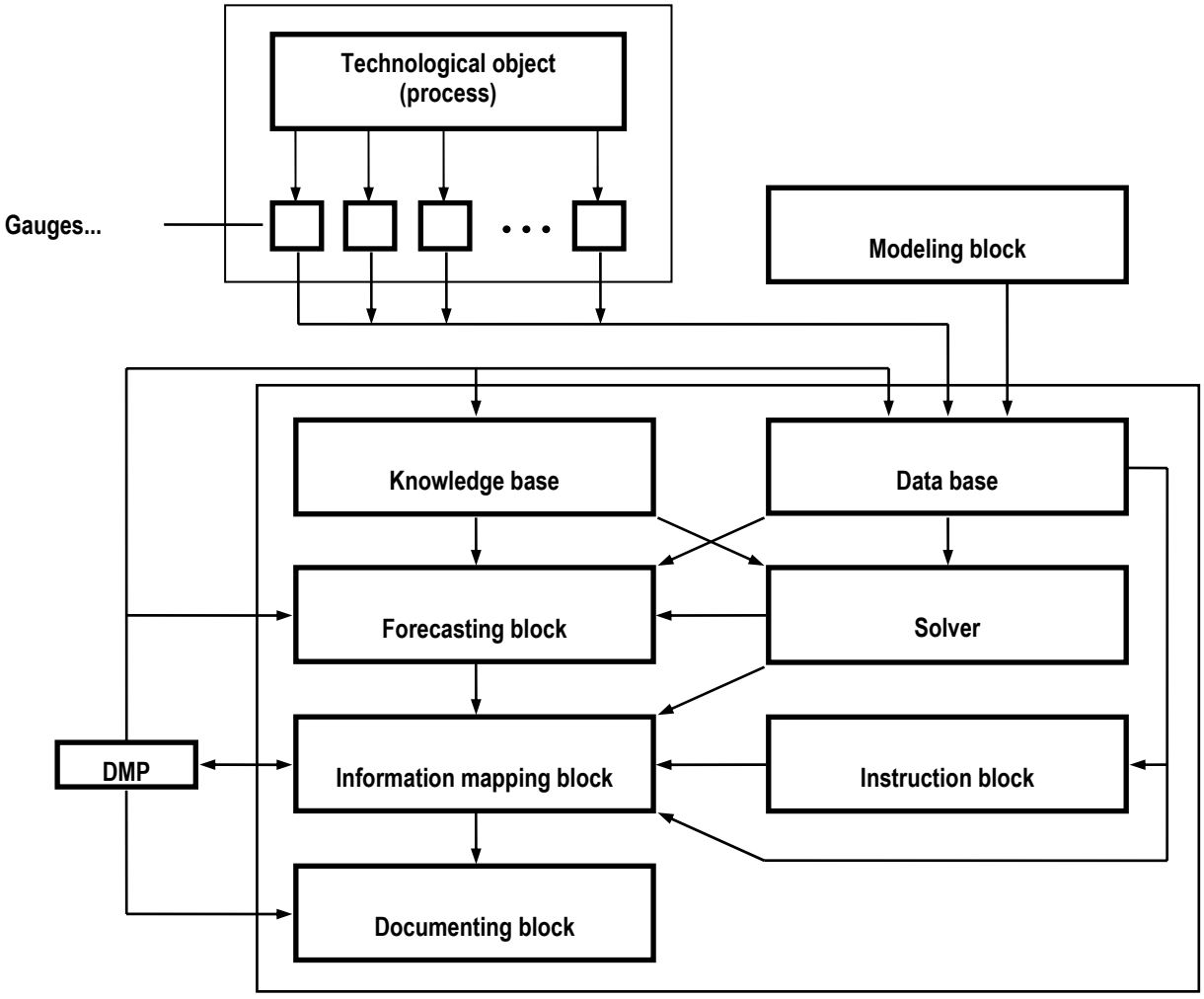


Fig. 1. The generalized architecture of RTDSS

The structure of base tools G2+GDA, necessary for RTDSS designing, consists of the interactive editor, tools of the graphic interface with an user, object-oriented graphics, graphic real time monitoring windows and animation, tools for display of connections between objects, interaction with an external environment, imitative modeling and processing of complex rules and procedures, tools for messages and explanations.

Such objects as a nuclear station power block are not made serially. Each object is unique and hence RTDSS for object management is also unique. But at designing RTDSS for various objects it is possible to use the same hardware platform and tools. Moreover, within the framework of G2+GDA class tools it is possible to design a tool environment of the same type of RTDSS. Such tool should give limited, but a rather complete set of primitives for the knowledge representation about an allocated class of objects and processes and about methods of

management by them. Naturally the tool environment, oriented on dynamic RTDSS, should be open for updating by new constructive elements.

RTDSS of the semiotic type is defined by the collection

$$SS = <M, R(M), F(M), F(SS)>, \text{ where}$$

- $M = \{M_1, ..., M_n\}$ is the set of formal or logic-linguistic models, implementing defined intelligent functions;
- R(M) is the function for selection of the necessary model in a current situation;
- $F(M) = \{F(M_1), ..., F(M_n)\}$ is the set of modification functions of models $M_1, ..., M_n$;
- F(SS) is the function for modification of SS system, i.e its base components M, R(M), F(M).

Applications of the first five design principles of RTDSS at implementing the prototype for monitoring and managing a nuclear station power block on the basis of tool complex G2+GDA was viewed in [3-10]. Here we present cognitive graphics means.

Cognitive Graphics in RTDSS

The instruction block (fig. 1) directs actions of DMP in planned transitive modes. It works automatically (on a situation) at switching on the appropriate mode. The information on a mode of object operation acts from a date base (DB).

The above requirements cause the necessity of the information representation in a knowledge base (KB) in the most convenient for DMP recognition a graphic form using the hypertext technology. The example of a fragment of external representation of KB for DMP as a decision tree with the necessary explanatory is submitted in fig.2, where 1, 2, ... 20 are block numbers in KB and selected items (1-2-3-6-13-19-18-20) map a control process. In a working mode activated by solver the chain of elements of a conclusion is allocated by other color. Between elements of KB in a working mode of the application, the relations can be established and be leafed.

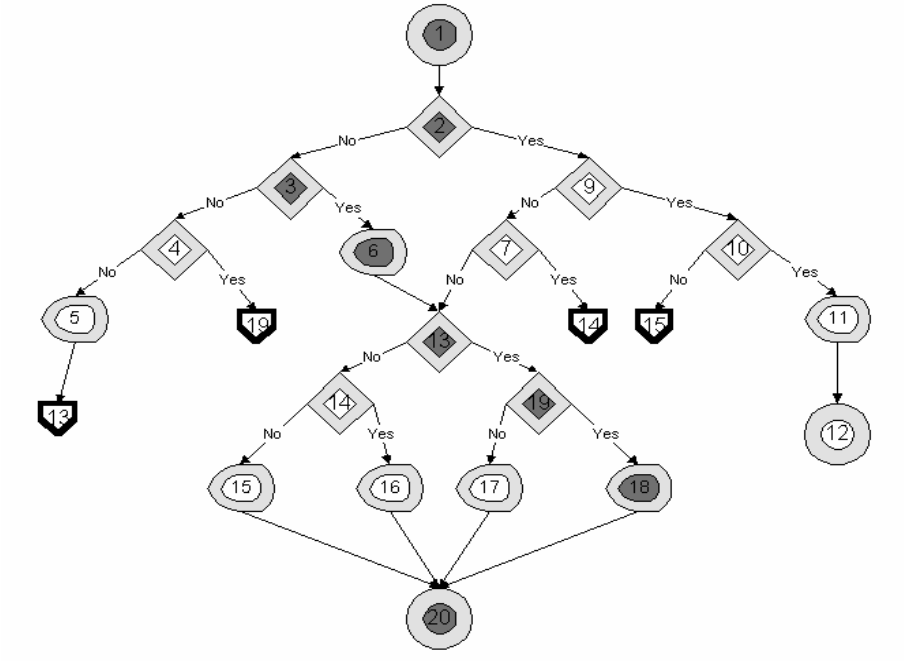


Fig.2. A fragment of the external representation of KB for DMP

The information mapping block carries out functions of information representation to DMP. Initial data for it are data from DB, results of an estimation of an object state, received by a solver, results of the forecasts, made by the forecasting block, and instructions given out by the instruction block.

The information, on the one hand, should be displayed in the form convenient for a fast recognition by DMP and, on the other hand, should be as it possible more complete. These requirements contradict each other, as the increasing of volume of the sign information decreases an ability of the person to perceive it. The problem is

decided by means of the multilevel circuit of information mapping with application of cognitive graphics and a hypertext technology.

The information mapping unit for a prototype is the display of a workstation. The motionless image on the screen corresponds to a static object state, and the movement displays a transition of an object in a new condition, on that DMP owes immediately respond.

For mapping the information a number of working spaces is entered.

1. A working space with the scheme of subsystems of a power block (MCP – the main circulating pump, CAP – the capacitor, EJ – the ejector plant, SCRZ – the subsystem of cooling the reactor zone), concerns just to such class MCP (CAP, EJ, SCRZ), its auxiliary systems and gauges. A subsystem is presented by an icon of a class. The given class has a few ports for connection with elements of the pump auxiliary subsystems: an independent contour, circulation of oil in bearings, locking water, cooling liquids. The gauges are represented on groups by graphic images of parameters fixed by the subsystem. The working space of parameters becomes visible at pressing of a mouse key on a parameter icon. Dynamics of processes, occurring in a subsystem, is displayed by change of a graphic image color of parameters. Looking on the scheme, DMP can qualitatively estimate a state of a subsystem and define which parameters are outside of a range of allowable meanings (these parameters are allocated by the red color). For more detailed acquainting with a state of the subsystem, the operator has access to the parameter subworkspace.

2. A working space of the urgent messages. At a normal condition of the pump in a working space of the urgent messages there is only one message "System <name> is in norm". This message has a green background and does not signal about any anomaly. At occurrence of abnormal situations in a working space there are the appropriate messages on a red background.

3. A working space of KB. In it the decision tree is located. Basically this space is intended for an expert and a knowledge engineer for creation and testing KB.

In a decision making mode the means for a choice, concealment, moving and change of the volume of the specified working spaces are given to DMP. With the help of these means he can design project interface with the application.

In addition to DB, a message base is implemented which contains all diagnostic messages, that can be given to the operator in a decision making mode. The message base places in a separate working space and consists of copies of a class "subsystem-message", being a subclass of a built-in class G2 message with an attribute text. The class "subsystem-message" is complemented by an integer attribute "message-number" and a logic attribute "message actuality".

In the mode of decision making a DMP receives the means for choosing, hiding, transferring and changing a dimension of given working spaces. With the help of these means DMP can configure interface with the application.

Analysis of activity of an operative-dispatching personnel has showed that it is preferably to use a three-level system of information representation on a control-labeled object or a process:

- a level of a system (or an object)on the whole, at which to be informed in what a (normal, abnormal or critical) state the system is and in what subsystems the deviations have arisen;
- a subsystem level at which a state of a particular subsystem appears;
- a level of directly measurable parameters with indicating not only parameter values but dynamics of their changing.

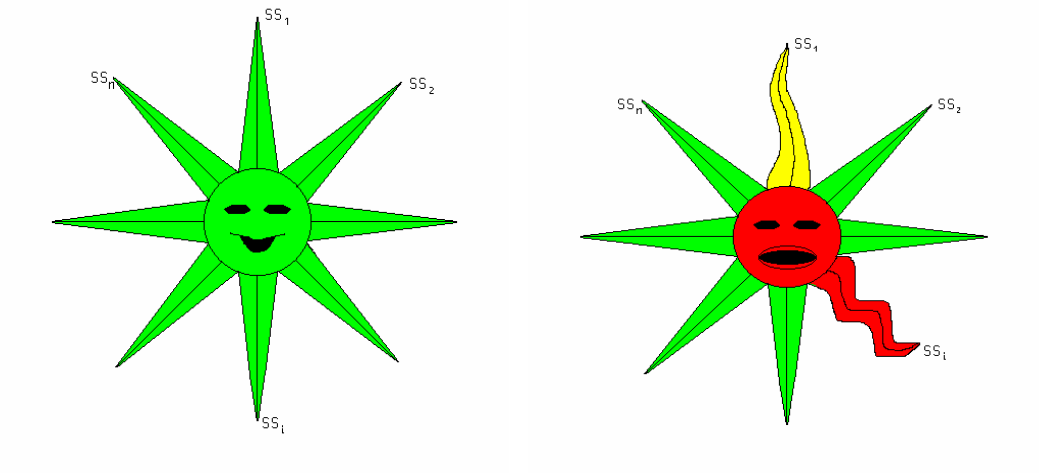
For building the graphic images a special editor is used. The basic type of image corresponds to a level of a system (an object) or a process on the whole and has a kind of the sun (kernel) with going out rays corresponding to subsystems (or generalized parameters of a process) SS_i of a top level. The number of rays is defined by a number of subsystems or generalized parameters and a thickness (a size of appropriate labels) of rays is determined by their importance.

The kernel may represent an image-face that can "smile", "wrinkle" or "cry" depending on an object state (normal, abnormal or critical accordingly) on the whole and its subsystems.

The kernel colour and rays may be red, yellow or green. The green colour of a kernel corresponds to a normal (on the staff) state (S_n) of the whole system (a process), the yellow one – to an abnormal state (S_{ab}) in presence of some deviations in subsystems and the red colour corresponds to a critical state (S_{cr}). In the state S_{cr} an

immediate interference of DMP is necessary. If transferring in S_{ab} has occurred from S_n (i.e. system functioning has become worse), such interference is also necessary. The connection between states (colours) of subsystems and a kernel colour is given by means of production rules storing in KB. The examples of images of the first type corresponding to the whole system for normal and critical object states are presented in fig. 3a, b accordingly.

The image of the second type corresponds to a subsystem level and characterises its state. If a subsystem SS_i (a



a – a normal state image b – a critical state image

Fig. 3. The image representation of a system state

generalized parameter), in turn, is a complex system, then an image type coincides with the previous one and a process of "disclosure" may be recursively continued.

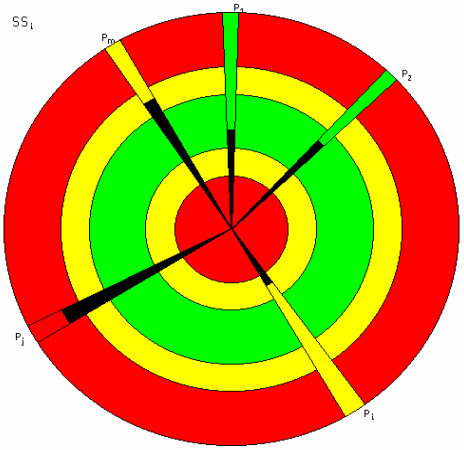


Fig. 4. The image of a parameter state

If the given subsystem SS_i is characterized by a collection of measurable parameters $\{P_i\}$, i = 1, 2, …, m, then the image of the third type occurs (see fig. 4) corresponding to the parameter level. The kernel and the external ring correspond to the zones of critical (upper and lower) parameter values and they are distinguished by red colour, the rings immediately adjoining to them present the zones of abnormal (upper and lower) parameter values and are marked by yellow colour and the central ring corresponding to the zone of normal parameter values has green colour. The rays present some normed indicators showing in what zone the values of appropriate parameters are found. The number and colour of zones are not strictly fixed and may be defined by DMP under adjustment.

Besides the main image types there are means of switching on additional screens mapping dynamics of changing appropriate parameters in the form of graphics and provided by timers (fig.5).

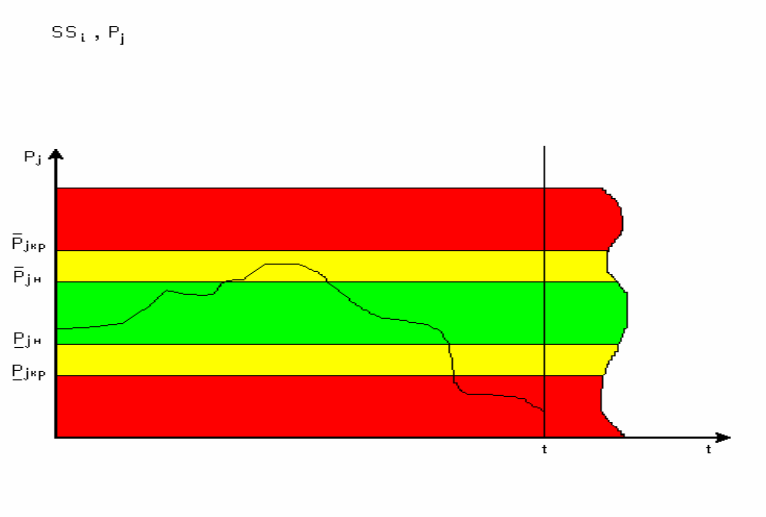


Fig. 5. The screen of changing a parameter

They allow to receive detail information about needed parameter values with the purpose of their normalization by means of some controlling actions. Note also, abnormal and critical states may be accompanied by appropriate sounds, twisting the rays changing the kernel "face" from "smiling" into "crying" and other additional means of paying attention to DMP.

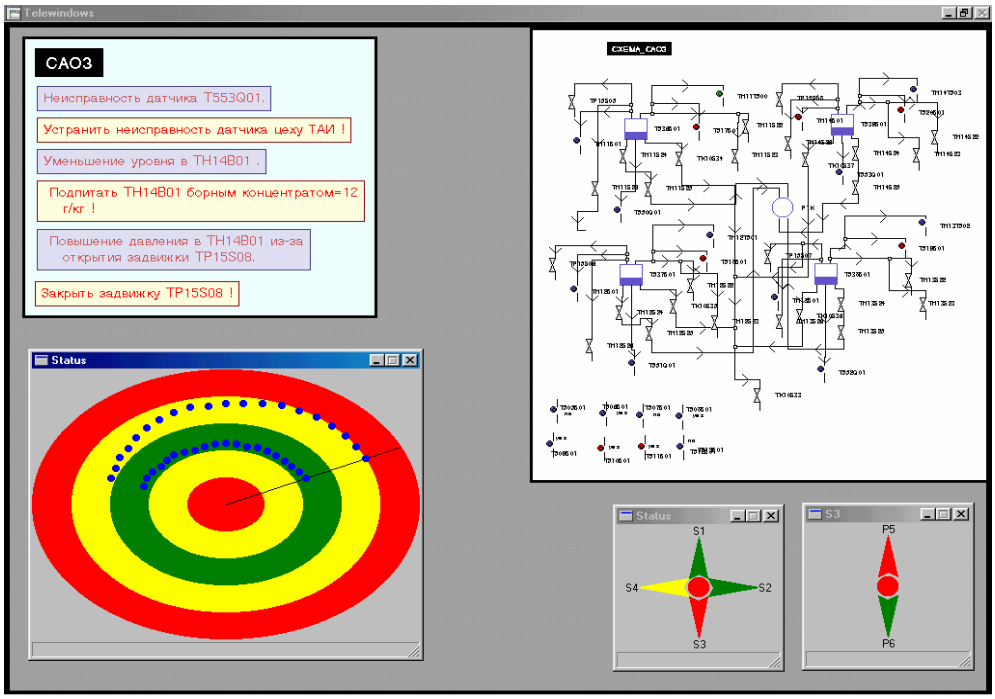


Fig. 6. The example of a poly-screen for DMP with cognitive graphics
(for the subsystem of cooling the reactor zone)

Setting up conformity between: a system state (an object, a process) on the whole and states of its subsystems SS_i, subsystem states SS_i and states of other subsystems, states of a subsystems SS_i and its measurable parameters P_j are accomplished by a special type of production rules containing in KB of the cognitive editor.

The example of a poly-screen for DMP with cognitive graphics is given in fig. 6.

Conclusion

During implementation of the RTDSS prototype for monitoring and management of a nuclear station power block, 31 classes of objects, 12 subclasses of variables, 2 subclass of connections between objects, 7 relations between objects were defined, 38 generalized rules, 45 procedures and 4 functions were written. For each continuous parameter, a subworkspace and an object for graphic display of dynamics of changing its values are defined. The cognitive graphics redactor was implemented as a separate module (block) and was connected with G2+GDA.

Bibliography

1. Vagin V.N., Eremeev A.P. Some Basic Principles of Design of Intelligent Systems for Supporting Real-Time Decision Making // J. of Computer and System Sciences International, Vol. 40, No. 6, 2001, pp. 953-961.

Vagin V.N., Yeremeyev A.P. Designing the Dynamic Decision Support Systems of a Semiotic Type // Proc. of the 1999 IEEE Internat. Symp. on Intelligent Control Systems / Intelligent Systems and Semiotics. Cambridge MA, Sept. 15-17, 1999, pp. 296-301.

2. Yeremeyev A.P. Organization of Semiotic Type Knowledge Representation Model for Dynamic Decision Support Systems // Proc. of Seventh Int. Conf. 'Artificial Intelligence and Information-Control Systems of Robots' AIICSR'97. Second Workshop on Applied Semiotics, Sept. 15, 1997, Smolenice Castle, Slovakia, pp 77-81.
3. Eremeev A.P., Shutova P.V. Learning and Adaptation in Real-Time Decision Support Systems of a Semiotic Type // Proc. of the IEEE Int. Conf. Artificial Intelligent Systems IEEE AIS'02, Sept. 5-10, 2002, Divnomorskoe, Russia, pp. 164-168.
4. Eremeev A.P., Vagin V.N. A Real Time Decision Support System for Monitoring and Management of a Complex Object Using Parallel Processing // Proc. of the IEEE Int. Conf. Artificial Intelligent Systems IEEE AIS'02, Sept. 5-10, 2002, Divnomorskoe, Russia, pp. 139-144.
5. Vagin V.N., Yeremeyev A.P. Parallel Inference in Knowledge Representation Models // Proc. of Symp. on Roboyics and Cybernetics, CESA'96 IMACS Multiconference, July 9-12, 1996. Lille-France, pp. 184-188.
6. Vagin V.N. Parallel Inference in Situation Control Systems // Architectures for Semiotic Modelling and Situation Analysis in Large Complex Systems. Proc. of the 1995 ISIC Workshop. 10th IEEE Intern. Symp. On Intel, Control. Monterey, USA, 1995, pp. 109-116.
7. Eremeev A.P., Tikhonov D.A., Shutova P.V. Support of Decision Making under Uncertainty on the Basis of the Non-Markov Model // Journal of Computer and Systems Sciences International, Vol. 38, No. 5, 1999, pp. 753-759.
8. Yeremeev A.P., Chibizova N.V. The prototype of a real time decision support system on the basis of the tool complex G2 // Proc. of Second Joint Conf. on Knowledge Based Software Engineering JCKBSE'96, Sazopol, Bulgaria, Sept. 21-22, 1996, pp.128-133.
9. Allachverdi N., Vagin V.N., Yeremeyev A.P. The Prototype of a Real Time Decision Support System for Monitoring and Management of a Nuclear Power Block // Proc. of 2nd Int. Conf. on Responsive Manufacturing ICRM-2002, June 26-28, 2002, University of Gaziantep, Turkey, pp. 602-608.

Author information

A. P. Eremeev - Applied Mathematics Department of the Moscow Power Engineering Institute (Technical University), eremeev@apmsun.mpei.ac.ru

V. N. Vagin - Applied Mathematics Department of the Moscow Power Engineering Institute (Technical University), vagin@apmsun.mpei.ac.ru

ОБ ОДНОМ МЕТОДЕ РЕШЕНИЯ КОНЕЧНОПАРАМЕТРИЗОВАННЫХ ЗАДАЧ С НЕЧЕТКИМИ ДАННЫМИ И ЕГО ПРОГРАММНАЯ РЕАЛИЗАЦИЯ

А.А. Лялецкий, А. Н. Яремчук

Резюме: *В данной работе развивается один из возможных подходов к нахождению решений параметризованных задач с нечеткими данными, а также приводится его связь с традиционными подходами теории вероятности. Описана специальная программная система прогнозирования, которая реализует математический аппарат, развитый в данной работе.*

Ключевые слова: *нечеткое множество, булева алгебра, теория вероятности.*

Введение

Математическая теория нечетких множеств, предложенная *Л.Заде* [1] позволяет описывать нечеткие понятия и знания, оперировать этими знаниями и делать нечеткие выводы. Основанные на этой теории методы поддержки систем принятия решений существенно расширяют области применения компьютеров для решения задач прогнозирования. В последнее время нечеткое прогнозирование является одной из самых активных и развивающихся областей исследований.

Предположим, что рассматривается некоторая (конечно) параметризованная задача и известна (экстенсиональная) частичная функция f, которая каждой последовательности $p_1,..., p_n$ точных значений всех параметров рассматриваемой задачи сопоставляет ее решение в виде точного значения функции $f(p_1,...,p_n)$, если последнее существует, и неопределена в противном случае. Теперь предположим, что значения параметров заданы нечетко, с некоторой долей уверенности, и что эти значения описываются подходящими нечеткими множествами, причем никакие ограничения на нечеткие данные не налагаются. Ставится проблема научиться находить решение параметризованной задачи и в этом нечетком случае.

Поставленная проблема формализуется следующим образом. Фиксируется некоторое множество-универсум A, которое включает в себя множества возможных значений всех параметров и множество всех решений, а функция нахождения решения f рассматривается как n-арная частичная операция над A. Нечеткие значения параметров $c_1,...,c_n$ определяются как частичные нечеткие множества над A, т.е. элементы множества $[0, 1]^A$, где для любых множеств X и Y запись Y^X обозначает множество всех частичных функций, действующих из X в Y. В рассмотрение также вводится специальное отношение $\leq$ над множеством $\{c_1,...,c_n\}$ нечетких значений параметров. Это отношение можно проинтерпретировать следующим образом: для любых нечетких множеств c_i и c_j $c_i \leq c_j$ (c_j $\leq$-предшествует c_j) в том и только том случае, когда нечеткое множество c_i "влияет" на нечеткое множество c_j. Здесь под термином "влияние" содержательно понимается способность нечеткого множества c_j изменяться в зависимости от того, какое подмножество событий множества A произойдет в множестве c_i, если элементы множества-носителя A рассматривать как события. Из такой интерпретации неформально следует, что разумно считать отношение $\leq$ отношением частичного порядка. При этом попарная независимость нечетких множеств интерпретируется как такое отношение частичного порядка $\leq$ над множеством $\{c_1,...,c_n\}$, при котором каждое нечеткое множество c_i $\leq$-сравнимо только с самим собой. Дополнительно каждому нечеткому множеству c_i из $\{c_1,...,c_n\}$, которое не является минимальным в смысле отношения $\leq$, приписывается специальная функция φ_{ci}: $Bool(A)\times...\times Bool(A)\times[0,1]^A \rightarrow [0,1]^A$, где в левой части Bool(A) встречается столько раз, сколько существует непосредственных $\leq$-предшественников элемента c_i. Функция φ_{ci} формально описывает изменения нечеткого множества c_i в зависимости от того, какие события случились в нечетких множествах, непосредственно $\leq$-предшествующих c_i. Отметим, что если события $c_1,...,c_n$ являются попарно независимыми, то φ есть пустая функция (т.е. ее график и области определения и значений являются пустыми множествами).

Таким образом, структуру входных данных рассматриваемой нечеткой задачи составляют:

1. множество-универсум A;

2. частичная операция f над ним;
3. конечная последовательность частичных нечетких множеств $c_1,...,c_n$, количество элементов которой совпадает с арностью f;
4. отношение частичного порядка $\leq$ над $\{c_1,...,c_n\}$;
5. отображение φ, которое каждому $\leq$-неминимальному нечеткому множеству c_i приписывает функцию φ_{ci}: $Bool(A)\times...\times Bool(A)\times[0,1]^A \rightarrow [0,1]^A$, арность которой совпадает с количеством непосредственных $\leq$-предшественников c_i.

В данной работе приводится краткое описание математического аппарата для решений задач такого типа. На основе этого аппарата создана программная система, которая решает такие задачи в случае, когда множество A является конечным, и выдает следующее:

(i) частичное нечеткое множество над совокупностью всех частичных нечетких множеств над A, которое может интерпретироваться как совокупность возможных прогнозов, каждому из которых поставлена в соответствие вероятность того, что именно этот прогноз случится;

(ii) частичное нечеткое множество над A, которое получается как «совокупный» прогноз, построенный из прогнозов (нечетких множеств) пункта (i).

Отметим, что если рассматривается случай, когда нечеткие множества являются попарно независимыми (и множество-универсум A является конечным), написанная программа дает на выход только «совокупный» прогноз из пункта (ii). Причиной этого служит то, что в этом случае совокупность всех возможных прогнозов состоит из единственного прогноза, которому приписана вероятность, равная 1.

Математическое обоснование

Ниже, параллельно с описанием, обосновывается корректность нашего метода путем установления его связи с традиционными подходами теории вероятностей.

Мы предполагаем известными основные теоретико-множественные понятия, понятия теории нечетких множеств [1], а также понятия и конструкции из теории частичного порядка [2] и, в частности, теории булевых алгебр [2-4]. Мы будем различать понятия алгебраической и теоретико-множественной полурешеток (решеток, булевых алгебр). Так, под алгебраической полурешеткой мы будем понимать идемпотентную коммутативную полугруппу [2], а под теоретико-множественной (верхней) полурешеткой [2] – частично упорядоченное множество, любая пара элементов которого имеет точную верхнюю грань (аналогично для решеток и булевых алгебр).

Известно [2, 4], что существует конструктивное взаимно однозначное соответствие между алгебраическими полурешетками (решетками, булевыми алгебрами) и их теоретико-множественными аналогами, и что каждому истинному свойству теории алгебраических полурешеток (решеток, булевых алгебр), выраженном на языке классической логики произвольного порядка, соответствует аналогичное свойство теории теоретико-множественных (верхних) полурешеток (решеток, булевых алгебр), также выраженном на языке классической логики такого же порядка. По умолчанию под булевой алгеброй будем понимать алгебраическую булеву алгебру.

Мы будем придерживаться определения квазимеры (меры), взятого из [3], где под квазимерой (мерой) понимается любая аддитивная (вполне аддитивная) функция, определенная на алгебраической булевой алгебре, которая принимает положительные действительные значения и значения которой на наименьшем элементе (нуле) равно числу 0. Под квазивероятностью (вероятностью), будем понимать любую такую квазимеру (меру), значение которой на наибольшем элементе (единице) булевой алгебры, над которой она задана, равно числу 1. Элементы a_1 и a_2 некоторой булевой алгебры B будем называть вероятностно независимыми относительно заданной над B квазивероятности P тогда и только тогда, когда $P(a_1 \wedge a_2) = P(a_1)*P(a_2)$.

Поскольку программа может работать только с конечным множеством-носителем A, а одна из его интерпретаций есть множество событий, которые являются элементами подходящей конечной булевой алгебры, то нас, прежде всего, будут интересовать именно конечные булевы алгебры.

Напомним, что все конечные булевы алгебры являются полными булевыми алгебрами. Следовательно [2,3], каждый элемент конечной булевой алгебры может быть выражен как конечная дизъюнкция атомов

булевой алгебры, где, как и в [2], атомом называется любой элемент булевой алгебры, который непосредственно предшествует наименьшему элементу (в классической теории вероятности атомы принято называть элементарными событиями).

Обозначим через Σ сигнатуру $<\vee, \wedge, ^{-1}, 0, 1>$ с арностями 2, 2, 1, 0, 0 соответственно, и пусть M – некоторое множество. Тогда под основным термом сигнатуры Σ над множеством M будем понимать любой терм сигнатуры Σ, который не содержит переменных и в котором элементы множества M играют роль констант. Значение основного терма в некоторой алгебраической системе сигнатуры Σ определяется так же, как и для обычного терма (см. [4]). Для каждой булевой алгебры B рассмотрим отношение эквивалентности $\sim_B$, собирающее вместе такие и только такие ее основные термы, которые имеют одинаковые значения, и для каждого основного терма t булевой алгебры B через $[t]_B$ обозначим класс эквивалентности $\sim_B$, который содержит терм t. Из вышесказанного следует, что каждое конечное множество M однозначно определяет алгебру $<T(M)/\sim_B, \vee, \wedge, ^{-1}, [0], [1]>$, где T(M) – множество, состоящее из всех основных термов над множеством M, дизъюнкция $\vee$ воспринимается как такая операция, которая каждые классы эквивалентности $[t_1]_B$ и $[t_2]_B$ отображает в класс $[t_1 \vee t_2]_B$, конъюнкция $\wedge$ – каждые классы эквивалентности $[t_1]_B$ и $[t_2]_B$ отображает в класс $[t_1 \wedge t_2]_B$, операция дополнения $^{-1}$ – каждый класс $[t]$ отображает в класс $[t^{-1}]$. Легко проверить, что отношение эквивалентности $\sim_B$ является конгруэнцией и что алгебра $<T(M)/\sim_B, \vee, \wedge, ^{-1}, [0], [1]>$ удовлетворяет всем тождествам булевой алгебры. Так построенные по любому конечному множеству M булевы фактор-алгебры $<T(M)/\sim_B, \vee, \wedge, ^{-1}, [0], [1]>$ будем называть термальными булевыми алгебрами над множеством M. В [4] доказано, что любая булева алгебра, для которой M является конечным множеством всех атомов, изоморфна термальной над M булевой алгебре. Следовательно, две конечные булевы алгебры являются изоморфными тогда и только тогда, когда существует взаимно однозначное соответствие между множествами их атомов. Если через A обозначить антицепь (в соответствии с [2], антицепь – это любое максимальное по мощности множество попарно несравнимых элементов) булевой алгебры B, то из предыдущего следует, что алгебра $<T(A)/\sim_B, \vee, \wedge, ^{-1}, [0], [1]>$ изоморфна термальной булевой алгебре $<T(M)/\sim_B, \vee, \wedge, ^{-1}, [0], [1]>$, где $\sim_B$ - отношение эквивалентности, определенное выше.

Из вышесказанного и из свойства аддитивности вероятности следует, что каждая вероятность над конечной булевой алгеброй полностью и однозначно определяется заданием вероятности на всех ее атомах. Заметим, что при этом вероятность любого элемента этой булевой алгебры можно эффективно вычислить.

Будем говорить, что две вероятности P_1 и P_2, заданные на булевых алгебрах B_1 и B_2 соответственно, изоморфны, тогда и только тогда, когда существует такой изоморфизм φ между алгебрами B_1 и B_2, который сохраняет вероятности, т.е. для любого элемента b_1 булевой алгебры B_1 верно, что $P_1(b_1) = P_2(b_1\varphi)$. Следовательно, для того, чтобы две вероятности, заданные над булевыми алгебрами B_1 и B_2 соответственно, были изоморфными, необходимо и достаточно, чтобы между множествами всех атомов булевых алгебр B_1 и B_2 существовало взаимно однозначное соответствие, которое сохраняет эти вероятности.

Упорядоченную пятерку $<A, f, (c_1,...,c_n), \leq, \varphi>$, все компоненты которой удовлетворяют условиям 1-5 из введения, будем называть структурой нечеткой задачи. Для каждой структуры нечеткой задачи введем “естественное” понятие интерпретации в соответствующую вероятностную структуру.

Вначале мы рассмотрим случай независимых между собой нечетких множеств (см. введение). В этом случае в структуре нечеткой задачи $<A, f, (c_1,...,c_n), \leq, \varphi>$ отношение $\leq$ совпадает с обычным отношением равенства и φ - пустая функция. Пусть $\{c_1,...,c_n\}$ – совокупность независимых нечетких множеств с попарно непересекающимися носителями $A_1,...,A_n$ соответственно. Тогда будем говорить, что совокупность нечетких множеств $\{c_1,...,c_n\}$ интерпретируется как квазивероятность P, заданная на булевой алгебре B, в том и только том случае, когда $A_1 \cup ... \cup A_n$ является вероятностно независимым множеством этой булевой алгебры, порождающим ее, причем для любого числа i ($1 \leq i \leq n$) и для любого элемента a множества A_i $P(a) = c_i(a)$ (при этом часто $A_1 \cup ... \cup A_n$ является антицепью). Следовательно, в соответствии с результатом из [3], совокупность нечетких множеств $\{c_1,...,c_n\}$ тогда и только тогда имеет интерпретацию посредством некоторой квазивероятности P, когда их носители попарно не пересекаются и $\Sigma c_i(a_j) = 1$. Поэтому, если совокупность нечетких множеств $\{c_1,...,c_n\}$ интерпретируются посредством квазивероятности P, то

квазивероятность того, чтобы одновременно «случились события» $a_1,...,a_n$, равна $P(a_1)*...*P(a_n)$ и квазивероятность того, чтобы после применения операции «нахождения решения» «случилось событие» b, равна $\Sigma P(a_1)*...*P(a_n)$, где сумма берется по таким и только таким n-кам $(a_1,..., a_n)$, составленным из элементов носителя нечетких множеств $c_1,...,c_n$ соответственно, что значение $f(a_1,...,a_n)$ определено и равно b.

Случай, когда структура нечеткой задачи $<A, f, (c_1,...,c_n), \leq, \varphi>$ содержит зависимости между нечеткими множествами, сводится к «независимому» случаю следующим образом. Поднимаясь по частично упорядоченному множеству «снизу вверх» посредством функций «зависимостей» φ_{ci} и просматривая определенным образом все подмножества множества-носителя, мы как бы учитываем зависимости, и, таким образом, приводим рассмотрение «зависимого» случаю к множеству «независимых» случаев, решения которых и образуют так называемое нечеткое множество «возможных прогнозов». По последнему строится также «совокупный прогноз», который является в некотором смысле его «осреднением».

Заметим, что проведены математические исследования, необходимые для обоснования корректности выбранного способа решения задачи в «зависимом» случае. При этом важную роль сыграли результаты из [3].

Теперь у нас есть все необходимое, чтобы описать схему работы программы.

Схема работы программы

Вначале в программу вводится структура нечеткой задачи $<A, f, (c_1,...,c_n), \leq, \varphi>$. При этом в программе отдельно выделены зависимый и независимый случаи (в частности, в независимом случае не требуется задавать отношение частичного порядка и функцию φ, поскольку тогда они определяются структурой однозначно). Далее ищется решение нечеткой задачи. В независимом случае для этого, в частности, перебираются всевозможные n-ки вида $(a_1,...,a_n)$, и вычисляются значения $f(a_1,...,a_n)$, где для любого i ($1 \leq i \leq n$) a_i – элемент множества-носителя A. В зависимом случае производится перебор по всевозможным k-кам подмножеств $A_{i1},...,A_{ik}$ носителя A, где $c_{i1},...,c_{ik}$ – подпоследовательность последовательности $c_1,...,c_n$, которая состоит из всех $\leq$-неминимальных нечетких множеств; для каждой такой k-ки $A_{i1},...,A_{ik}$ «снизу вверх» происходит подъем по отношению частичного порядка $\leq$ с помощью функций «зависимостей» φ_{ci}, а далее над «переработанными» нечеткими множествами проводятся те же вычисления, что и в независимом случае. Окончательным результатом работы программы является частичное нечеткое множество над совокупностью всех частичных нечетких множеств над A, которое затем посредством определенной процедуры преобразуется в «совокупный прогноз» (см. пункты (i), (ii) из введения).

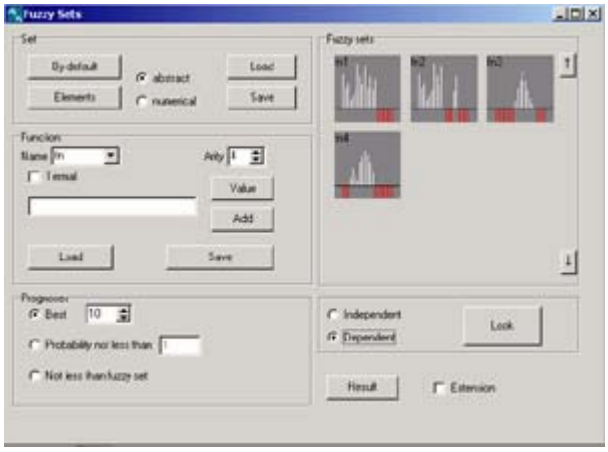
Описание интерфейса программы с пользователем

Интерфейс программы с пользователем осуществляется в обычном стиле Microsoft Windows в виде определенных окон. Примером может служить стартовое окно программы.

В виде окон специального вида осуществляется выполнение следующих операций:

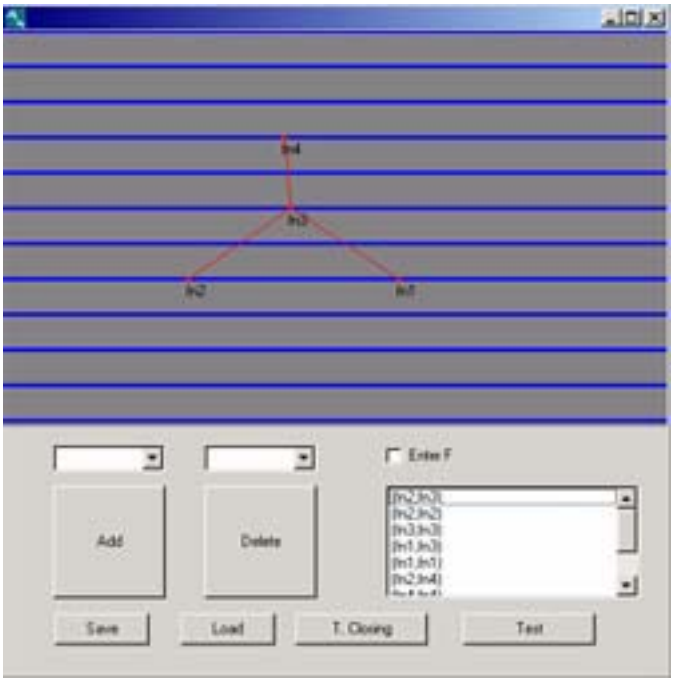
1. *Ввод множества-носителя*. При этом отдельно предусмотрены возможности специальных «удобных» заданий конечных подмножеств множества всех рациональных чисел (в этом случае множество считается “числовым” («numerical»); в общем же случае оно считается «абстрактным» («abstract») и задается перечислением элементов). После того как множество задано, его можно сохранить для дальнейшего использования.

2. *Ввод операции «нахождения решения»*. На заданном выше множестве-носителе необходимо определить операцию «нахождения решения». Определяются арность операции и ее имя. Функции задаются двумя способами. Первый – с помощью ее графика. Для «числовых» множеств и функций +, -, /, * (их арность должна равняться 2), появляется кнопка «Count», с помощью которой можно автоматически заполнить значения соответствующей функции. Второй способ ввода функции – с помощью ее термальной записи, основываясь на ранее введенных функциях. Как и в случае с множествами, введенные функции можно сохранять для их дальнейшего использования.



3. *Задание нечетких множеств*. После ввода некоторой функции, существует возможность задать соответствующую ей конечную последовательность нечетких множеств. После ввода нечетких множеств, в случае их независимости на данном этапе можно уже получить результат.

4. *Случай с зависимыми множествами*. Если нечеткие множества являются зависимыми между собой, то для учета этой зависимости необходимо ввести отношение частичного порядка: для этого в стартовом окне следует отметить пункт «Dependent», и после нажатия на кнопку «Look» откроется окно:



Существует 2 типа ввода порядка. Первый – с помощью пар вида (c_i,c_j), где c_i и c_j –некоторые нечеткие множества, для которых $c_i \le c_j$. Второй состоит в представлении отношения частичного порядка в виде графа, вершины которого являются нечеткими множествами. При этом для любой пары нечетких множеств, c_i и c_j, $c_i \le c_j$ тогда и только тогда, если существует путь из c_i в c_j, такой, что для любых c_k, c_l этого пути c_k «ниже» c_l.

5. *Ввод функций "зависимостей"* φ_{ci}. Каждому нечеткому множеству в случае их зависимости необходимо приписать функцию "зависимостей" φ_{ci}: $Bool(A) \times \ldots \times Bool(A) \times [0,1]^A \rightarrow [0,1]^A$, задание которых регулируется с помощью соответствующего окна.

6. *Получение результатов в «зависимом» случае*. Вначале необходимо определить, какой результат (совокупность нечетких множеств и «совокупный прогноз») нам нужен. В программе имеется возможность конструировать из допустимых предикатов новые предикаты с помощью логических операций отрицания, дизъюнкции и конъюнкции. В последнем предложении под допустимыми предикатами понимаются такие предикаты:

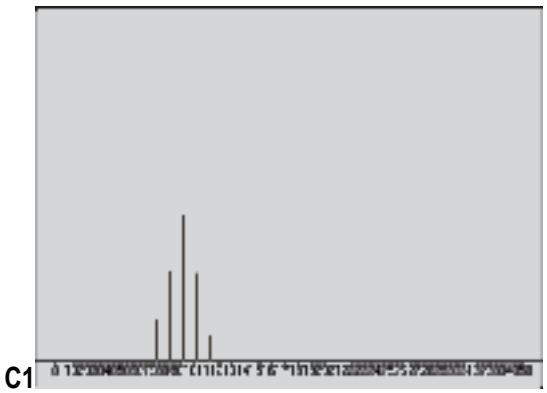
- «Лучших n нечетких множеств», т.е. таких n нечетких множеств, которым сопоставлена наибольшая вероятность – n самых вероятных прогнозов;
- «Вероятность не ниже u», т.е. такие и только такие нечеткие множества, которым сопоставлена вероятность не ниже действительного числа u из отрезка [0,1];
- «Не ниже нечеткого множества c», т.е. такие и только такие нечеткие множества над носителем, графики которых «не ниже» нечеткого множества c.

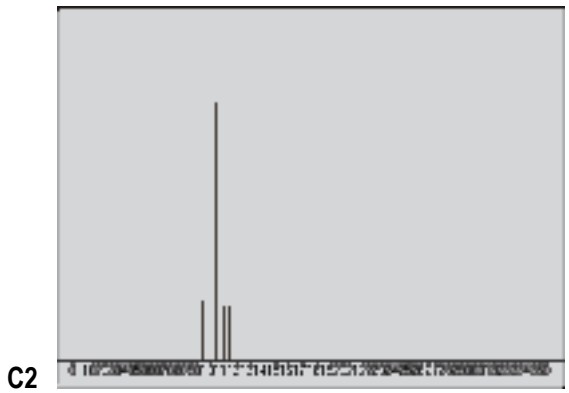
7. *Расширения основного множества-носителя*. В программе встроена опция, позволяющая расширять носитель в случае, когда имеется возможность получать прогнозы над надмножествами носителя. Это свойство может быть полезным тогда, когда операция «нахождения решения» задана над собственным надмножеством носителя.

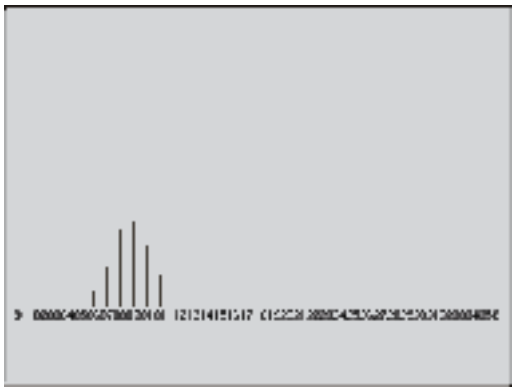
Пример

Метод, которому посвящена эта работа, может быть полезен для решения многих «нечетких» задач, возникающих на практике. Приведем один из них.

Задача. Два инвестора I_1 и I_2 планируют вложить деньги в три организации O_1, O_2 и O_3, причем первый собирается инвестировать сначала организацию O_1, а затем – в O_2 при (нечетко) ограниченном бюджете, а второй – только организацию O_3. При этом объемы инвестиций описываются нечеткими множествами c_1, c_2 и c_3 соответственно (которые программа отображает на экране в виде графиков). Надо найти нечеткое множество, которое описывает сумму вложений во все три организации.


C1


C2


C3

В результате работы программы над поставленной задачей генерируется несколько «возможных прогнозов». Ниже приводятся два самых вероятных «возможных прогноза».

Отметим, что программа написана в системе разработки приложений Delphi, и что работа [5] служила источником примеров для отладки программы.

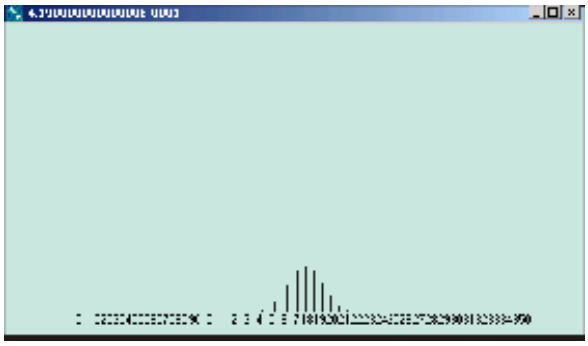
Заключение

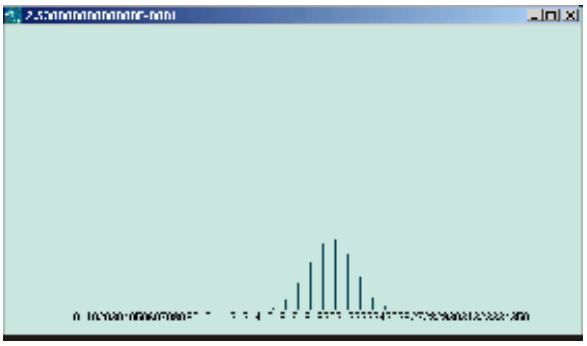
Развитый в данной работе математический аппарат имеет широкий спектр практических приложений. Он может быть использован для нахождения решений таких нечетко поставленных параметризованных

задач, для которых известны точные решения при точных значениях параметров. В частности, этот аппарат может быть полезен при решении целого ряда задач из следующих областей:

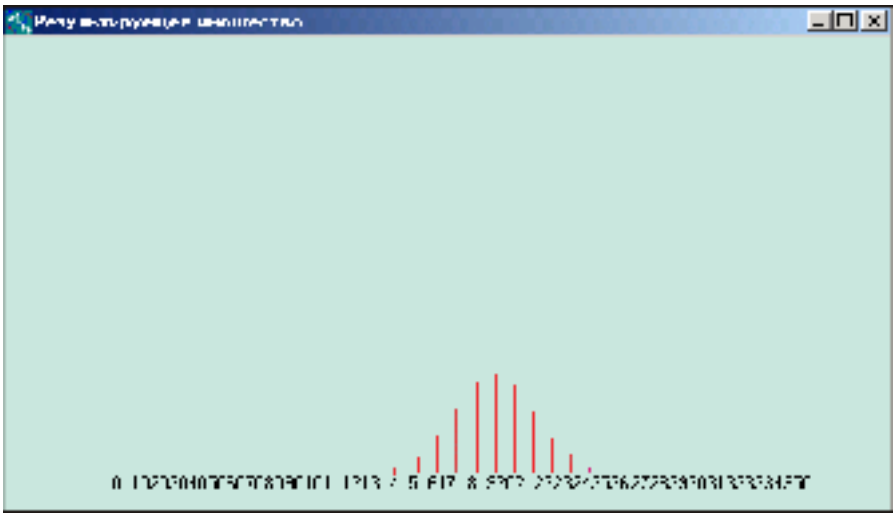
- проблемы классификации (медицинская диагностика и т.п.);
- исследование рисковых и критических ситуаций;
- финансовый анализ (рынки ценных бумаг);
- исследование данных (корпоративные хранилища);
- совершенствование стратегий управления и координации действий.

В дальнейшем данный подход планируется расширить до различных видов нечетких логик.





«Совокупный прогноз», построенный по всем «возможным прогнозам»:



Библиография

1. Заде Л. Понятие лингвистической переменной и ее применение к принятию приближенных решений, М.:Мир,1976
2. Гретцер Г. Общая теория решеток. М.: Мир, 1982, 452 с.
3. Владимиров Д.А. Булевы алгебры. М.: Наука, 1969, 318 с.
4. Мальцев А. И. Алгебраические системы. М.: Наука, 1970, 392 с.
5. Бочарников В.П., Свешников С.В., Возняк С.Н. Прогнозные коммерческие расчеты и анализ рисков на Fuzzy for Excel. К.: Инекс, 2000, 159 с.

Сведения об авторах

А.А. Лялецкий, А. Н. Яремчук - Факультет кибернетики, Киевский Национальный Университет имени Тараса Шевченко, бул. Акад. Глушкова, дом 6, 03022, Киев, Украина, e-mail: lav@unicyb.kiev.ua

МОДЕЛИРОВАНИЕ ЗАДАЧИ ВЫБОРА С ИСПОЛЬЗОВАНИЕМ ГИПЕРОТНОШЕНИЙ

М.В. Масалитина

Аннотация: *В работе затрагиваются методологические вопросы описания моделей для задач выбора и принятия решений. На примере одного метода многокритериальной задачи принятия решений рассматривается возможность и преимущества описания модели, основанной на гиперграфовых структурах.*

Введение

Как известно, знания о предметной области можно разделить на интенсиональные и экстенсиональные. Интенсиональное представление знаний описывает закономерности и связи, которыми объясняется структура данных. В свою очередь, экстенсиональное представление знаний связано с описанием и фиксацией конкретных объектов из предметной области, что позволяет в некоторой степени учитывать семантику конкретной задачи. Если рассмотреть задачи выбора с точки зрения методологии описывающих их моделей, то существующие механизмы и методы выбора можно также разделить на интесиональные и экстенсиональные. Например, те механизмы выбора в многокритериальной задачи принятия решений, в которых решающее правило определено в явном виде (например, функция полезности) можно отнести к интенсиональным. Экстенсиональное представление знаний используется в методах с конструктивным построением решающих правил, в том числе последовательное выявление предпочтений эксперта. Заметим, что для задач со слабо формализуемыми условиями целесообразнее, на наш взгляд, изначально применять экстенсиональный подход к построению решающих правил. А переход от экстенсионального к интесиональному знанию возможен на той стадии, когда формальный алгоритм построения решающего правила уже сконструирован и продемонстрировал свою эффективность. Лишь после этого возможно изучение механизмов, в том числе и анализ структуры данных, за счет которых достигается полученная эффективность.

Языки описания выбора

К настоящему времени сложилось три основных языка описания выбора. При использовании *критериального* языка предполагается, что каждую отдельно взятую альтернативу можно оценить некоторой величиной, после чего сравнение альтернатив сводится к сравнению соответствующих им величин. Многокритериальные задачи не имеют однозначного общего решения. Поэтому предлагается много способов придать многокритериальной задаче частный вид, допускающий единственное общее решение. Естественно, что для разных способов эти решения являются в общем случае различными. Поэтому едва ли не главное в решении многокритериальной задачи – обоснование данного вида ее постановки.

Язык бинарных отношений является обобщением многокритериального языка и основан на учете того факта, что когда дается оценку некоторой альтернативе, то эта оценка всегда является относительной, т.е. явно или чаще неявно в качестве базы или системы отсчета для сравнения используются другие альтернативы из исследуемого множества или из генеральной совокупности. Существуют различные способы задания бинарных отношений: непосредственный, матричный, с использованием графов предпочтений, метод сечений и др. Отношения между альтернативами одной пары выражают через понятия эквивалентности, порядка и доминирования.

Наиболее общим является *язык функций выбора*. Он основан на теории множеств и позволяет оперировать с отображениями множеств на свои подмножества, соответствующие различным вариантам выбора. Потенциально этот язык позволяет описывать любой выбор.

Среди задач выбора и принятия решений очень широкий класс составляют многокритериальные задачи выбора. Здесь можно выделить два основных направления. В методах обоих направлений в качестве

основы для построения решающего правила использует модель предпочтений эксперта. Разница заключается в способах агрегирования этих предпочтений. В первом локальные предпочтения (по каждому критерию) сворачиваются в функцию, при этом изучаются условия для свертки, специальные формы и методы построения этой функции, а также исследуются возможности оптимизации. В методах второго направления предпочтения эксперта описываются бинарными отношениями. В том числе для моделирования локальных предпочтений эксперта также вводятся ограничивающие пороги (безразличия, предпочтения, вето) по каждому критерию. Заметим, что конструктивный подход к построению решающих правил выбора, который используется в некоторых методах второго направления, позволяет учитывать более сложные факторы, влияющие на выбор – то есть частично учитывают семантику (контекст) задачи принятия решения.

Если теперь описать методы обоих направлений на языке функций выбора, то оказывается, что большинство из них реализует так называемые классически-рациональные механизмы выбора. В основе построения таких механизмов и функций лежит "презумпция парнодоминантности", а именно предположение о том, что любой разумный выбор всегда может быть сведен к выбору лучших ("доминирующих") вариантов при их попарном сравнении по так или иначе понимаемым отношениям предпочтения.

В рамках этого класса задач функции выбора принадлежат пересечениям некоторых характеристических областей. Каждая такая область (класс функций) выделяет все функции выбора, обладающие некоторым свойством. Основными свойства являются следующие.

1. Свойство наследования. Функция выбора $C(\cdot)$ удовлетворяет условию наследования (обозначение - H), если для всех $X, X' \;\; X' \subseteq X \Rightarrow C(X') \supseteq C(X) \cap X'$

2. Свойство константности. Функция $C(\cdot)$ удовлетворяет условию строгого наследования, или константности выбора (обозначение - K), если одновременно выполняются два следующих независимых условия: для всех X, X'

$$X' \subseteq X \Rightarrow \begin{cases} \text{если } C(X) = \emptyset, \text{ то } C(X') = \emptyset \\ \text{если } C(X) \cap X' \neq \emptyset, \text{ то } C(X') = C(X) \cap X' \end{cases}$$

3. Условие согласия. Функция $C(\cdot)$ удовлетворяет условию согласия (обозначение - C), если для всех $X, X' \;\; X = X' \cup X'' \Rightarrow C(X) \supseteq C(X') \cap C(X'')$

4. Условие независимости от отбрасывания отвергнутых вариантов. Функция $C(\cdot)$ удовлетворяет условию независимости от отбрасывания отвергнутых вариантов (обозначение - O), если для всех X, X' $C(X) \subseteq X' \subseteq X \Rightarrow C(X') = C(X)$

Нарушение классической рациональности

Как уже было сказано, функции выбора для большинства известных методов решения многокритериальных задач относятся к классически-рациональным, или, в терминах характеристических областей, принадлежат пересечениям H∩C, H∩C∩O либо области K. Однако в [1] показано, что естественны также механизмы выбора, которые порождают функции выбора, не удовлетворяющие условиям классической рациональности. Заметим, что необходимость выхода за рамки схемы классической рациональности прослеживается и самих методах указанных направлений, так как предлагаемые в них механизмы выбора довольно часто являются эффективными на практике, но теоретически не удовлетворяющими условиям классической рациональности.

Таким образом, понятие "рационального" выбора требует более тщательного изучения. Если же обратиться к формальному критерию классической рациональности выбора – к принципу Кондорсе, в основе которого лежит механизм выявления лучшей альтернативы путем попарного сравнения альтернатив, – то оказывается, что главной "причиной" нарушения классической рациональности является то, что классическая логика выбора, которая в своем чистом виде воплощена в абстрактном парнодоминантном механизме, опирается на бинарные структуры. В отличие от этого, примеры неклассической логики выбора имеют в качестве своих структур более сложные, многоместные отношения. Использование многомерных структур – гиперотношений – в качестве основы для описания

решающих правил может позволить описать более сложные механизмы выбора и приблизиться к более четкому пониманию тех механизмов выбора, которые человек использует на практике.

Если предположить, что решающее правило в модели выбора удалось описать в терминах гиперотношений, следующим шагом должно быть исследование механизмов, позволяющих *выявлять* предпочтения эксперта и представлять их в виде гипрграфовых конструкций. Здесь существенную помощь может оказать исследование свойств гиперотношений, в частности – разложимость гиперотношений по бинарным, замкнутость классов функций выбора относительно операций объединения и пересечения, а также условия рациональности функции выбора. Все эти свойства могут быть полезны для построения инструментария, позволяющего выявлять предпочтения эксперта, так как психологически человеку легче осуществлять попарные, а не многомерные сравнения.

Пример перехода к гиперграфовой модели выбора

Возможность перехода от классической модели, построенной на бинарных отношениях, к модели, использующей гиперотношения, рассмотрим на примере метода ЗАПРОС (замкнутые процедуры у опорных ситуаций) [2]. Выбор этого метода не случаен: на наш взгляд, это один из методов, позволяющих строить решающие правила с наименьшей потерей информативности. С одной стороны, этот метода опирается на *качественный* анализ проблемы; кроме того, способ построения решающего правила здесь является экстенсиональным, то есть позволяющим учесть семантику задачи тогда, когда предметная область (в данном случае это способ выбор альтернатив, описанных на качественном уровне) не имеет четкого описания.

Основная идея метода ЗАПРОС заключается в следующем. Для всех альтернатив, представленных векторами критериальных оценок, выявляются все попарные отношения одного из трех типов: превосходства, эквивалентности или несравнимости. Тем самым для всех пар критериев строятся единые порядковые шкалы (ЕПШ). Тогда отношения на совокупности альтернатив можно представить графом, вершины которого соответствуют альтернативам, а дуги – отношениям. Далее к этому графу применяют процедуру разборки графа, результатом чего является общая ЕПШ. При этом если на каком-то этапе разборки графа нельзя выделить недоминируемую критериальную оценку, то в рамках рассматриваемой модели это свидетельствует о противоречии в информации ЛПР.

Дадим некоторые формальные определения.

Ранжированием элементов множества U называют любую их нумерацию, то есть взаимно однозначно отображение $\rho : U \to \{1,...N\}$.

Вершина $a \in U$ называется *первой* для графа G на U, если $\neg\exists b \in U : b \to a$.

Разборка ориентированного графа проводится последовательным выделением первых вершин (доминирумых над другими или несравнимых альтернатив) из последовательно сужающихся подмножеств вершин.

Пусть x_1 – (некоторая) первая вершина орграфа G; положим $\rho(x_1) = 1$; далее индуктивно: пусть x_k – первая вершина подграфа $G_V, V = U \setminus \{x_1 ... x_{k-1}\}$; положим $\rho(x_k) = k$, и т.д. Полученная последовательность вершин $< x_1,..., x_k >$ называется *приоритетным упорядочением множества* U.

Тогда реализуемость процедуры разборки графа эквивалентна существованию приоритетного упорядочения элементов множества U. А для того, чтобы процедура разборки графа была реализуема, необходимо, чтобы на каждом ее шаге существовала (хотя бы одна) первая вершина. Другими словами, для реализуемости процедуры разборки графа, необходимо и достаточно, чтобы граф был ациклицен.

Распространяя представления об упорядоченности на случай множественных связей, то есть при переходе от графов G связей $a \to b (a, b \in U)$ к гиперграфам H связей $A \to b (A \in A_b, b \in U)$, где через A_b обозначается семейство всевозможных подмножеств элементов A, находящихся в отношении "гипердоминирования" с элементом b. Понятие "первая вершина" естественным образом переносится с графов на гиперграфы. А именно, вершину $a \in U$ называют *первой в гиперграфе* H на U, если $\neg\exists A \subset U : A \to a (A \in A_b)$.

Чтобы представить возможность последовательного выделения первых элементов и тем самым – возможность аналога "приоритетного упорядочения" заданной гиперграфовой структуры, определим, что понимается под сужением гиперграфовой структуры на подмножество V множества U, если исходная гиперграфовая структура H на множестве U задана связями $A \to u, A \in A_u, u \in U$. Определим *сужение* гиперграфа H на множество $V \subset U$ следующим образом: связи в H_V имеют вид $A \to v, v \in U, A \in A_v \cap 2^V$ (то есть из семейства A_v оставляются только те подмножества A, которые целиком лежат в V). Тогда на основе такого определения сужения гиперграфовой структуры определение первых элементов в подмножествах $V \subset U$ будет следующим: элемент-вершина $v \in V$ называется *первым* на подмножестве $V \subset U$ для гиперграфовой структуры H_V, если в V не найдется множества $A \subseteq V$ такого что $A \in A_v$. Аналогично случаю ориентированного графа, можно описать *процедуру разборки гиперграфа* H. Сначала в качестве x_1 берется первый элемент H на U;... в качестве x_k берется первый элемент H на $U \setminus \{x_1,...,x_{k-1}\}$, и т.д. Получаемую последовательность $< x_1,...,x_k >$ также называют *приоритетным упорядочением множества* U. Наконец, аналогом цикла ориентированного графа является *гиперцикл*: непустое множество элементов $C \subseteq U$ называется *гиперциклическим* для гиперграфа H, если для каждого $x \in C$ существует $A \subseteq C$ такое что $A \to x, A \in A_x$. Иначе говоря, гиперциклическое множество C -это, по определению, множество, на котором соответствующий подгиперграф H_C гиперграфа H не дает первых элементов. Доказано [3], что для того, чтобы процедура разборки гиперграфа H на множестве U была реализуема, необходимо и достаточно, чтобы U не содержало гиперциклического множества элементов для H.

Заключение

Как известно, решающее правило метода ЗАПРОС предполагает выполнение условия независимости сравнения альтернатив от изменения опорной ситуации. В случае же гиперграфовой модели это довольно сильное ограничение может быть значительно ослаблено, поскольку сама структура гиперграфа неявно включает логические связки между оценками альтернатив. Предполагается, что использование гиперграфовых структур может оказаться полезным и в случаях, когда сравниваемые альтернативы оцениваются не по одному, а по разным наборам критериев.

Использование гиперотношений – достаточно новый подход, и для того, чтобы максимально использовать его возможности при моделирования задач выбора и принятия решений, требуется более детальное изучение подобного рода структур.

Список литературы

1. Айзерман М.А., Алескеров Ф.Т. Выбор вариантов: основы теории. -М.: Наука, 1990.
2. Ларичев О.И. Теория и методы принятия решений. М.: Логос, 2002
3. Малишевский А.В. Качественные модели в теории сложных систем. -М.: Наука: Физматлит, 1998. -227 с.
4. Малишевский А.В.Разработка и исследование метода множественных свойств и отношений в качественной теории принятия решений: Автореферат диссертации на соискание ученой степени д-ра физ.-мат.наук: 01.01.11. -М., 1995.
5. Martel J.M., L'aide multicritère à la decision: metchodes et application CORS-SCIRO, 1999

Информация об авторе

М.В. Масалитина - Российский Университет дружбы народов, Россия, 117923, г. Москва, ул. Орджоникидзе, д. 3, Тел.: (095)955-07-20, e-mail: mmassal@sci.pfu.edu.ru

MODEL OF ACTIVE STRUCTURAL MONITORING AND DECISION-MAKING FOR DYNAMIC IDENTIFICATION OF BUILDINGS, MONUMENTS AND ENGINEERING FACILITIES.

S. V. Mostovoi, A.E. Gui, V. S. Mostovoi and A. E. Osadchuk

***Abstract**: Structural monitoring and dynamic identification of the manmade and natural hazard objects is under consideration. Math model of testing object by set of weak stationary dynamic actions is offered. The response of structures to the set of signals is under processing for getting important information about object condition in high frequency band. Making decision procedure into active monitoring system is discussed as well. As an example the monitoring outcome of pillar-type monument is given.*

***Keywords**: math model of structural active monitoring, set of weak stationary dynamic actions.*

Introduction

Structural monitoring and dynamic identification are extremely important topics in the maintenance of civil engineering structures in general, but especially in the case when the objects are dangerous from ecological angle of view or very valuable from historical point of view as example monuments. Taken into account that technical systems are damaged by overloading, fatigue, aging and environmental influences the method of dynamic identification provide the possibility to investigate the dynamic behavior of a given structure by means of non-sensitive for constructive tests, and consequently enable to assess the structure “health” and the possible need for structural maintenance. For such a goal there is proposed math model of structural active monitoring, which was realized into the automatic monitoring system. The power of a test signal should not be high. At least after testing the environment should not be essentially changed. The following experimentation has to correspond to these conditions: there is a set of weak signals, energy of each one is commensurable with a background noise level, they are dispatched and followed strictly periodically in time in encoded shape, and the response of an object is accrued in correspondent to to this periodicity. Here is used a black box model of the tasted system, when for analyses is used only the system response on outside influence. But the above mentioned experimentation has some imperfection. Instability of test signals source leads to the accentual accumulated signal disfigure.

The real signal sources ensure stability only in some domain of modification parameters of these signals. In this sense it is possible only to state, that the vector of parameters, which defined testing signal, is random with a priori known distribution. This circumstance can be considered as a possibility to research a priori statistic of parameter stability of a source, or as a possibility formally entered into the model of the researcher's heuristics. Further we shall show, that the drift in time leads to "fading" a high-frequency component in a response function of structure, while the fluctuations of testing signal energy are not so essential. In work [1] the model of a uniform distribution of the start moment is fragmentary reviewed. Under consideration is more global model and we shall construct algorithm for a signal restoring. A set of physically feasible signals with fluctuating amplitude and initiation delay can model the active monitoring. These models deserve attention, as the active monitoring is a regime testing of a structure by a testing signal, on response which one can make a conclusion about its state, and about possible modifications in its state. Term "regime" means, first of all, experimentation in time. As soon as absolutely precise temporal experimentation we can not provide then it is necessary for us to model stochastic character of this process. The model of a binomial flow that embodies determined component of testing process and stochastic character of parameter instability of this flow is further given. The paper introduces into this solution problem and a methodology for the identification of the main structural parameters (as a time history of the structure response function, in order to identify the main eigenfrequencies and associated modal shapes of the structure and so on) by means of an active monitoring. The response of structures to weak stationary dynamic actions is under consideration. There are proposed practical examples for illustration the main aspects of theory.

The mathematical model of active monitoring.

In this formula the symbol $*$ indicates the operator of convolution, t_d - transport delay in concrete experiment, i.e. time from start of a signal before its receipt on a receiver recorder T - the time between sending of signals and is simultaneous the value defining length of the signal carrier, i.e. time, when the signal is distinct from zero point. This value should be less than T. Explicitly this model is considered in treatise [1].

In this treatise the processing algorithm of observations permitting to augment a signal - noise ratio is offered and was shown, that at uniform distribution of the moments instability of sounding start the high-frequency components of a signal is "disappear". In this treatise we shall consider more general case of binomial flow with beta distributed partial density and we shall offer algorithm of a signal restoring

Let's consider that the partial density function of the moment of a k signal start $e_k(\tau)$ looks like:

$$e_k(\tau) = e_k(\tau + (k-1)\cdot T) = e(\tau_k);\ \tau \in (0,\varepsilon);\ k = 1,2,...,K. \tag{2}$$

It means, that the form of distribution is identical to all moments of the introduction, and for the k signal the shift on an interval is carried out $\tau + (k-1)\cdot T$

With allowance for of transport delay t_d partial density of the moment of a k signal reception will be:

$$e_k(\tau) = e_k(\tau + (k-1)\cdot T + t_d) = e(\tau_k);\ \tau \in (0,\varepsilon);\ k = 1,2,...,K. \tag{3}$$

And hereinafter (3) under τ_k is realized $\tau_k = \tau + (k-1)\cdot T + t_d$

Let's consider a case of not intersected partial density, i.e. case when $T > \varepsilon$, and completely solved i.e. with not intersected carriers, signals.

These two assumptions mean, that the carrier of a signal is less than $T - \varepsilon$ and that the probability density $\pi(\tau_1,...,\tau_K,\Omega)$ of appearance in the observations domain $\Omega = K\cdot T$ precisely K signals with the moments $\tau_1,...,\tau_K$ is introduce the multiplication of partial density and looks like**:**

$$\pi(\tau_1,...,\tau_K,\Omega) = \prod_{k=1}^{K} e_k(\tau_k) \tag{4}$$

Two of these assumptions completely correspond to conditions of experiment realization .

Fluctuations θ_k is accepted independent, with density functions, $p_k(\theta)$, $\theta_k \in \Theta$.

The independence means, that the probability density of a vector parameters $\{\ \theta_1,...,\theta_K\ \}$ looks like:

$$p(\theta_1,...,\theta_K) = \prod_{k=1}^{K} p_k(\theta_k) \tag{5}$$

In our model is accepted, that the exploring signal $u(t)$ is physically feasible

After convolution with a transfer function of the environment $h(t)$ the signal $S(t)$ will be with the final carrier of length no more than $T - \varepsilon$, i.e. two conditions are executed: causalities (6) and stability (7)

$$S_k(t) = \begin{cases} \theta_k \cdot S(t-\tau_k), & t \in (\tau_k, \tau_k + \alpha T); \\ 0, & t \notin (\tau_k, \tau_k + \alpha T),\quad \alpha T < T - \varepsilon\ u\ \alpha \in (0,1) \end{cases} \tag{6}$$

$$\int_0^{\alpha T} S^2(t)dt < \infty \tag{7}$$

The last condition can be lead to following by normalization:

$$\int_0^{\alpha T} S^2(t)dt = 1 \tag{7a}$$

With normalization's allowance (7a) in distribution of fluctuations (5) for expectation $E\{\theta_k\}$ can be accepted, that

$$E\{\theta_k\} = 1. \tag{8}$$

The expectation of the environment response on flow of separately fluctuated signals with allowance for that $E\{n(t)\} = 0$ will look like

$$E\{y(t)\} = \int_{\Omega}\ldots\int_{\Omega}\int_{\Theta}\ldots\int_{\Theta}\sum_{k=0}^{K}\theta_k S(t-\tau_k)\cdot\prod_{k=1}^{K}e_k(\tau_k,\gamma,\eta,\varepsilon)\cdot p_k(\theta_k)\cdot d\tau_k\cdot d\theta_k + E\{n(t)\} =$$

$$= \sum_{k=0}^{K}\int_{\Theta}\theta_k\cdot p_k(\theta_k)\cdot d\theta_k\cdot\int_{\Omega}S(t-\tau_k)\cdot e_k(\tau_k,\gamma,\eta,\varepsilon)\cdot d\tau_k \tag{9}$$

Taking into account that the first integral, standing in the last expression in the sum sign, is expectation of fluctuation of each flow K signals and taking into account (8) last expressions becomes:

$$E\{y(t)\} = \sum_{k=0}^{K}\int_{\Omega}S(t-\tau_k)\cdot e_k(\tau_k,\gamma,\eta,\varepsilon)\cdot d\tau_k \tag{10}$$

In our case for quite satisfactory approximating for partial density of the moments of the signals introduction can be beta distributions with parameters γ,η,ε. Varying these parameters it is possible to receive approximating practically of any distribution on an interval of length [3].

$$e_k(\tau_k,\gamma,\eta,\varepsilon) = \begin{cases} \frac{1}{\varepsilon}\cdot\frac{\Gamma(\gamma+\eta)}{\Gamma(\gamma)\Gamma(\eta)}\cdot\left(\frac{\tau_k}{\varepsilon}-(k-1)\cdot T\right)^{\gamma-1}\cdot\left(1-\frac{\tau_k}{\varepsilon}-(k-1)\cdot T\right)^{\eta-1}; \tau_k\in\Delta_k; \\ 0, \text{ когда } \tau_k\notin\Delta_k. \quad 0<\gamma,0<\eta; \quad k=1,2,\ldots,K. \quad \Delta_k=((k-1)\cdot T,(k-1)\cdot T+\varepsilon) \end{cases} \tag{11}$$

The expectation for model (1) with allowance for (2) - (11) becomes:

$$E\{y(t)\} = \sum_{k=0}^{K}\frac{1}{\varepsilon}\cdot\frac{\Gamma(\gamma+\eta)}{\Gamma(\gamma)\Gamma(\eta)}\cdot\int_{(k-1)\cdot T}^{(k-1)\cdot T+\varepsilon}\left[S(t-\tau_k)\right]\cdot\left(\frac{\tau_k}{\varepsilon}-(k-1)\cdot T\right)^{\gamma-1}\cdot$$

$$\cdot\left(1-\frac{\tau_k}{\varepsilon}-(k-1)\cdot T\right)^{\eta-1}d\tau_k. \tag{12}$$

Condition of physical feasibility (causality (6) and the stability (7)) allows to present a signal as following series:

$$S(t-\tau_k) = X(t,\tau_k,\alpha\cdot T)\cdot\sum_{i=1}^{\infty}s_i\cdot\varphi_i(t-\tau_k) \tag{13}$$

Characteristic interval function (14) and orthonormalized on $(0,\alpha\cdot T)$ basis $\varphi(t)=\{\varphi_i(t)\}$.

$$X(t,\tau_k,\alpha\cdot T) = \{1, \text{npu } t\in(\tau_k,\tau_k+\alpha\cdot T);\ 0, \text{npu } t\notin(\tau_k,\tau_k+\alpha\cdot T)\} \tag{14}$$

Than

$$E\{y(t)\} = \frac{1}{\varepsilon}\cdot\frac{\Gamma(\gamma+\eta)}{\Gamma(\gamma)\Gamma(\eta)}\cdot\sum_{k=0}^{K}\left[\sum_{i=1}^{\infty}s_i\cdot\int_{\Omega}X(t,\tau_k,\alpha\cdot T)\ \varphi_i(t-\tau_k)\cdot\left(\frac{\tau}{\varepsilon}-(k-1)\cdot T\right)^{\gamma-1}\cdot\right.$$

$$\cdot\left(1-\frac{\tau}{\varepsilon}-(k-1)\cdot T\right)^{\eta-1} d\tau_k =$$

$$=\frac{1}{\varepsilon}\cdot\frac{\Gamma(\gamma+\eta)}{\Gamma(\gamma)\Gamma(\eta)}\cdot\sum_{k=0}^{K}\left[\sum_{i=1}^{\infty} s_i\cdot\int_{(k-1)\cdot T}^{(k-1)\cdot T+\varepsilon} {}_i(t-\tau_k)\right]\cdot\left(\frac{\tau}{\varepsilon}-(k-1)\cdot T\right)^{\gamma-1}\cdot$$

$$\cdot\left(1-\frac{\tau}{\varepsilon}-(k-1)\cdot T\right)^{\eta-1} d\tau_k. \tag{15}$$

If to realize shift of each signal on $(k-1)\cdot T$ and then to apply $L_K[y(t)]$ - operator of K aliquot shift and summation of results of shift with factor $1/K$, we shall receive following result.

$$L_K\{y(t)\}=\overline{E\{S(t)\}}=\frac{1}{\varepsilon}\cdot\frac{\Gamma(\gamma+\eta)}{\Gamma(\gamma)\Gamma(\eta)}\cdot\sum_{i=1}^{\infty} s_i\cdot\int_0^{\varepsilon}\varphi_i(t-\tau)\cdot\left(\frac{\tau}{\varepsilon}\right)^{\gamma-1}\cdot\left(1-\frac{\tau}{\varepsilon}\right)^{\eta-1} d\tau \tag{16}$$

Here $\overline{E\{S(t)\}}$ is estimation of a signal expectation.

$$S(t)\cong s(t)=X(t,0,\alpha\cdot T)\cdot\sum_{j=1}^{Q} s_{i_j}\varphi_{i_j}(t)=X(t,0,\alpha\cdot T)\cdot(\mathbf{f}(t),\mathbf{s}) \tag{17}$$

From (16) follows, that the signal is approximately recover by calculation of a vector $\mathbf{s}$. We shall make dot product of a vector $\mathbf{s}$ on the function $\mathbf{f}(t)$ we shall receive estimation (17).

Let:

$$\varphi_j(t)=\frac{1}{\varepsilon}\cdot\frac{\Gamma(\gamma+\eta)}{\Gamma(\gamma)\Gamma(\eta)}\cdot\int_0^{\varepsilon} {}_j(t-\tau)\cdot\left(\frac{\tau}{\varepsilon}\right)^{\gamma-1}\cdot\left(1-\frac{\tau}{\varepsilon}\right)^{\eta-1} d\tau \tag{18}$$

Than

$$L_K\{y(t)\}=X(t,0,\alpha\cdot T+\varepsilon)\cdot\sum_{j=1}^{Q} s_{i_j}\cdot\phi_j(t)=X(t,0,\alpha\cdot T+\varepsilon)\cdot\mathbf{s}^T\cdot\mathbf{F}(t), \tag{19}$$

$$\mathbf{F}(t)=\{\phi_j(t)\};\ j=\overline{1,Q}$$

Vector of factors $\mathbf{s}$, we shall determine from the last expression as follows.

$$\int_0^{\alpha\cdot T+\varepsilon} L_K\{y(t)\}\cdot\mathbf{g}^T(t)\,dt=\int_0^{\alpha\cdot T+\varepsilon} X(t,0,T+\varepsilon)\cdot\mathbf{s}^T\cdot\mathbf{F}(t)\cdot\mathbf{g}^T(t)\,dt=\mathbf{s}^T\cdot\int_0^{\alpha\cdot T+\varepsilon} X(t,0,T+\varepsilon)\cdot\mathbf{U}(t)dt. \tag{20}$$

$\mathbf{g}^T(t)=\{g_j(t)\};\ j=\overline{1,Q}$ - Is the vector function composed from a subset Q of functions of orthonormalized on $(0,\alpha\cdot T+\varepsilon)$ basis.

$$\mathbf{U}(t)=\{u_{ij}(t)\};i,j=\overline{1,Q}\ .u_{ij}(t)=(\phi_j(t),g_i(t)) \tag{21}$$

It is a matrix, elements by which one are the dot products of basic functions $\phi_j(t)$ and $g_i(t)$

Let's designate

$$\mathbf{I}^T=\int_0^{\alpha\cdot T+\varepsilon} L_K\{E\{y(t)\}\}\cdot\mathbf{g}^T(t)\,dt \tag{22}$$

We transpose the left and right parts of expression (20). We receive a set of equations concerning a vector $\mathbf{s}$:

$$\mathbf{U}^T \cdot \mathbf{s} = \mathbf{I} \tag{23}$$

The algorithm of restoring of a signal in a series from K tests in binomial flow is obtained, using prior regards about distribution of the unstable moment of start of an exploring signal. The key moment here is the prior knowledge of the value ε, as it defines a choice of basis $\mathbf{g}(t)$.

Processing algorithm. The result of experiment $y(t)$ exposed to transformation $L_K\{y(t)\}$ Then the vector of scalar multiplication is created $(L_K\{y(t)\}, g_i(t))$ and matrix $u_{ij}(t) = (\phi_j(t), g_i(t))$. The vector $\mathbf{s}$ is discovered from the equation (23). Substitute this vector in (17) and we receive a signal estimation.

Example. Such an approach for solution of the engineering problems in creation of high-rise extended pillar type monument in Kyiv was used [4].

With the purpose to get the monument spectral characteristics, logarithmic decrement of the oscillations of the object and to analyses of damping ability of the system, which was realized at the monument for oscillation reduction, the site tests were carried out. For registration of fluctuations three-directional geophone with gauges located on three mutually perpendicular axes was used. The special characteristics of gauges represent one-modal curve with the extreme point in f=l Hz. Geophones were placed at a horizontal surface, on the level of 42 meters. They served as a part of interface of the monitoring registration and processing automated system. This system allows correcting the spectral characteristic up to uniform in the chosen range of frequencies. The first part of experiment consisted in registration of monument reaction on a natural background as an input signal. This signal represents a superposition of the large number of the external factors from natural microseism noise and men made one up to signals from ground transport. The important moment is that the total spectrum of this signals is much wider then the response spectrum of the monument. For the monument it was obtained three modes on frequencies 0.48 Hz, 0.93 Hz and 1.47 Hz with corresponding amplitudes 1.0, 0.07 and 0.12. The frequency of 1.47 Hz with rather intensive amplitudes hypothetically is devoted to the mode of the top sculpture, the framework of which is less rigid then the framework of the self column. The second part of experiment was consisted in to get a logarithmic decrement of oscillation of the monument on the basic resonant frequency. For this purpose was used a damp of pendulum type. By compulsory swinging of this pendulum the monument was coupled in fluctuations and then the fluctuations faded by a natural way. The average value estimation of the logarithmic decrement of the oscillations was equaled 0.055. This figure shows that the metal column with granite shell has rather low capacity to dampen fluctuations. The damper, when it was put in operation during the tests, has increased the ratio of the logarithmic decrement of the oscillations up to the level was equaled 0.18-0.25. The damper construction gives the possibility to obtain greater ratio of logarithmic decrement of the oscillations via increasing of the friction coefficient the energy absorber. It's necessary to note that the spectrum of a structure is its steady characteristic. This function varies with change of mechanical parameters of a structure and can be used for detection of "age" changes of a structure while in exploitation. It's possible to consider that the fixed spectral monument characteristics further can be used as reference for detection of a beginning of the moment "age" changes during a structure-monitoring period.

Conclusion

Here is proposed and analyzed math model of an active monitoring system which is based on stochastic flow process of accruing data. For response signal correction is used premature probability of instability parameters of testing signals set generator. It is shown that the main source of instability testing signals is the time of signal departure and decision-making procedure is proposed.

Bibliography

A.E. Gay, S.V. Mostovoi, V.S. Mostovoi, A.E. Osadchuk. Model and Experimental Studies of the Identification of Oil/Gas Deposits, Using Dynamic Parameters of Active Seismic Monitoring, Geophys. J., 2001, Vol. 20, pp. 895-9009.

K. Bolshakov, Statisticheskoe Vydelenie Potoka Signalov i'z Shuma (Statistical Identification of a Series of Signals from Noise) (Moscow: Sov. Radio: 1969) (in Russian).

Feller W. An introduction to probability theory and its applications. Vol. 2. John Willey & Sons, Inc. New York. 1971. 751 c.

Kondra M., Lebedich I., Mostovoi S. Pavlovsky R., Rogozenko V. Modern approaches to assurance of dynamic stability of the pillar type monument with an application of the wind tunnel assisted research and the site measuring of the dynamic characteristics. Eurodyn 2002, Swets & Zeitlinger, Lisse, 2002, p. 1511 - 1515.

Authors information

Sergey V. Mostovoi - Institute of Geophysics of the National Academy of Sciences, Kiev, Ukraine. E-mail: smost@i.com.ua; most@igph.kiev.ua

Angela E.Guy - Institute of Geophysics of the National Academy of Sciences, Kiev, Ukraine. E-mail: most@igph.kiev.ua

Vasiliy S. Mostovoi - Institute of Geophysics of the National Academy of Sciences, Kiev, Ukraine. E-mail: most@igph.kiev.ua

Ascold E. Osadchuk - Institute of Geophysics of the National Academy of Sciences, Kiev, Ukraine. E-mail: most@igph.kiev.ua

ИНТЕЛЛЕКТУАЛЬНЫЕ РЕШЕНИЯ ПРИ УПРАВЛЕНИИ ДВИЖЕНИЕМ ГОРОДСКОГО ЭЛЕКТРОТРАНСПОРТА

М. А. Столяренко

Аннотация: *В данной работе приводится описание структуры и основ функционирования диспетчерских служб и интеллектуальной системы принятия решений при управлении движением городского электротранспорта.*

Ключевые слова: *электротранспорт, система естественного интеллекта, система искусственного интеллекта*

Задачи, возникающие при управлении состоянием городского электротранспорта

Первоочередной задачей диспетчерских служб является обеспечение бесперебойной перевозки пассажиров. В силу специфики функционирования городского электротранспорта эта задача распадается на следующие подзадачи:

1. Обеспечение достаточного количества подвижного состава на линиях маршрутов.
2. Обеспечение надежности функционирования подвижного состава, контактной сети и оборудования тяговых подстанций.
3. Принятие решения по ликвидации аварии при ее возникновении.

Решением первой задачи занимается *служба движения*. Второй и третьей проблемой занимается *энергодиспетчер* и *служба электрохозяйства*. Основную проблему при управлении состоянием городского электротранспорта включает в себя третья задача, поэтому её и рассмотрим.

Анализ явлений, происходящих в системе электроснабжения

Аварийные ситуации, происходящие в системе питания городского электротранспорта, можно классифицировать по месту возникновения на следующие четыре типа:

1. Аварии, происходящие в вагонах подвижного состава.
2. Аварии, происходящие на контактной сети.
3. Аварии, происходящие в питающих и отсасывающих кабелях.
4. Аварии, происходящие на тяговой подстанции.

Первый и второй типы аварий обусловлены износом подвижного состава и контактного провода и ликвидируются путём выезда оперативной бригады на место происшествия.

Третий и четвёртый типы обусловлены перегрузками оборудования ТПС и питающих кабелей. В соответствии со структурной схемой электроснабжения, детально рассмотренного в [1], ликвидация последних двух типов аварий возможна в автоматическом режиме, путем подключения резервного питания к обесточенному участку.

Если это невозможно, то необходимо замыкание секционов для питания аварийного участка от соседнего (рис.1).

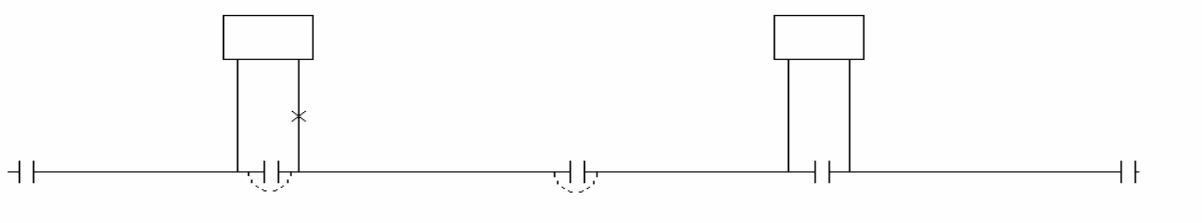


Рис.1. Восстановление электроснабжения аварийного участка путём замыкания секциона.

В этом случае присоединение тяговой подстанции, питающее удлинённый участок, работает в режиме перегрузки. При таком питании аварийного участка для обеспечения надёжной работы системы электроснабжения выдвигаются следующие ограничения на:

1. Плотность тока в контактном проводе, а также в питающих и отсасывающих кабелях.

2. Падение напряжения на конце питаемого участка.

Выполнение первого ограничения позволяет снизить риск выхода из строя контактного провода и питающих кабелей. На практике максимально допустимая плотность тока составляет 5 А/мм^2.

Второе ограничение связано с обеспечением нормального функционирования подвижного состава на конце питаемого участка. При значительной нагрузке на длинном участке значение напряжения в наиболее отдалённых от питающей подстанции местах может опускаться до таких величин, при которых невозможна нормальная работа двигателей вагонов. На практике падение напряжения на концах участка в нормальном режиме работы, согласно [2], не должно превышать 90 В, а в аварийном режиме – 170 В.

При включении присоединения подстанции в аварийный режим работы на нем необходимо поменять уставки линейной и токовременой защит. Невыполнение этого требования приведет к частому отключению быстродействующего автомата под действием увеличившихся нагрузок, что повлечет за собой сбой движения вагонов.

Линейные автоматы, на которых выполнены устройства линейной защиты, токовые реле и реле времени, на которых собраны схемы токовременной защиты, также имеют пределы регулировок. Это обуславливает еще одно условие на выбор варианта резервирования аварийного участка, которое можно сформулировать следующим образом:

3. Значения уставок защит на присоединении тяговой подстанции, устанавливаемые в случае включения его в аварийном режиме, не должны превышать максимально возможные на аппаратуре защиты присоединения.

Если подходящего варианта питания не найдено, то применяются следующие способы корректировки нагрузки на питаемом участке:

1. Сокращение подвижного состава на маршрутах, проходящих через аварийный участок
2. Питание аварийного участка одновременно от нескольких (в основном, двух) присоединений или подстанций.
3. Перенос секционов для уменьшения длины аварийного участка.

Второй и третий способы на практике применяются достаточно редко. Это объясняется следующими причинами.

При двустороннем питании аварийного участка резко возрастает опасность несрабатывания защиты на тяговых подстанциях. В случае значительного расстояния между присоединениями, питающими участок,

при возникновении короткого замыкания возле одного из них не срабатывают устройства защиты на отдаленном присоединении. В результате место аварии находится под напряжением, что может привести к крайне нежелательным последствиям.

Причина редкого использования третьего способа корректировки нагрузки заключается в достаточно больших затратах труда и времени на перенос секциона.

Таким образом, наиболее эффективным способом корректировки нагрузки на участок является сокращение подвижного состава. Но с другой стороны, всякое изменение количества подвижного состава энергодиспетчер вынужден согласовывать со службой движения. По этой причине сокращение применяется как крайняя мера и в минимальных количествах.

Особую проблему составляет принятие решения по восстановлению энергоснабжения при выходе из строя присоединения, включенного в аварийном режиме. При этом замыкание секциона, граничащего с этим участком, как правило, приводит к превышению предельно допустимых условий системы электроснабжения. В этом случае аварийной участок разделяется на части путём размыкания секционов, и каждая часть подключается к отдельному присоединению тяговой подстанции. Таким образом, в этом случае в аварийном режиме функционируют несколько присоединений соседних подстанций (рис.2).

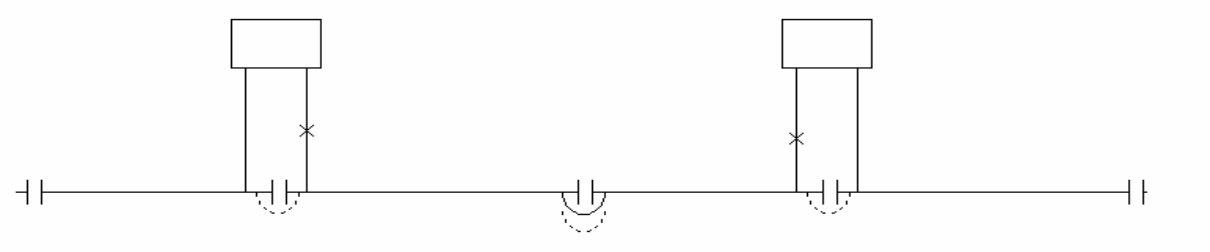


Рис.2. Восстановление электроснабжения при выходе из строя присоединения, включённого в аварийном режиме.

Анализ функционирования системы естественного интеллекта управления состоянием городского электротранспорта

Информация об аварии поступает энергодиспетчеру от *дежурного персонала* по телефону или через средства телемеханики, которыми оборудовано большинство тяговых подстанций, а также от центрального диспетчера, который связан с *линейными диспетчерами*, находящимися на конечных остановках маршрутов (рис.3).

Решение по ликвидации аварий принимает энергодиспетчер, согласовывая свои действия с *главным инженером* службы электрохозяйства. Накопленный опыт отразился в рекомендациях принятия решений при возникновении аварийной ситуации. Но эти рекомендации не являются универсальными и не учитывают следующие факторы:

1. Изменение количества подвижного состава на маршрутах.
2. Сезонные изменения температуры окружающей среды и пассажиропотока.

По этим причинам каждое переключение подстанций сопровождается расчетом токов нагрузки, короткого замыкания, падений напряжения и рекомендуемых уставок защит при новом варианте включения присоединений подстанций. Эти расчёты производит *инженер по расчетам*. На основе этих данных и принимается решение о целесообразности принимаемого решения. При этом в качестве эксперта выступает главный инженер службы электрохозяйства, который может пренебречь незначительным превышением некоторых ограничений для обеспечения работоспособности всей системы. Окончательное решение выдаётся дежурному персоналу, который приводит его в исполнение. В случае необходимости выдаётся наряд работникам *ремонтных бригад контактной сети* на выезд на место аварии. Восстановлением работоспособности тяговых подстанций занимается *бригада по ремонту и обслуживанию оборудования тяговых подстанций,* за устройства линейной защиты ответственна *бригада по наладке и испытаниям электрооборудования тяговых подстанций*.

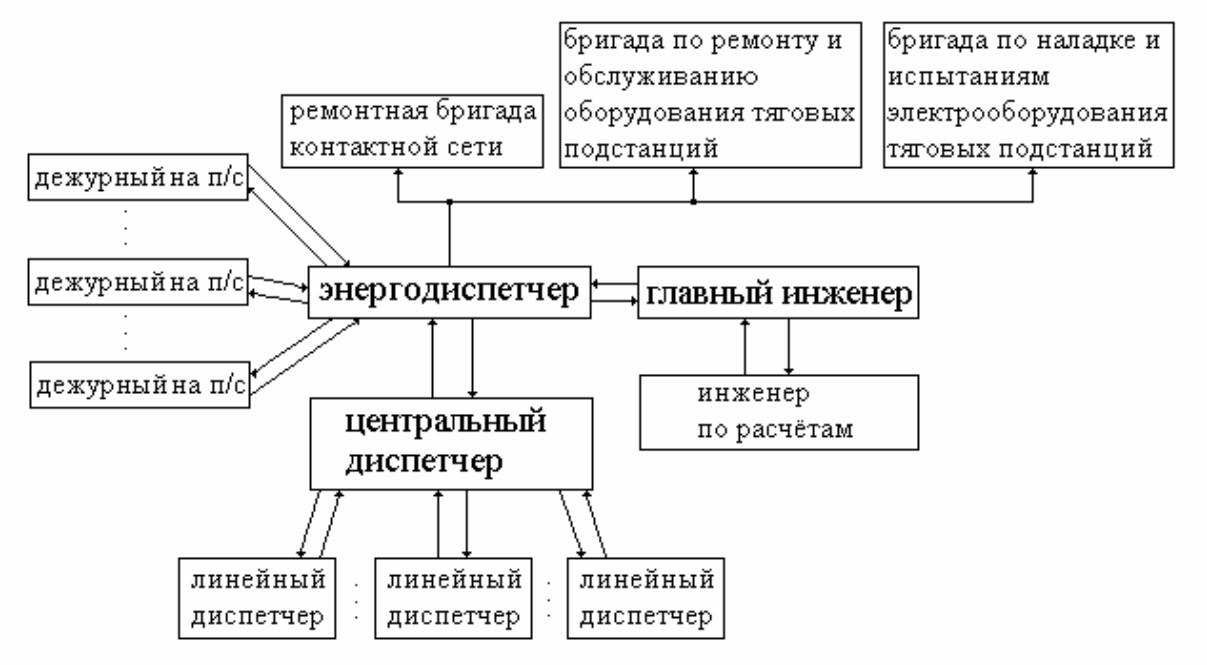


Рис.3. Структурная схема системы естественного интеллекта управления состоянием городского электротранспорта.

Обозначим составные блоки системы естественного интеллекта множествами. Пусть

D – множество дежурных на подстанции,

EDIS – множество энергодиспетчеров,

CDIS – множество центральных диспетчеров,

LDIS – множество линейных диспетчеров,

GL_I – множество, соответствующее главному инженеру,

I_R – множество, соответствующее инженеру по расчётам,

RBKS – множество работников ремонтных бригад контактной сети,

BROOTP – множество работников бригад по ремонту и обслуживанию оборудования тяговых подстанций,

BNIETP – множество работников бригады по наладке и испытаниям электрооборудования тяговых подстанций.

Таким образом, действие, выполняемое при ликвидации аварии определяется следующим набором:

$$\{D, EDIS, CDIS, LDIS, GL\_I, I\_R, RBKS, BROOTP, BNIETP\} \quad (1)$$

При этом каждый элемент набора (1) может допустить ошибку, определяемую человеческим фактором. В результате вероятность ошибки может достигать нескольких процентов, что недопустимо для управления системой такого уровня.

Анализ функционирования системы искусственного интеллекта управления состоянием городского электротранспорта

Интеллектуальной моделью действий энергодиспетчера служит экспертная система принятия решений при управлении состоянием городского электротранспорта (рис.4).

Она включает в себя следующие блоки:

1. Базу знаний **БЗ**, содержащую информацию о структуре контактной сети, характеристиках подстанций и подвижного состава, формализованных согласно [1].
2. Блок расчета **БР** токов нагрузки, короткого замыкания и уставок защиты согласно методологии, предложенной в [3].

3. Блок выбора, оценки и принятия решений **БПР**.

При возникновении аварийной ситуации информация поступает на вход блока БПР. На основе анализа структуры питающей сети из базы знаний выбираются все возможные варианты восстановления энергоснабжения обесточенного участка. При этом для каждого варианта просчитываются электрические характеристики каждого присоединения, подключаемого для ликвидации аварии.

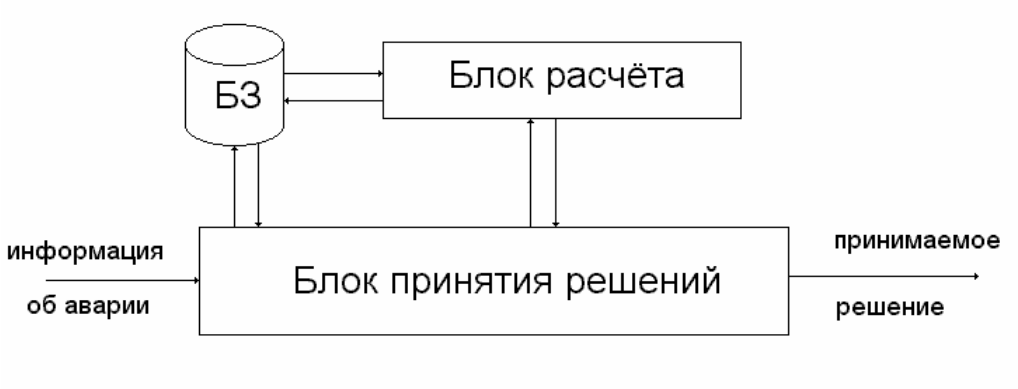


Рис.4. Структурная схема экспертной системы принятия решений
при управлении состоянием городского электротранспорта.

Ограничения на предельно допустимые значения рассматриваются, как нечеткие числа. Для каждого варианта восстановления питания строится функционал превышения предельно допустимых значений. Он имеет следующий вид:

$$I = \sum_j k_j \varphi_j \qquad (2)$$

где k_j – весовой коэффициент превышения предельного значения j-й характеристики, φ_j – величина превышения j-й характеристики предельного значения, которая определяется следующим образом:

$$\varphi_j = \begin{cases} \dfrac{a_j - a_j^{np}}{a_j^{np}}, \; при \; a_j > a_j^{np} \\ 0, \; при \; a_j < a_j^{np} \end{cases} \qquad (3)$$

где a_j и a_j^{np} – расчётное и предельно допустимое значение j-й величины.

Минимизируя функционал (2), БПР находит наиболее приемлемый вариант восстановления электроснабжения. Если таких вариантов оказывается несколько, то все приемлемые решения выдаются энергодиспетчеру, который на основе субъективных соображений (таких, как очередность выполнения ППР на подстанциях, техническая трудность замыкания и размыкания отдельных секционов и т.д.) принимает окончательное решение.

Заключение

Таким образом, интеллектуальная система может облегчить работу энергодиспетчера при ликвидации аварийной ситуации и повысить надёжность работы электрооборудования. Но полностью заменить работу человека в данном случае невозможно, т.к. экспертная система не учитывает ряд субъективных факторов, которые не возможно формализовать. Тем не менее, в данном случае совместная работа человека и вычислительной машины позволяет увеличить эффективность его труда и снизить риск принятия ошибочного решения.

Библиография

1. Столяренко М.А. Ситуационный прогноз и распознавание решений в управлении электроснабжением городского транспорта. / Искусственный интеллект. 2002. №1 с. 111-118

2. Тяговые подстанции трамвая и троллейбуса: Справочник / Под ред. И. С. Ефремова. – М.:Транспорт, 1984 – 311с.
3. Криводубский О. А., Столяренко М. А. Математическая модель питающей сети городского электротранспорта. - Сборник «Научные труды Донецкого государственного технического университета». Донецк, 2002 г. с. 223-227

Авторская информация

Столяренко Максим Александрович – аспирант Донецкого государственного института искусственного интеллекта, ул. Розы Люксембург, д.12, к. 1001, г. Донецк-55, 83055, Украина, e-mail: stolyarenko@rambler.ru

CLASSIFICATION-BASED METHOD OF LINEAR MULTICRITERIA OPTIMIZATION

V. Vassilev, K. Genova, M. Vassileva, S. Narula

***Abstract**: The paper describes a classification-based learning-oriented interactive method for solving linear multicriteria optimization problems. The method allows the decision makers, describe their preferences with greater flexibility, accuracy and reliability. The method is realized in an experimental software system supporting the solution of multicriteria optimization problems.*

***Keywords**: linear multicriteria optimization, interactive methods, decision support systems.*

1. Introduction

The problems of multicriteria optimization are multicriteria decision making problems with infinite number of alternatives [Steuer, 1986]. The interactive methods are the most widely spread methods [Gardiner and Vanderpooten, 1997] for solving problems of multicriteria optimization. Every iteration in such method consists of two phases: a computation and a decision one. One or more non-dominated solutions are generated with the help of a scalarizing problem at the computation phase. At the decision phase these non-dominated solutions are presented for evaluation to the decision maker (DM). In case the DM does not approve any of these solutions as a final solution (most preferred solution), he/she supplies information concerning his/her local preferences with the purpose to improve these solutions. This information is used to formulate a new scalarizing problem, which is solved at the next iteration.

The efficiency of each interactive method depends to a great extent on the type of the information, which the DM sets in order to improve the local preferred non-dominated solution, on the time for scalarizing problem solution, on the possibilities to learn the DM with respect to the multicriteria problem being solved, on the type and number of the non-dominated solutions being compared with the local preferred solution.

When solving linear problems of multicriteria optimization, linear programming problems are used as scalarizing problems. They are easy solved problems. Hence, in the interactive methods solving multicriteria linear problems, the time for scalarizing problems solution does not play a significant role. In the development of these methods main attention is paid to the type of information, which the DM can put forward in the attempt to improve the local preferred non-dominated solution. In a large part of the interactive methods recently known, this information comprises basically the aspiration levels of the criteria [Wierzbicki, 1980], which the DM wishes to achieve. The aspiration levels define the so-called reference point in the criteria space. These interactive methods use scalarizing problems belonging to the group of scalarizing problems of the reference point. The classification-oriented scalarizing problems (with some exclusions, like STEP scalarizing problem [Miettinen, 1999]) have been more rarely used in the interactive methods for linear multicriteria problems solving up to nowadays. The

possibilities to learn the DM during the time of linear multicriteria problem solution figure another significant feature of the interactive methods. In addition to DM's freedom to move in the non-dominated space, these capabilities are expressed in the determination of more than one non-dominated solution in the computing phase. These solutions are presented to the DM for evaluation [Korhonen and Laakso, 1986]. It should be noted that in modern interactive methods solving multicriteria linear problems, it is accepted by default, that the DM can evaluate more than two non-dominated solutions without problems. In the comparison and evaluation of more than two non-dominated solutions, especially when the criteria number is large and when the non-dominated solutions do not differ significantly, the DM can meet considerable difficulties in the selection of a local (global) preferred non-dominated solution [Jaszkiewicz and Slowinski, 1995].

On the basis of new classification-oriented scalarizing problems, an interactive method is proposed in the paper, which to a high degree combines the positive aspects of the interactive methods for solution of linear multicriteria optimization problems developed up to now. The basic characteritics of this interactive method are the following:

- possibility to enlarge the information, with the help of which the DM can put down his/her local preferences, setting desired or acceptable directions and intervals of change in the criteria values, in addition to the criteria desired or acceptable levels.

- possibility for comparatively quick learning of the DM concerning the specific multicriteria linear problems solved, which results from representing more non-dominated solutions for evaluation at each iteration, as well as from DM's free movement in the whole area of these solutions;

- comparatively easy evaluation by the DM of the solutions presented, on account of the fact that they are close one to another.

2. Problem formulation

The linear problem of multicriteria optimization (denoted by LMK), can be formulated as follows:

(1) $\text{"max"}\{f_k(x)\}, \quad k \in K$

subject to:

(2) $\sum_{j \in N} a_{ij} x_j \le b_i, \qquad i \in M$

(3) $0 \le x_j \le d_j, \qquad j \in N$

(4) $x_j \ge 0, \qquad j \in N,$

where the symbol "max" means that all the objective functions should be simultaneously maximized.

$K = \{1,2,...,p\}$, $M = \{1,2,...,m\}$, and $N = \{1,2,...,n\}$ are the index sets of the linear criteria (objective functions), of the linear constraints and of the variables (the solutions) respectively;

$f_k(x), \ k \in K$ are linear criteria (objective functions);

$f_k(x) = \sum_{j \in N} c_j^k x_j$;

$x = (x_1, x_2, ..., x_j, ..., x_n)^T$ is the vector of variables (solutions).

The constraints (2)-(4) define the acceptable set of the variables (solutions). This set will be denoted by X.

Several definitions will be introduced for greater precision.

Definition 1. The solution x will be called efficient solution of the problem, if there does not exist another solution $\bar{x}$, such that the following inequalities be satisfied:

$f_k(\bar{x}) \ge f_k(x),$ for each $k \in K$ and

$f_k(\bar{x}) > f_k(x),$ for at least one index

Definition 2. The vector $f(x) = (f_1(x),..., f_p(x))^T$ is called a non-dominated solution of the problem in the criteria space, if x is an efficient solution of the corresponding problem in the variables space.

Definition 3. Desired or acceptable directions of change in the values of some of the criteria are the directions, in which the values of these criteria in the last non-dominated solution obtained, the DM wishes to be improved or agrees to be deteriorated, so that this solution is improved according to his/her local preferences.

Definition 4. Desired or acceptable intervals of change in the values of some of the criteria are the intervals, in which the DM wishes to find the improved or deteriorated values of these criteria with respect to their corresponding values in the last non-dominated solution obtained.

The problems of multicriteria optimization do not possess an optimal solution. Hence, it is necessary to select such a solution among the non-dominated solutions, which suits best the DM's global preferences. This choice is personal and it depends entirely on the DM.

3. Classification-oriented scalarizing problems of desired or acceptable levels, directions or intervals

The classification-oriented scalarizing problems decrease the requirements towards the DM when comparing and evaluating the new solutions obtained. Relating to the information, required from the DM in the search of new solutions, these scalarzing problems are relatively near to the scalarizing problems of the reference point [Wierzbicki, 1980], but unlike them, here the DM is not obliged to determine the desired or acceptable levels for all the criteria. In the scalarizing problems, suggested in this chapter, the DM can represent his/her local preferences not only by desired or acceptable levels, but also by desired or acceptable directions and intervals of change in the values of separate criteria. In this way he/she can describe his/her local preferences with greater flexibility, accuracy and reliability. Depending on these preferences, the set of the criteria at each iteration can be indirectly divided into seven or less than seven classes, denoted as follows: $K^{>}$, $K^{\geq}$, $K^{=}$, $K^{<}$, $K^{\leq}$, $K^{><}$ and K^{0}. Each criterion $f_k(x)$, $k \in K$ may belong to one of these classes, as given below:

$k \in K^{>}$, in case the DM wishes the criterion $f_k(x)$ to be improved;

$k \in K^{\geq}$, if the DM wants the criterion $f_k(x)$ to be improved by a desired value $\Delta_k > 0$;

$k \in K^{=}$, in case the DM wishes that the current value of the criterion $f_k(x)$ is not deteriorated;

$k \in K^{<}$, in case the DM agrees the criterion $f_k(x)$ to be deteriorated;

$k \in K^{\leq}$, if the DM wishes the criterion $f_k(x)$ to be deteriorated by a acceptable value $\delta_k > 0$;

$k \in K^{><}$, if the DM wishes the criterion $f_k(x)$ not to be altered beyond the limits of a given interval, determined as:
$f_k - t_k^- \leq f_k(x) \leq f_k + t_k^+$;

$k \in K^{0}$, in case the DM is not interested how the criterion $f_k(x)$ will be altered at this iteration.

In order to obtain a solution, which is better than the current non-dominated solution of the linear problem of multicriteria optimization, the following Chebyshev scalarizing problem L1 can be used [Vassileva et al., 2001] on the basis of the implicit criteria classification done by the DM:

Minimize:

$$S(x) = \max\left[\max_{k\in K^{\geq}}(\bar{f}_k - f_k(x))/|f_k^{'}|,\ \max_{k\in K^{<}\cup K^{\leq}}(f_k - f_k(x))/\left|f_k^{'}\right|\right] + \max_{k\in K^{>}}(f_k - f_k(x))/\left|f_k^{'}\right| +$$

$$+\rho\left[\sum_{k\in K^{\geq}}(\bar{f}_k - f_k(x)) + \sum_{k\in K^{<}\cup K^{\leq}}(f_k - f_k(x)) + \sum_{k\in K^{>}}(f_k(x) - f_k)\right] \quad (1)$$

under constraints:

(2) $f_k(x) \geq f_{k,} \quad k \in K^{>} \cup K^{=}$,

(3) $f_k(x) \geq f_k - \delta_k, \quad k \in K^{\leq}$,

(4) $f_k(x) \geq f_k - t_k^-, \quad k \in K^{><}$,

(5) $f_k(x) \leq f_k + t_k^+, \quad k \in K^{><}$,

(6) $x \in X$,

where f_k is the value of the criterion $f_k(x)$ in the current preferred solution;

$\bar{f}_k = f_k + \Delta_k$ is the desired level of the criterion $f_k(x)$;

$f_k^{'}$ is a scaling coefficient, defined as follows:

(7) $f_k^{'} = \begin{cases} \varepsilon, \ a\kappa o \left| f_k^{'} \right| \leq \varepsilon \\ f_{k,} \ a\kappa o \left| f_k^{'} \right| > \varepsilon \end{cases}$,

where ε is a small positive number.

Given that the objective function (1) of scalarizing problem L1 is a non-differentiable function, an equivalent L2 linear problem could be solved instead of it (Murtaph (1981), [Padberg, 2000]:

(8) $\min\left(\alpha + \beta + \rho \sum_{k \in K^{\geq} \cup K^{>} \cup K^{<} \cup K^{\leq}} y_k \right)$

subject to constraints:

(9) $\alpha \geq \left(\bar{f}_k - f_k(x) \right) / \left| f_k^{'} \right|, \quad k \in K^{\geq}$,

(10) $\alpha \geq \left(f_k - f_k(x) \right) / \left| f_k^{'} \right|, \quad k \in K^{<} \cup K^{\leq}$,

(11) $\beta \geq \left(f_k - f_k(x) \right) / \left| f_k^{'} \right|, \quad k \in K^{>}$,

(12) $f_k(x) \geq f_k, \quad k \in K^{>} \cup K^{=}$,

(13) $f_k(x) \geq f_k - \delta_k, \quad k \in K^{\leq}$,

(14) $f_k(x) \geq f_k - t_k^-, \quad k \in K^{><}$,

(15) $f_k(x) \leq f_k + t_k^+, \quad k \in K^{><}$,

(16) $\bar{f}_k - f_k(x) = y_k, \quad k \in K^{\geq}$,

(17) $f_k - f_k(x) = y_k, \quad k \in K^{<} \cup K^{\leq}$,

(18) $f_k(x) - f_k = y_k, \quad k \in K^{>}$,

(19) $x \in X$,

(20) α, β - arbitrary.

With the help of problem L2, a non-dominated solution of LMK linear multicriteria problem is obtained. In order to obtain more non-dominated solutions of LMK problem, some problems could be used, that are parametric extensions of problem L2 (Murtaph (1981), [Padberg, 2000]. One parametric extension of problem L2, called L2P, may have the following form:

(21) $\min\left(\alpha + \beta + \rho \sum_{k \in K^{\geq} \cup K^{>} \cup K^{<} \cup K^{\leq}} y_k \right)$

under constraints:

(22) $f_k(x) + \left| f_k^{'} \right| \alpha \geq \bar{f}_k + \Delta f_k \, t, \quad k \in K^{\geq}$,

(23) $f_k(x) + \left| f_k^{'} \right| \alpha \geq f_k - \Delta f_k \, t, \quad k \in K^{<} \cup K^{\leq}$,

(24) $f_k(x) + \left| f_k^{'} \right| \beta \geq f_k + \Delta f_k \, t, \quad k \in K^{>}$,

(25) $t \geq 0$

and constraints (12)-(20),

where Δf_k is a parameter.

4. GAMMA-L interactive method

GAMMA-L interactive method designed to solve linear problems of multicriteria optimization is developed on the basis of scalarizing problems L2 and L2P. It is an interactive method oriented towards learning [Gardiner and Vanderpooten, 1997], which means that the existence of an implicit utility function of the DM is not presumed. The DM can seek freely non-dominated solutions in the set of the non-dominated solutions, evaluating on his own whether the current solution found is the most preferred or the final solution of the initial multicriteria problem.

The classification-oriented problems L2 and L2P enable to a different extent the expansion of DM's possibilities to describe his/her local preferences, connected with the improvement of the current non-dominated solution found. These scalarizing problems enable the DM set in addition to the desired or acceptable levels of the criteria also desired or acceptable directions and intervals of change in the criteria values.

The algorithmic scheme of GAMMA-L interactive method consists of the following main steps:

Step 1. Finding an initial non-dominated solution of the multicriteria problem by setting f_k =1, $k \in K$ and $\bar{f}_k$ =2, $k \in K$, and solving problem L2.

Step 2. Representing of the current non-dominated solution obtained to the DM for evaluation. If the DM considers, that this non-dominated solution satisfies his/her global preferences, Step 6 is executed, otherwise – Step 3.

Step 3. A request to the DM to determine his/her local preferences for improving the current non-dominated solution found by defining desired or acceptable levels, directions and intervals of change of a part or of all the criteria.

Step 4. A requirement towards the DM to estimate whether one or more new non-dominated solutions he/she wishes to consider in the evaluation. In the first case scalarizing problem L2 is solved and Step 2 is executed, and in the second case – Step 5 is accomplished.

Step 5. A question to the DM to determine the maximal number s of new non-dominated solutions that he/she wishes to obtain. Solution of scalarizing problem L2P and representing of less or equal to s new non-dominated solutions for evaluation and for choice of a current preferred solution. In case the DM decides that this non-dominated solution satisfies his/her global preferences, Step 6 is executed, otherwise – Step 3.

Step 6. Stop of the process of the linear multicriteria problem solving.

In GAMMA-L interactive method the DM controls the dialogue, the computing process and the conditions for canceling the process of linear multicriteria problem solution.

5. Conclusion

The interactive GAMMA-L method is included in the software insurance of the experimental system MOLIP developed at the Institute of Information Technologies of the Bulgarian Academy of Sciences. This system is designed for interactive solution of continuous and integer multicriteria optimization problems with different number and type of the criteria, with different number and type of the variables and constraints. The advantages of GAMMA-L method and of the interface modules of MOLIP system allow decision makers with different degree of classification to describe comparatively easy his/her local preferences, to evaluate the new solutions obtained, to be trained in the specifics of the multicriteria problems solved and to find the most preferred solution of these problems with a large degree of reliability.

Bibliography

[Gardiner and Vanderpooten, 1997] L.R.Gardiner and D.Vanderpooten. Interactive Multiple Criteria Procedures: Some Reflections. In: *Multicriteria Analysis* (J. Climaco, Ed.). Springer, 290-301, 1997.

[Jaszkiewicz and Slowinski, 1995] A.Jaszkiewicz and R.Slowinski. The Light Beam Search-Outranking Based Interactive Procedure for Multiple-Objective Mathematical Programming. In: *Advances in Multicriteria Analysis* (P. Pardalos, Y. Siskos and C. Zopounidis, Eds.), Kluwer Academic Publishers, Dortrecht, 129-146, 1995.

[Korhonen and Laakso, 1986] P.Korhonen and J.Laakso. A Visual Interactive Method for Solving the Multiple Criteria Problem. *European Journal of Operational Research*, 24, 277-287, 1986.

[Miettinen, 1999] K.Miettinen. Nonlinear Multiobjective Optimization. *Kluwer Academic Publishers*, Boston, 1999.

[Murtagh, 1981] B.A.Murtagh. Advanced Linear Programming: Computation and Practice. McGraw-Hill, New York, 1981.

[Padberg, 2000] M.Padberg. Linear Optimization and Extensions. (Algorithms and Combinatorics, vol. 12), Springer-Verlag, 2000.

[Steuer, 1986] R.E.Steuer. Multipe Criteria Optimization: Theory, Computation, and Applications. John Wiley & Sons, Inc, 1986.

[Vassileva et al., 2001] M.Vassileva, K.Genova and V.Vassilev, (). A Classification based Interactive Algorithm of Multicriteria Linear Integer Programming. *Cybernetics and Information Technologies*, 1, 5 – 20, 2001.

[Wierzbicki, 1980] A.P.Wierzbicki. The Use of Reference Objectives in Multiobjective Optimization. In: *Multiple Criteria Decision Making: Theory and Application. Lecture Notes in Economics and Mathematical Systems* (G. Fandel and T. Gal, Eds.), Springer Verlag, *177*, 468-486, 1980.

Author information

Vassil Vassilev - Institute of Information Technologies, BAS, Acad. G. Bonchev St., bl. 29A, Sofia 1113, Bulgaria; e-mail: vasko@iinf.bas.bg

Krassimira Genova - Institute of Information Technologies, BAS, Acad. G. Bonchev St., bl. 29A, Sofia 1113, Bulgaria; e-mail: krasi@iinf.bas.bg

Marjana Vassileva - Institute of Information Technologies, BAS, Acad. G. Bonchev St., bl. 29A, Sofia 1113, Bulgaria; e-mail: mari@iinf.bas.bg

Subhash Narula - School of Business, Virginia Commonwealth University, Richmond, VA 23284- 4000, USA; e-mail: snarula@vcu.edu

THE DECISION MAKING PROBLEM IN FUZZY CONDITIONS WITH FUZZY MEMBERSHIP FUNCTIONS.

A. Voloshin, G. Gnatienko, E. Drobot

Abstract: *Authors analyses questions of the subjective uncertainty and inexactness situations in the moment of using expert information and another questions which are connected with expert information uncertainty by fuzzy sets with rough functions of belonging in this article. You can find information about integral problems of individual expert marks and about connection among total marks "degree of inexactness" with sensibility of measurement scale. A lot of different situation which are connected with distribution of the function accessory significance and orientation of the concrete take to task decision making are analyses here.*

Keywords: *expert, fuzzy sets.*

Introduction

Different kinds of uncertainty are to some extent characteristics of practically any situation of making decision in which the expert information is used. The nature of uncertainty is essentially different. Firstly, it's necessary to

point out the objective uncertainty which is peculiar to all real constants and is connected with our world's "organization" itself [Нариньяни, 1994]. Secondly, the subjective uncertainty is peculiar to human's nature on the whole and to his abilities to evaluate the information in particular. The reasons of the origin of the second type of uncertainty are являются [Нариньяни, 1994; Орловский, 1981]: expert's lack of knowledge about the characteristics of the objects; his unsatisfied confidence degree in correctness of his marks; knowledge contradiction; in distinction of the information presentation and the semantic uncertainty corrected with different are indirect meanings of the natural language, indefiniteness of the keywords and definitions. Beside, indefiniteness appears in the process of marks integration coming from different experts.

The result of the research [Ларичев, 2002; Борисов, 1989] show that the main difficulty is caused by the necessity to appraise numerical values of the objects (variants, criteria) or to give numerical evaluation on the ratio scale between them. But the person fully much more confident [Орловский, 1981; Ларичев, 2002] if the has an opportunity to give unclear marks in the form of the intervals or with the help of rough sets pointing out the degree of belonging of the objects to these sets. In particular, more often in a form of package the additive criterion is used and its objectivation is connected with defining of weight coefficients of the objects [Волошин, Гнатиенко, Дробот, 2003]. This approach seemed to be even more justified because in prevailing number of the cases it is enough to have the approximated characteristic of the data set and the expert info does not demand high accuracy.

So, the application of fuzzy sets and the function of belonging allow to formalize factors of uncertainty and unclearness with might happen in expert evaluation situation by some means. The function of belonging was interpreted in various works differently: as "subjective probability" [Орловский, 1981], expert's confidence degree in object's belonging to the concept described by fuzzy set [Борисов, 1989], the opportunity of its interpretation by this concept [Мелихов, 1990] and so on. For all this was traditionally considered reflection. But "unnatural" distinctiveness and simple concept that according to its destination is called to reflect distinction and inexactness of subjective marks. According to the author's opinion it is incorrect to demand precise meanings of the function of belonging from the expert. It is more logical to demand rough marks of the function of belonging because the situation of subjective evaluation is connected with the principle inexactness ("objective", "situational", "semantic" and so on [Нариньяни, 1994]) in the definition of the value of function of belonging.

In connection with this author analyses the problems of setting and processing of expert information with the help of rough functions of belonging.

The forms of presentation of rough function of belonging

Let's X as the universal set of n-measured alternatives that describes totality of all possible variants of expert's choice. Let's describe rough concept by fuzzy set $A = \{(x,\ \mu_A(x))\}$, $x \in X$, $\mu_A(x)$, where $\mu_A : x \to [0,1]$, - function of belonging.

The expert is suggested to set the information about the value of function of belonging:

a) in the form of the interval $\mu_A(x) \in [\mu^H{}_A(x), \mu^B{}_A(x)]$;

b) pointing the absolute inexactness $\mu_A(x) = \mu^0{}_A(x) \pm \Delta\mu_A(x)$, where $\mu^0{}_A(x)$ - "exactness of measureness";

c) pointing relative inexactness $\mu_A(x) = \mu^0{}_A(x) \pm \varepsilon\mu^0{}_A(x)$, $\varepsilon \in (0,1]$.

We accept as well to get the expert's information about inexact function of belonging is a form of

d) triad $\{(x,\ \mu_A(x),\ \gamma(\mu_A(x)))\}$, where γ, $\gamma \in [0, 1]$, - expert's confidence degree is his mark.

Rough function of belonging is interpreted as sphere of expert's insensibility (inexactness, uncertainty) while defining function of objects belonging x, $x \in X$, to set A. And the sweep of the interval (expert's confidence degree in his mark) characterizes quantitative measure of this inexactness. Formally this uncertainty might be defined with the help of so called "granularity" [Нариньяни, 1994] which reflects inexactness mark of the concrete parameter according to the granular size which is indivisible (and, of course, inexact) total mark of this parameter. In our case this parameter is the function of belonging that is defined on the interval [0,1], the smallest union of

which defines the maximum granularity of the scale. The size of the granular is defined according to consideration of "limit of distinction"of granular for the expert.

With it is the help of the concept of granularity it's case to define the absolute mark of the "limit of distinction" of the expert mark for cases a)-d). Let's define this mark as ξ, and the granular size as h. Then constant ξ in cases a) - d) is defined correspondingly:

$$\xi = h/(\mu^{B}{}_{A}(x) - \mu^{B}{}_{A}(x)) ,$$

$$\xi = h/\Delta\mu_{A}(x) ,$$

$$\xi = h/\gamma(\mu_{A}(x)) .$$

The term "exactness degree" of the expert mark is used further on while working out methods of operating with "unclear" function of belonging.

Methods of defining unclear group mark

In the article [Волошин, Гнатієнко, 2002] two ways of working out of mathematical apparatus for operating unclear function of belonging are pointed out.

The first way – preceding scalarization of rough function of belonging with consequent usage of standard apparatus of unclear analysis.

Because of the principle position of the subjective factor in drawing the function of belonging (as "value of function of belonging is subjective possibility"), it is necessary to take into consideration the expert's "psychological parameters" ("realism", "independence", "truthfulness", "inclination to risk" and so on [Волошин, Панченко, 2002]). For example, in case of one expert

$$\mu_{A}(x) = \alpha\mu^{\min}{}_{A}(x) + (1-\alpha)\mu^{\max}{}_{A}(x) , \quad (1)$$

here α, $0 \le \alpha \le 1$, - the coefficient of expert's "riskiness".

In case of a group of expert two-phase procedure of scalarizations suggest (for each expert and for group). In general case the expert marks are correlated to some weight coefficients which reflect there expert confidence degree, i.e. $\lambda_{i} \in [0;1]$, $\sum_{i=1}^{m}\lambda_{i} = 1$, where m – expert quantity. Competence coefficients are calculated, for example, according to the method [Волошин, Гнатієнко, 1993].

Let's define a set of objects indexes as $I = \{1,..,n\}$, here n – is objects quantity in set X, and set of experts indexes - $J = \{1,..,m\}$. Let's define individual function of belonging, scalarized by the formula (1) as $\mu_{A}(x_{i}, p^{j}{}_{i})$.

One of the most wide spread ways of the choice of the best object [Мелихов, 1990] consist in the choice of the object which has the maximum degree of belonging to fuzzy set A (criterion of minimax), i.e.

$$\mu_{A}(x_{i}) = \max_{i\in I}\min_{j\in J}(\lambda_{j}\mu_{A}(x_{i}, p^{j}{}_{i}) . \quad (2)$$

In [Орловский, 1981] the effectiveness of linear package for the task of choice is pointed out:

$$\mu_{A}(x_{i}) = \max_{i\in I}\sum_{j\in J}(\lambda_{j}\mu_{A}(x_{i}, p^{j}{}_{i})) \quad (3)$$

It coefficient of experts competence are considered as distribution of possibilities we can use the criteria of Bayes-Laplas or Yurnits [Мушик, 1990], so

$$\mu_{A}(x_{i}) = \max_{i\in I}\sum_{j\in J}\lambda_{j}\mu_{A}(x_{i}, p^{j}{}_{i})/m , \quad (4)$$

$$\mu_A(x_i) = \max_{i\in I}(\alpha \min_{j\in J} \lambda_j \mu_A(x_i, p^{j}_{i}) + (1-\alpha) \max_{j\in J} \lambda_j \mu_A(x_i, p^{j}_{i}) . \quad (5)$$

The methods of scalarization of rough function of belonging as for one expert and so as a group of expert have one drawback in common. In a result mark the level of individual measure of inexactness while defining function of belonging by the ways a)- c) is not taken into consideration, so as the degree of collective inexactness while integrating individual opines.

There are real situation in which it's more preferable to get rough functions of belonging which use the expert information more "carefully". It is connected first of all, with the principle inexactness of waited result when there is sense to consider all the range of possible values of resultant functions of belonging. The most typical example is medical diagnosis and prognosis.

In the cases we consider working out the operations for "rough" functions of belonging, presented in form a)- d) as more advisable which allow to preserve information about the degree of individual "uncertainly" of functions of belonging.

So for the cases a)- d) be the way of definite transformations come to each other, let's consider different ways of formalization of individual rough functions of belonging, defined by the expert is a form of intervals (case a)) is a collective form of belonging. The procedures are realized in two phases as wall: in the first phase the integration of individual interval marks into collective rough mark in a form of interval takes place; in the second phase the criteria of optimum decision taking are defined.

The most is the integrative mark suck as:

$$\mu^{*(\min)}{}_A(x_i) = \min_{j\in J} \lambda_j \mu^{*(\min)}{}_A(x_i, p^{j}_{i}), \qquad \mu^{*(\max)}{}_A(x_i) = \min_{j\in J} \lambda_j \mu^{*(\max)}{}_A(x_i, p^{j}_{i}) . \quad (6)$$

Defining the resultant interval in suck a way $[\mu^{*(\min)}{}_A(x_i), \mu^{*(\max)}{}_A(x_i)]$ there are no preferences inside the sphere, i.e. all the values are equally possible. Evidently, the more sphere sweep is according to (6), the lest accurate the resultant function of belonging is defined.

The other approach to word defining if resultant interval on a set of individual intervals consists in defining its measure as centers of gravity of the sets accordingly $\mu^{(\min)}{}_A(x_i, p^{j}_{i})$ и $\mu^{(\max)}{}_A(x_i, p^{j}_{i})$, $i \in I$, $j \in J$, according to the formulas:

$$\mu^{*(\min)}{}_A(x_i) = \sum_{j=1}^{m} \lambda_j \mu^{(\min)}{}_A(x_i, p^{j}_{i}) / m \quad , \qquad \mu^{*(\max)}{}_A(x_i) = \sum_{j=1}^{m} \lambda_j \mu^{(\max)}{}_A(x_i, p^{j}_{i}) / m . \quad (7)$$

It the distribution inside the individual intervals is not equal, for example, is defined by b), so while defining the resultant intervals it is also taken into consideration accordingly.

Finally the resultant intervals of functions of belonging are defined by usage of the of the definite criteria of "inexactness degree" of value which are based on total marks of the concrete interval with "granularity" of measurement scale. For example, the resultant function of belonging are defined by points or the help of the interval according to the principle:

$$\mu^{*}{}_A(x_i) = \begin{cases} 1/2(\mu^{*(\max)}{}_A(x_i) + \mu^{*(\min)}{}_A(x_i)), & \mu^{*(\max)}{}_A(x_i) - \mu^{*(\min)}{}_A(x_i) \sim h \\ [\mu^{*(\max)}{}_A(x_i), \ \mu^{*(\min)}{}_A(x_i)], & \mu^{*(\max)}{}_A(x_i) - \mu^{*(\min)}{}_A(x_i) >> h \end{cases}, \ i \in I .$$

For the tasks of medical and legal diagnosis "displacement" of values distribution of function of belonging toward, the left border occurs; and for technical and social diagnosis – toward the right. So we've got "the second derivative"of unclear function of belonging that is defined by the group of system experts according to the context of the problem [Ларичев, 2002] .

Further on the criteria (2)-(5) are used with the resultant intervals of possible values of functions of belonging in each perimeter.

The procedures on the analogy are suggested for unclear binary relations with rough functions of belonging.

Conclusion

The development of mathematical apparatus for operating with unclear functions of belonging is very actual because it would allow to solve the tasks of decisions taking on the basis of expert information with the help of the methods more adequate to the nature of expert mark which is characterized by different types of inexactness and uncertainty.

Bibliography

[Нариньяни, 1994] Нариньяни А.С. Неточность как Не-фактор. Попытка доформального анализа. – Москва-Новосибирск, 1994 г. Препринт РосНИИ ИИ, № 2. – 34 с.

[Орловский, 1981] Орловский С.Г. Проблемы принятия решений при нечеткой исходной информации. М.: Наука, 1981. – 208 с.

[Ларичев, 2002] Ларичев О.И. Свойства методов принятия решений в многокритериальных задачах индивидуального выбора // Автоматика и телемеханика, № 2, 2002. – С. 146-158.

[Борисов, 1989] Борисов А.Н., Алексеев А.В. и др. Обработка нечеткой информации в системах принятия решений. – М: Радио и связь, 1989. – 304 с.

[Волошин, Гнатиенко, Дробот, 2003] Волошин А.Ф., Гнатиенко Г.Н., Дробот Е.В. Метод косвенного определения интервалов весовых коэффициентов параметров для метризированных отношений между объектами // Проблемы управления и информатики, 2003, № 2.

[Мелихов, 1990] Мелихов А.Н., Берштейн Л.С., Коровин С.Я. Ситуационные советующие системы с нечеткой логикой. – М.: Наука, 1990. – 272 с.

[Волошин, Гнатієнко, 2002] Волошин О.Ф. , Гнатієнко Г.М. Прийняття рішень в нечітких умовах з розмитою функцією належності // Праці міжнародної школи-семінару “Теорія прийняття рішень”. – Ужгород. – УжНУ. – 2002. – С. 20-21.

[Волошин, Панченко, 2002] Волошин О.Ф., Панченко М.В. Використання експертного оцінювання для якісного прогнозування на основі багатопараметричних залежностей // Математичні машини і системи, 2002. - № 2. – С. 83-89.

[Волошин, Гнатієнко, 1993] Волошин О.Ф., Гнатієнко Г.М. Процедури визначення компетентності експертів // Вісник Київського університету. Фізико-математичні науки. – К., 1993, № 3. – С. 102-111.

[Мушик, 1990] Мушик Э., Мюллер П. Методы принятия технических решений..- М.: Мир, 1990.-208 с.

Author information

Alexey F. Voloshin – Taras Shevchenko University of Kiev, Ukraine, e-mail: ovoloshin@unicyb/kiev/ua

Grygoriy N. Gnatienko – Kiev, Ukraine, e-mail: G.Gnatienko@veres.com.ua

Elena V. Drobot - Kiev, Ukraine, e-mail: elena_drobot@ukr.net

THE SYSTEM OF QUALITY PREDICTION ON THE BASIS OF A FUZZY DATA AND PSYCHOGRAPHY OF THE EXPERTS

Voloshin O.F., Panchenko M.V.

***Abstract**: The system of development unstable processes prediction is given. It is based on a decision-tree method. The processing technique of the expert information is offered. It is indispensable for constructing and processing by a decision-tree method. In particular data is set in the fuzzy form. The original search algorithms of optimal paths of development of the forecast process are described. This one is oriented to processing of trees of large dimension with vector estimations of arcs.*

***Keywords**: Method of a decision-tree, fuzzy expert data, search of optimal paths.*

Introduction

Methods of quantitative prediction (the time series, regression the analysis, simulation modeling etc.) give poor outcomes at prediction of "unstable" processes. In their basis there is "the prolongation past". The unstable processes are characterized “by violation of monotonicity”. It means, that there are discontinuous changes, unrepresentative for development of the process in past. The problem consists in representation of the future, which one can not be interpreted as customary prolongation past. The future can accept the in essence new forms. In a basis of such prediction (“quality prediction”, “guessing”) lays idea of immediate usage of knowledge of the person (expert). Thus first of all it is necessary to allow for “fuzzy” of the expert information [T.Terano, 1993], which one depends on its professional and psychological characteristics (competence, independence, objectivity, realism, tendency to hazard etc.). In a basis of the system, which one is described, the method of a decision-tree lays. The expert information will be utilized both for constructing the tree, and for an estimation of its arcs on the basis of method of on - pair matching. The expert information can be set both in the determined sort, and in the indistinct form. At processing the expert information for finding collective estimations the algebraic method will be utilized, in which one the metrics Hamming and measure of an incongruity of ranks of objects is applied.

In closing section the description of the system, ways of representation and input data processing is given.

As by search of optimal paths of the process development (optimal path from the root to leaves of a tree with allowance for of vector estimations) there is a task of large dimension, the original methods based on generalization of known scalar methods with usage of the schemes of a sequential analysis of variants are tendered [Voloshin, 1989].

The introduced outcomes are development of works [Voloshin, 1999], [Voloshin, 2001].

Decision-tree constructing

The group from n ($n \geq 1$) jointly working experts selects and sub problems and creates a decision-tree by determining importance (“weight”) each unit of a tree. The processing of the expert information at all stages will be carried out with allowance for of “weights” of the experts and degree of coordination of their judgments [Makarov, 1982].

In leaves of a tree there are tops, for which one sub problems are not determined any more. After arrangement of weights of arcs in a decision-tree the weight (probability, importance) each leave, that is path from the root into this leave is evaluated.

The decision-tree is created by a commission of experts (or which is a making decision person - a MDP). For each of tops of a tree (except for leaves) they determine the subordinate tops (for each problem there are determined sub problems). A weight of arcs can be set both in scalar and in the vector form. Last can be interpreted as a vector estimation of an arc (for example, efficiency and cost), or as an indistinct estimation of the expert assigned to a vector of values of the function of an accessory.

Such variants of the decision-tree definition are possible:

- The tree creates a MDP. Then the operation of the experts is only to place of a weight in a tree;
- The tree is created by a commission of experts with application of a method of pair matching.

Thus, if the task of "measurement" (obtaining of a scalar estimation) is considered, each expert gives three estimations: a_1^i - "optimistic", a_2^i - "realistic" and a_3^i - "pessimistic".

The resulting estimation is thus. An estimation of each expert $a_i = \frac{a_1^i * \gamma_1 + a_2^i * \gamma_2 + a_3^i * \gamma_3}{\gamma_1 + \gamma_2 + \gamma_3}$ at first averages, and then, with allowance for of weights of the experts, we compute a resulting estimation. The coefficients γ_1, γ_2, γ_3 are defined empirically. On one technique $\gamma_1 = \gamma_3 = 1$, $\gamma_2 = 4$ (expert - "realist"), on other - $\gamma_1 = 3$, $\gamma_2 = 0$, $\gamma_3 = 2$ (for "optimist") and $\gamma_1 = 2$, $\gamma_2 = 0$, $\gamma_3 = 3$ (for "pessimist").

For definition of a psychological type of the expert (pessimist, optimist, realist) by included in the system resources, will be carried out psychological testing of the experts and the coefficients of "realness" ($\frac{1}{3} \le \lambda \le \frac{2}{3}$ for the realist, $0 \le \lambda \le \frac{1}{3}$ for the pessimist, $\frac{2}{3} \le \lambda \le 1$ for the optimist) are further allowed.

The coefficient of competence k_2^i is calculated on the basis of accuracy of the previous prognoses on a technique [Makarov, 1982]. The initial coefficients k_2^i are selected equal $\frac{1}{n}$.

One consists of main reasons of unauthenticity of the expert information in not the registration of the fact of possible "dependence" of the experts (from an eventual result, from administrative subordination etc.) or their truthfulness (for definite reasons, including, irrelevant with competence). Sometimes person is realized or is not realized speaks a lie. Also, on the basis of the psychological tests the coefficients of "truthfulness" k_3^i, "independence" k_4^i and "caution" k_5^i of the experts are defined. In outcome are evaluated of a weight of the experts $\alpha_i \frac{\sum \mu_i * \kappa_i}{\sum \mu_i}$, where μ_i, in turn, weight of the factors κ_j^i, $i = \overline{1,n}$, $j = \overline{1,s}$. For different problems their value can vary. The indistinct definition of parameters κ_j^i is admitted.

Pair matching method for weak ranging

The definition of advantage in the determined form. Experimentally is shown, that the complication for the expert represents constructing ranging on the basis of the simultaneous registration of several different properties, on which one the objects $o_i, i = \overline{1,n}$ are evaluated. In such cases the experts decide the tasks of pair matching.

Each expert makes C_n^2 of matching, comparing each object with each. The outcome of matching j-th of the expert is represented by a matrix A^j by a size $n \times n$. A unit $a_{ik} = 1$ in only case when in judgment of the j-th expert the i-th object is more preferential then k - th. By an necessary and sufficient condition that it is possible to set advantages thus, there is the ratio acyclicity of the expert advantage [Makarov, 1982].

After the definition of matrixes A^j the matrix $A = (a_{qt}) = \sum_{j=1}^{N} A^j$ is evaluated then one finds $a_s = \sum_{t=1}^{n} a_{is}$ (here $s = \overline{1,n}$). It also is estimations every variants. Such processing technique of the information is further offered:

- The definite level L and is set, if $a_s < L$ ($s = \overline{1,n}$), the variants are discarded, if the estimations do not exceed a given level. For variants, which one have remained, the probabilities, proportional their estimations a_s are evaluated;
- Any of variants is not discarded and to each the appropriate probabilities are assigned.

The definition of advantage in a fuzzy form. The units of a matrix A^j are vectors by dimension m (values of the accessory function). Each unit of such vector is real number from 0 up to 1. Here is $a^j_{tq} = | a^j_{qt} - 1 |$, $a^j_{ii} = (\frac{1}{2},...,\frac{1}{2})$; (here 1 is a unit vector of dimension m). Accordingly, in a matrix A there is $a_{qt} = \frac{\sum_{j=1}^{N} a^j_{qt}}{N}$, $a_s = \sum_{i=1}^{n} a_{is}$ $o_i > o_j$ in only case when $a_i > a_j$.

Algebraic processing techniques of the expert information

The essence of an algebraic processing technique of the expert information consists in introduction of some distance between estimations and according to this comparison to the system of some ranging.

Let Ω - set of all weak ranging of objects. Then the resulting estimation is on one of the formulas:

$A_1 \in Arg \min_{A \in \Omega} \sum_{i=1}^{N} d(A, A^i)$ (Cameni-Snell median);

$A_2 \in Arg \min \sum d^2$ (average meaning);

$A_3 = Arg \min_{A \subseteq \Omega, i=1,N} d(A, A^i)$ (Compromis);

d is distance between ranging.

As distance between ranging the metrics of the Humming is used [Voloshin, 2001]:

$$d(A,B) = \frac{1}{2} * \sum_{i,j=1} | a_{ij} - b_{ij} | \quad \textbf{(1)}$$

Or there is used measure of an incongruity of object ranks

$$d(A,B) = \sum_{i} | a_i - b_i |, \quad \textbf{(2)}$$

Here a_i, b_i - rank of i-th object in ranging, which one is set by matrixes A and B.

In case of the definition of advantage in a fuzzy form in the formulas (1), (2) units are set through values of the accessory function, and the complication is encompass bought a large information content, which one is necessary for analyzing.

Search of optimal paths

There is a necessity for finding optimal paths in a decision-tree. It combines definite tops, with highest weight (that is most probable of variants of a development of events). For this purpose the method of a sequential analysis of variants will be used [Voloshin, 1989]. It grounded on a method Dejkstra [Edward Minieka, 1981]. It will be utilized for search of the shortest path from top s in top t:

Step 1. Before a start of algorithm execution all tops and arcs are not colored. at algorithm fulfillment of the number $d(x)$ is assigned To each top, which one is equal to length of the most short path from s in x, which one includes only colored tops ($d(s) = 0$, $d(x) = \infty$).

Step 2. For each uncolored top x we enumerate the value $d(x)$:

$$d(x) = \min\{d(x), d(y) + a(y,x)\}$$

If there is $d(x) = \infty$ for all uncolored tops x: to complete execution of algorithm. It means that in the initial graph there are paths from top s in uncolored tops. Otherwise it is necessary to color that of top x that has the least value $d(x)$. Let's assume y=x.

Step 3. If y=t to complete the procedure. The most short path from s in t is retrieved (it is alone path from s in t, which one consists of colored arcs). Otherwise: to go to step 2.

In case of the arcs graph definition is fuzzy (It means the vectors of numbers $d(x) = (d_k(x)), k = \overline{1,n}$ are used) the following methods of the most probable paths finding are applied.

Method of convolutions. It is offered to substitute vector estimations by numerical one. For this purpose apply known convolutions. For example, it can be average value of a vector units or value computed by “Hodge - Leman method”:

$$a_i = \beta * \min a_{ij} + (1-\beta) * \sum_i a_{ij} * p_j$$

a_{ij} - are units of a vector, p_j are their weights, $\beta \in (0;1)$ - is a “collective caution” coefficient

$$\beta = \frac{1}{n}\sum_{i=1}^{n} k_s^i$$

After that the circumscribed above Dejkstra method is applied. Let's mark, that if the convolutions are additive, one of Pareto-optimal paths will be retrieved [Makarov, 1982].

Modified Dejkstra algorithm. The following generalization of Dejkstra algorithm is offered. It is necessary to find the shortest path from top s in top t.

Step 1. $d_i(s) = (0,...,0)$ and $d_i(x) = (\infty,...,\infty)$ for all $x \neq s$, i=0.

Step 2. For each uncolored top x as follows we enumerate the value $d_i(x)$:

$$d_i(x) = \min\{d_i(x), d_i(y) + a(y,x)\}$$

If the vectors $d_i(x)$ and $d_i(x) + a(y,x)$ are incomparable: to store them both.

So, for tops x_i there are such characteristics:

$$d(x_i) = (d_1(x_i),..., d_k(x_i))$$

$d_i(x_i)$ - is a length of one of possible paths in top x_i.

After that we select dominating tops $x_i, i \in A$. A is some set, for which one is fulfilled such condition $\neg\exists x_j, j \notin A \neg\exists k : d_k(x_j) \geq d_p(x_i) \forall p$, and color them.

Step 3. If y=t, we complete the procedure. The shortest path from s in t is retrieved. Otherwise: go to step 2.

The best case, when all paths in a tree will be comparable, the algorithm works with speed of Dejkstra algorithm.

Matrix method. Will be utilized in that case, when the tree is divided into levels, and each top of a tree refers only to tops of a lower layer. Each level of a tree is set by a matrix $A^k = \{a_{ij}\}^k = \{a_{ij}^1,...,a_{ij}^m\}^k$, here a_{ij} - is probability of transition from i-th top of a k - th level into j-th top of a к+1 level is possible. And, if for к-й of a matrix was i of columns, for к+1 of a matrix will be i of rows. To each matrix is added one row and one column. If the maiden matrix has only one row, in the last column the unit vector is written. If it has some

rows, in the last column then the vector $v_i = (\frac{1}{c},....,\frac{1}{c})$ is written, where with - amount of tops at the maiden level (that is all tops of the maiden level are equivalent). A unit of the last row is $a_{l+1,i} = (\sum_{j=1}^{n} a_{ij}^1 * a_{jm+1}^1)$.

Thus, in last column of the last matrix we obtain of tops weights of the lowest level in a tree. Utilizing the obtained information of tops weights in a tree, it is possible to define optimal paths in a tree. In case of large information content it is expedient to apply approximate methods of peephole optimization. For example, known method of a drop down vector [Sergienko, 1985]:

It is necessary to find the shortest path from top s to top t.

Step 1. y=s. We define a neighborhood r (it can be both path length, and amount of tops on the way). In the given neighborhood we discover top x in a tree. It is a shortest route from top y (and which one, accordingly, smaller than r, or consists less than r of tops). We color path from y in x.

Step 2. Let's assume y=x. We repeat the procedure.

Step k. y=t. The path from s in t is retrieved.

In case of a large information content the splitting of a tree into the sub trees is carried out with the help of decomposition methods of a sequential analysis of variants [Voloshin, 1989].

The burn-time of Dejkstra algorithm is $O(1.5N^2)$, N there is an amount of tops in a tree. In case of the fuzzy definition of arcs, it is necessary to produce the convolution operation, then the burn-time of algorithm will be equal $O(1.5N^2 + K)$, here K is an amount of arcs.

The description of the system.

The decision-tree is set by a matrix of incidences. In each cell of a matrix there is a vector a_{ij}, which one sets a transition probability from top i in top j. It consists of ten natural numbers $(a_1,...,a_{10}), 0 \le a_i \le 1$. The sum of units of each row is equal to a unit vector. The matrix is filled in by interrogation of the experts. There are functions: addition of rows and columns, backup of number, dictionary, saving of the table in the file, loading of the table from the file.

For expert interrogation it is necessary to take advantage of the form, which one will allow to set up to 10 matrixes of identical dimension. Each of such matrix is an outcome of matching by the expert of tops variants, which one can be included in a tree (matrix A^j). The analysis of these matrixes will further be carried out. In outcome the tops are determined, which one is included in a tree. Probabilities of transition in them from high level top are determined. These outcomes are recorded in a current row of a matrix of the table, which one describes a decision-tree level.

After the definition of an incidences matrix, its analysis is possible. For this purpose it is necessary to set two tops in a decision-tree and the shortest (least probability) and most lengthy (most probable) path connecting these tops will be retrieved. As well the count of total load of each top in a tree is possible.

If the decision-tree is divided on some sub trees, which one have identical leaves, probabilities of these leaves in each of the sub trees are evaluated at first, and then there are average probabilities for entire tree as a whole.

Conclusion

The application of the given system is possible in such areas as medical diagnostic, the prediction of currency course etc. And system accuracy depends only on proficiency of the experts.

Bibliography

1. [T.Terano, 1993] T.Terano, K.Asai, M.Sugeno. “Applied fuzzy systems” (in Russian) M.:Mir 1993 – P. 230-241.
2. [Волошин, 2001] Voloshin O.F., Panchenko M. V. Unstable processes prediction by decision-tree method is based on pair matching method for analyzing expert data. Proceedings of KDS-2001 conference. Saint Petersburg. 2001.-p. 50-53.
3. [Voloshin O.F., 1989] Voloshin O.F. “Method of localization of area of an optimum in tasks of mathematical programming” (in Ukrainian)// The reports of AS USSR t. 293 №3, 1989. – P.264-273.
4. [Voloshin O.F., 1999] Voloshin O.F., Panchenko M.V., Pihotnyk E.P. “Expert system of support of forecasting of a grivna rate” (in ukranian)// Artificial intelligence, 1999, №2, -P.354-359.
5. [Makarov V.I., 1983] Makarov V.I. “Bases of acceptance of the decisions and theory of a choice” (in Russian) M.Nauka, 1983.–P.185-189.
6. [Voloshin O.F., 2001] Voloshin O.F. Gnatienko G.M. “Construction collective ranging by the measure ranks of objects” (in Ukrainian)// Bulletin of Kiev University, Series: Cybernetic, № 4, 2001. – P. 155-158.
7. [Sergienko I.V., 1985] Sergienko I.V. “Mathematical models and methods of the decision of tasks of discrete optimization” (in Russian) Kiev: Nauka Dumka, 1985.-384 p.
8. [Edvard Minieka, 1981] Edward Minieka “Optimization Algorithm for Network and Graphs” (in Russian) –M.:Mir,1981. –P. 40-50.

Authors information

Oleksii F. Voloshin, Kyiv T. Shevchenko national University, the faculty of cybernetics, Kiev, Ukraine. Professor. E-mail: ovoloshin@unisyb.kiev.ua; smost@i.com.ua

Maksim V. Panchenko - Kyiv T. Shevchenko national University, the faculty of cybernetics, Kiev, Ukraine. Post-graduate. E-mail: panchenko@ukr.net

Section 3: Intelligent Networks and Agents

ONE APPROACH FOR THE OPTIMIZATION OF ESTIMATES CALCULATING ALGORITHMS

A.A. Dokukin

***Abstract**: In this article the new approach for optimization of estimations calculating algorithms is suggested. It can be used for finding the correct algorithm of minimal complexity in the context of algebraic approach for pattern recognition*

***Keywords**: Pattern recognition, estimates calculating algorithms.*

Introduction

This work is made in the context of algebraic approach [1] (in what follows, we use the notation and definitions from [1,2]) for pattern recognition. The task of recognition is considered. We have a set *M* of possible objects. It is presumed that $M = M_1 \times ... \times M_n$, there M_i are sets of possible values of *i*-th feature, and some semi-metrics are defined on each of them. The set *M* is divided into *l* classes $K_1,...,K_l$. The task of recognition is defined by the conventional learning information $I_0 = \{S_1,...,S_m,\alpha(S_1),...,\alpha(S_m)\}$, and the finite sample, $\beta(S^i) = (\beta_{i1},...,\beta_{il})$ of test objects. Here $S_1,...,S_m$ are descriptions of training sequence objects $S_i = (a_{i1},a_{i2},...,a_{in}), a_{ij} \in M_j$, $i = \overline{1,m}, j = \overline{1,n}$, and $\alpha(S_i) = (\alpha_{i1},...,\alpha_{il})$ are information vectors of objects S_i, with respect to the properties $P_j(S) \equiv \{S \in K_j\}$, $j = \overline{1,l}$. Correspondently $\beta(S^j) = (\beta_{i1},...,\beta_{il})$ are information vectors of S^j.

The task is to find algorithm in the algebraic closure of some set of recognition operators that calculates information vector for each $S^i \in \widetilde{S}^q$. As such system the defined below class of ECA (estimates calculating algorithm) is considered.

Yu.I. Zhuravlev have proved [1] that there exixts a correct polynomial in the algebraic closure of ECA, i.e. polynomial that provides no errors on the control information $\widetilde{S}^q, \{\beta(S^1),...,\beta(S^q)\}$.

Estimates calculating algorithm A is defined as $A = B \cdot C$, where $B(I_0,\widetilde{S}^q) = \left\|\Gamma_{ij}\right\|_{q\times l} = \left\|\Gamma_j(S^i)\right\|_{q\times l}$ is recognition operator, $C(\left\|\Gamma_{ij}\right\|_{q\times l}) = \left\|\beta_{ij}\right\|_{q\times l}$ is solving rule.

$$\Gamma_j(S^i) = x_1\Gamma_j^1(S^i) + x_0\Gamma_j^0(S^i). \quad (1)$$

$$\Gamma_j^1(S^i) = \frac{1}{Q_1}\sum_{S\in\widetilde{K}_j}\sum_{\omega\in\Omega_A}\gamma(S^i)p(\omega)B(\omega S^i,\omega S) \quad (2)$$

$$\Gamma_j^0(S^i) = \frac{1}{Q_0}\sum_{S\in C\widetilde{K}_j}\sum_{\omega\in\Omega_A}\gamma(S^i)p(\omega)\overline{B(\omega S^i,\omega S)}. \quad (3)$$

Following notation is used:

- The *j*-th class and its addition are denoted as $\widetilde{K}_j = K_j \cap \{S_1,...,S_m\}$ and $C\widetilde{K}_j = \{S_1,...,S_m\} \setminus \widetilde{K}_j$.
- Let $\{\Omega\}$ is the set of all subsets of $\{1,...,n\}$. Some subset Ω_A of Ω is attributed to an algorithm. Its elements $\omega_t = \{i_1,...,i_{k_t}\} \in \{\Omega_A\}$ are called support sets and $p(\omega_t) = p_{i1} + ... + p_{ikt}$ are their weights, $p(\omega_t) \geq 0$.
- $\gamma(S^i) \geq 0$ are weights of training objects.
- $B(\omega S^i, \omega S)$ is proximity function. We use proximity functions only of the following type. Let $\varepsilon_1,...,\varepsilon_n$ are non-negative numbers, let also $\omega S = \{a_{i1},...,a_{ik}\}$, $\omega S' = \{b_{i1},...,b_{ik}\}$ then $B(\omega S, \omega S') = \begin{cases} 1, & \rho_{i1}(a_{i1},b_{i1}) \leq \varepsilon_{i1},....,\rho_{ik}(a_{ik},b_{ik}) \leq \varepsilon_{i\varepsilon} \\ 0, & otherwise \end{cases}$.

$\overline{B(\omega S^i, \omega S)} = 1 - B(\omega S^i, \omega S)$.

Denote a set of recognition operators by $\{\widetilde{B}\}$. Let $B', B'' \in \{\widetilde{B}\}$, $B'(I_0, \widetilde{S}^q) = \left\| \Gamma_{ij}{}' \right\|_{q \times l}$, $B''(I_0, \widetilde{S}^q) = \left\| \Gamma_{ij}{}'' \right\|_{q \times l}$, *b* is a scalar. Following operations bB', $B' + B''$, $B' \cdot B''$ can be defined on this set as shown below.

$$(bB')(I_0, \widetilde{S}^q) = \left\| b\Gamma_{ij}{}' \right\|_{q \times l} \quad (4)$$

$$(B' + B'')(I_0, \widetilde{S}^q) = \left\| \Gamma_{ij}{}' + \Gamma_{ij}{}'' \right\|_{q \times l} \quad (5)$$

$$(B' \cdot B'')(I_0, \widetilde{S}^q) = \left\| \Gamma_{ij}{}' \cdot \Gamma_{ij}{}'' \right\|_{q \times l} \quad (6)$$

The closure $M(\{\widetilde{B}\})$ with respect to operations (4)-(6) is associative algebra with commutative multiplication. Operators from $M(\{\widetilde{B}\})$ can be presented as polynomials of operators from $\{\widetilde{B}\}$. If $B \in M(\{\widetilde{B}\})$ then $B = \sum B_{i_1} \cdot B_{i_2} \cdot ... \cdot B_{i_t}$. The maximum number of multipliers in its items is called the degree of recognition operator.

The family $M(\{A\})$ of algorithms $A = B \cdot C$ such that $B \in M(\{\widetilde{B}\})$ is called algebraic closure of $\{A\}$.

Finally we will need some more terms from [3] to continue the statement. The informational matrix $\left\| \beta_{i,j} \right\|_{q \times l}$ is considered. Suppose $M = \{(i,j)\}$, $i = 1,...,q$, $j = 1,...,l$, $M_\alpha = \{(i,j) \mid \beta_{i,j} = \alpha\}$, $\alpha \in \{0,1\}$.

Operator $B \in M(\{\widetilde{B}\})$ is called admissible if there exists at least one pair $(i,j) \in M_1$ such that for all pairs $(u,v) \in M_0$ $\Gamma_j(S^i) > \Gamma_v(S^u)$. This pair is called marked. It is proved also [3] that the greater value $d(i,j,B) = \min_{(u,v) \in M_0} (\Gamma_j(S^i) - \Gamma_v(S^u))$ is the smaller degree of item will be needed to construct the correct polynomial.

Thus in order to construct a correct algorithm of minimal complexity or to make inductive procedure of constructing it (like for example one in [4]), we need to find the algorithm of maximum $d(i,j,B)$ in some family of algorithms. This article is devoted to solving of maximization task in two particular subsets of ECA.

γ-optimization

First, denote by $\{B\}_\gamma$ the subset of ECA with the following parameters:

- $x_0 = 0, x_1 = 1$,

- Ω_A consists of all support sets of equal fixed power k. $p_i = 1/k, i = 1,...,n$,
- $\tilde{\gamma} \in [0,1]^m$,
- $\varepsilon_1,...,\varepsilon_n$ are fixed.

Let we have $(i,j) \in M_1$. The task is to find $\tilde{\gamma}^* \in [0,1]^m$ such that

$$\max_{B \in \{B\}_\gamma} \min_{(u,v) \in M_0} (\Gamma_j(S^i) - \Gamma_v(S^u)) = \min_{(u,v) \in M_0} (\Gamma_j(S^i) - \Gamma_v(S^u))|_{\tilde{\gamma} = \tilde{\gamma}^*}. \quad (7)$$

As shown in [1], in case of this special format of support vectors, the estimations (1)-(3) can be transformed into simple view:

$$\Gamma_j(S^i) = x_1 \Gamma_j^1(S^i) + x_0 \Gamma_j^0(S^i) = \Gamma_j^1(S^i),$$

$$\Gamma_j^1(S^i) = \frac{1}{Q_1} \sum_{S \in \tilde{K}_j} \sum_{\omega \in \Omega_A} \gamma(S^i) p(\omega) B(\omega S^i, \omega S) =$$

$$= \frac{1}{Q_1} \sum_{S \in \tilde{K}_j} \gamma(S)((\delta(S,S^i) \cdot p(\omega))V^1 + (\tilde{\delta}(S,S^i) \cdot p(\omega))V^0)$$

Here $\delta(S,S^i) \in \{0,1\}^n$ is the characteristic vector $\delta_u((a_1,...,a_n),(b_1,...,b_n)) = \begin{cases} 1, \rho_u(a_u,b_u) \le \varepsilon_u \\ 0, \rho_u(a_u,b_u) > \varepsilon_u \end{cases}$,

$\tilde{\delta}(S,S^i) \in \{0,1\}^n$ is its denial : $\tilde{\delta}_u = 1 - \delta_u$, $q(S,S^i) = \sum_{u=0}^{n} \delta_u$.

$V^1(S,S^i) = \sum_{u=0}^{\varepsilon} C_{n-q(S,S^i)}^{u} C_{q(S,S^i)-1}^{k-u-1}$, $V^0(S,S^i) = \sum_{u=1}^{\varepsilon} C_{n-q(S,S^i)-1}^{u-1} C_{q(S,S^i)}^{k-u}$.

So in the $\{B\}_\gamma$ family of ECA, the estimation $\Gamma_j(S^i) - \Gamma_v(S^u)$ is linear function on $\tilde{\gamma} \in [0,1]^m$, that is $\Gamma_j(S^i) - \Gamma_v(S^u) = L_{i,j,u,v}(\tilde{\gamma})$. So the task transforms into another one, i.e. to find $\arg\max_{\tilde{\gamma} \in \tilde{\gamma}^*} \min_{(u,v) \in M_0} L_{i,j,u,v}(\tilde{\gamma})$, there $L_{i,j,u,v}(\tilde{\gamma}) = \sum_{s=1}^{m} l_{s,i,j,u,v} \gamma_s$. This task in turn can be transformed into *t* tasks of linear programming, there $t = |M_0|$ (we enumerate all those linear combinations as $L_1,...,L_t$ in any order):

$$\tilde{\gamma}^* = \arg\max_{\tilde{\gamma} \in \{\tilde{\gamma}_1^*,...,\tilde{\gamma}_t^*\}} L_i(\tilde{\gamma}_i^*)$$

$$\begin{cases} \tilde{\gamma}_i^* = \arg\max_{\tilde{\gamma}} L_i(\tilde{\gamma}) \\ L_i(\tilde{\gamma}) \le L_1(\tilde{\gamma}) \\ ... \\ L_i(\tilde{\gamma}) \le L_t(\tilde{\gamma}) \\ \tilde{\gamma} \in [0,1]^m \end{cases}, \ i = 1,...,t.$$

These tasks can be solved with, for example, simplex method. So the precise solution of the initial task can be found.

γ,ε-optimization

The second task is more complex. As in previous chapter we choose parametrical subset $\{B\}_{\gamma,\varepsilon}$ of ECA first:

- $x_0 = 0, x_1 = 1$,
- Ω_A consists of the single support set (the method can be simply generalized to include cases of small number of support sets),
- $\widetilde{\gamma} \in [0,1]^m$,
- $\varepsilon_1,...,\varepsilon_n \geq 0$.

The task is the same as in previous section, i.e. to find in $\{B\}_{\gamma,\varepsilon}$ the algorithm with the maximum value of $d(i, j, B)$.

The algorithm for solving of this task consists of two parts. First one is the construction of auxiliary finite system of parallelepipeds P:

1. Build new sequence of objects $\{S'_1,...,S'_t\}$: for all $S \in \widetilde{K}_j$ add differences $S^i - S$ to the sequence.
2. Find the minimal system P of parallelepipeds $[-\varepsilon_1, \varepsilon_1] \times ... \times [-\varepsilon_n, \varepsilon_n]$ containing all different combinations of objects from $\{S'_1,...,S'_t\}$.

To construct the system P we must for all subsets $S \subset \{S'_1,...,S'_t\}$ find out if its combination is possible, i.e. if there exists any parallelepiped $E = [-\varepsilon_1, \varepsilon_1] \times ... \times [-\varepsilon_n, \varepsilon_n]$ such that $S' \in S$ if and only if $S' \in E$, and for all possible combinations add the minimal parallelepiped spanning it to the system. In practice there is no need to enumerate all different subsets of $\{S'_1,...,S'_t\}$. If we have found any impossible one, every combination containing it is impossible too.

The following theorem can be proved: $\max\limits_{\varepsilon \in [0,\infty)^n} \min\limits_{(u,v) \in M_0} (\Gamma_j(S^i) - \Gamma_v(S^u)) = \max\limits_{\varepsilon \in P} \min\limits_{(u,v) \in M_0} (\Gamma_j(S^i) - \Gamma_v(S^u))$.

Indeed for any ε-neighborhood $[-\varepsilon_1, \varepsilon_1] \times ... \times [-\varepsilon_n, \varepsilon_n]$ the maximum one from P containing in it will give not more estimations.

The second part is to calculate estimations themselves and solve the task. From (1)-(3) we have

$\Gamma_j(S^i) = g_1\gamma_1 + ... + g_n\gamma_n, g_k \in \{0,1\}, k = 1,...,n$,

$\Gamma_v(S^u) = g_1^{u,v}\gamma_1 + ... + g_n^{u,v}\gamma_n, g_s^{u,v} \in \{0,1\}, s = 1,...,n$.

And the difference is $\Gamma_j(S^i) - \Gamma_v(S^u) = g_1\gamma_1 + ... + g_n\gamma_n - g_1^{u,v}\gamma_1 - ... - g_n^{u,v}\gamma_n$. (8)

So the solution is $\widetilde{\gamma}^* : \gamma_s^* = \begin{cases} 1, & g_s = 1 \\ 0, & otherwise \end{cases}, \ s = 1,...,n$.

Indeed for any $\widetilde{\gamma} \in [0,1]^n$, difference (8) is smaller than $\Gamma_j^*(S^i) - \Gamma_v^*(S^u) = g_1\gamma_1^* + ... + g_n\gamma_n^* - g_1^{u,v}\gamma_1^* - ... - g_n^{u,v}\gamma_n^*$. The initial task transforms into finding $\arg\max\limits_{\varepsilon \in P} \min\limits_{(u,v) \in M_0} \Gamma_j(S^i) - \Gamma_v(S^u)$ and the precise solution can be found too.

Though the solution is precise the necessity to construct system P makes the task extremely difficult with multidimensional data. In order to make calculation faster we suggest proximate method for the same task.

The method starts with the parallelepiped spanning the whole sequence $\{S'_1,...,S'_t\}$. Then on every step we enumerate all admissible combinations of t-1 objects and leave the best one for next step, there we consider neighborhood spanning those best combination. Here t is the number of objects in current parallelepiped. The best combination is one that maximizes the value of $d(i, j, B)$.

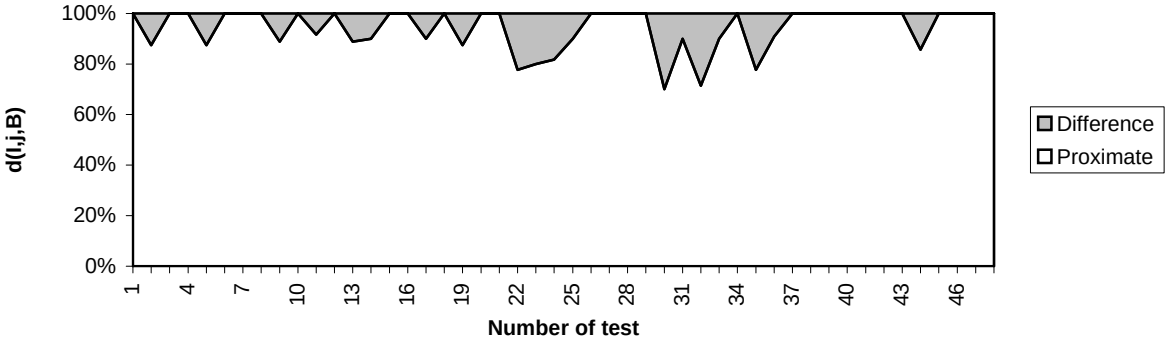
The following diagram shows results of hands-on testing of this method in comparison with the precise one. The table of descriptions of forty-eight patients was considered. It consists of three classes of correspondingly seventeen, twenty and twelve objects and thirty-three features. As the M_1 in turns every object was considered.

All other objects from its class were considered as the training sequence. All objects from other classes formed M_0. For example the twentieth object generated the following (20-th) test:

$M_1 = \{(20,2)\}$

$M_0 = \{(1,2),(2,2),...,(17,2),(38,2),(39,2),...,(48,2)\}$

$\{S'_1,...,S'_t\} = \{S_{18}, S_{19}, S_{21}, S_{22},..., S_{37}\}$.



It's easy to see that in most cases the precise solution or solution of acceptable precision has been found. And while the precise solution takes about two minutes to find (in case of twenty training objects and the difficulty extremely grows with increasing of their number), the proximate algorithm performs all forty-eight tests within about ten seconds.

Conclusion

In this article we have suggested the new approach for optimization of estimations calculating algorithms. It can be used for finding of the correct algorithm of the minimal complexity in the context of the algebraic approach for the pattern recognition.

Also we have considered two parametrical subsets of ECA and have find precise algorithms for solving optimization task for them.

Finally the fast proximate method with acceptable precision has been suggested.

Acknowledgements

The research described in this publication was made possible as a part of Grants 02-01-00558, 00-01-00650, 02-07-90134, 02-07-90137 from the Russian Fund of Fundamental Research, and INTAS 00-650, INTAS 00-370.

Bibliography

[1] Yu.I.Zhuravlev. Ob algebraicheskom podhode k resheniyu zadach raspoznavaniya ili klassifikacii // Problemy kibernetiki. 33, M: Nauka, 1978. (in russian)

[2] Yu.I.Zhuravlev. Korrektnye algebry nad mnozhestvom nekorrektnyh (evristicheskih) algoritmov // Kibernetika. 1977. 4. (in russian)

[3] Yu.I.Zhuravlev, I.V. Isaev. Postroenie algoritmov rasposnavania, korrektnyh dlya zadannoi kontrol'noi vyborki // Zhurnal vychislitel'noi matematiki I matematicheskoi fiziki, t.19, N3, may-june1979. (in russian)

[4] A.A.Dokukin. Induktivnyi metod postroeniya korrektnogo algoritma v algebrah nad modelyu vychisleniya ocenok // Zhurnal vychislitel'noi matematiki I matematicheskoi fiziki, 2003. (in russian)

Author information

Alexander A. Dokukin – Dorodnicyn Computing Centre of the Russian Academy of Sciences, Vavilov st., 40, Moscow GSP-1, 119991, Russia; e-mail: dalex@ccas.ru

SEGMENTATION OF A SPEECH SIGNAL WITH APPLICATION OF FAST WAVELET-TRANSFORMATION

T. Ermolenko

Abstract: *The article describes the method of preliminary segmentation of a speech signal with wavelet-transformation use, consisting of two stages. At the first stage there is an allocation of sibilants and pauses, at the second – the further segmentation of the rest signal parts.*

Key words*: wavelet-transformation, detailed elaboration, approximation, segmentation.*

Introduction

As known, the speech signal will consist of quasi-stationary parts corresponding to voice and sibilant phonemes, alternated by parts with rather fast changes of signal spectral characteristics (interphoneme transitions, explosive and occlusive phonemes, interword transitions speech - pause). It is possible to say, that the speech signal is characterized by nonlinear fluctuations of various scales. Therefore divisible analysis and wavelet – transformation is considered to be rather effective for the analysis of a speech signal.

Segmentation of a speech signal (SS) means allocation of signal parts corresponding to separate structural units of SS. Considering phonemes as such units the task of segmentation is reduced to detection of interphoneme transitions. Within the framework of traditional approaches the decision of this task is rather problematic. However WT allows to solve this problem at least for the phonemes corresponding to rather extensive quasi-stationary SS parts. The matter is that on interphoneme transitions the signal undergoes significant changes at once on many research scales, and, accordingly, is characterized by increase of wavelet-factors for many levels of detailed elaboration while on stationary parts of phonemes wavelet-factors appear grouped near to the certain scales. Thus, search of interphoneme borders can be reduced to search of moments of wavelet-factors increase for a significant amount of scaling levels.

Algorithm of sibilants and pauses border detection

Decomposition of lengths N of readout on SS wavelets makes the sum

$$f(t) = \sum_{k=0}^{\frac{N}{2^n}-1} s_{nk}\varphi_{nk} + \sum_{j=1}^{n}\sum_{k=0}^{\frac{N}{2^j}-1} d_{jk}\psi_{jk} \qquad (1)$$

thus n is a level of detailed elaboration, s_{jk}, d_{jk} - the factors of wavelet-decomposition, named average and differences accordingly (in this work we shall call them as factors of approximation and detailed elaboration) $\varphi_{jk} = 2^{j/2}\varphi(2^j t - k), j,k \in Z$, φ - scaling-function or scale function $\psi_{jk} = 2^{j/2}\psi(2^j t - k), j,k \in Z$, ψ - basic or "parent" wavelet.

In the researches we used fast SS wavelet-transformation, which was displayed on 6 levels, believing $s_{0,k}$ equal to readout of an initial signal. SS, digital with frequency of digitization of 22050 Hz is broken into overlapped windows in size 20 ms with half overlap windows.

Apparently from figure 1, 2, 3 for allocation of pauses and sibilants the signal has enough information on behavior of detailed elaboration factors on the 6-th level as for them the small amplitude in comparison with other signal parts is typical.

We build numerical sequence $\{a_{i6}\}_{i=1}^{N/256}$:

$$a_{i6} = \sum_{k=0}^{n_6-1} d_{6,i+k}^2 ,$$

where i - number of a sliding window, $n_6 = \frac{n}{2^6}$ - the size of a sliding window on the 6-th level, n - the size of a window in an initial signal (512 readouts).

The prospective beginning of sibilant (pause) is placed in the beginning of i windows for which the condition $a_{i-1,6} \geq 1000, a_{i,6} < 1000$ is implemented. It is obvious, that at the end boarder of sibilant (pause) this condition is carried out just the other way. The threshold has been received experimentally and is independent of announcer.

Results of work of algorithm with the word *обеспечение* made by the announcer - man (figure 1), and the announcer - woman (figure 2), and also with a word *кошачий*, pronounced by the announcer - woman (figure 3) are shown below. Last case shows the work of a method in conditions of a significant noise (relation signal / noise makes 15 decibel).

For this algorithm simplicity of realization, independence of announcer, low sensitivity for noise is characteristic. One of properties of this algorithm is reference to a pause of the site corresponding to mute vowel.

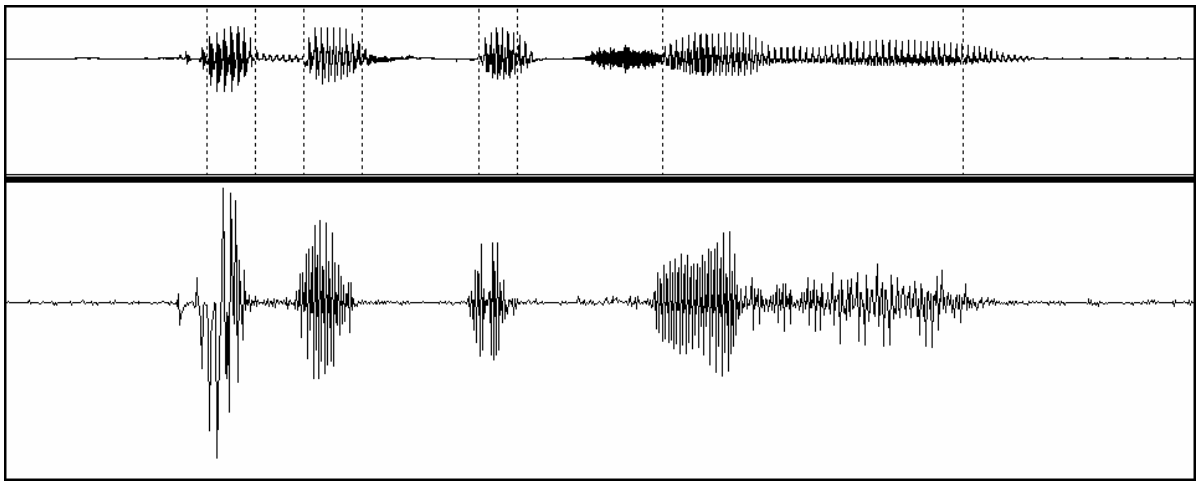


Figure 1. Segmentation of a word *обеспечение* made by the announcer - man. Above – amplitude-time representation of a signal, below – factors of detailed elaboration of the 6-th level.

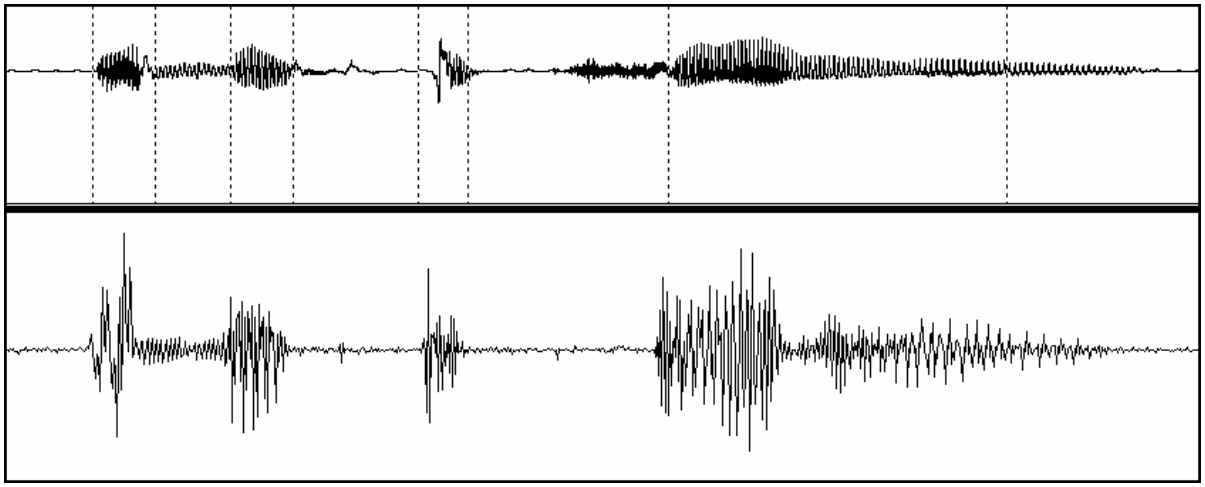


Figure 2. Segmentation of a word *обеспечение* made by the announcer - woman. Above – amplitude-time representation of a signal, below – factors of detailed elaboration of the 6-th level.

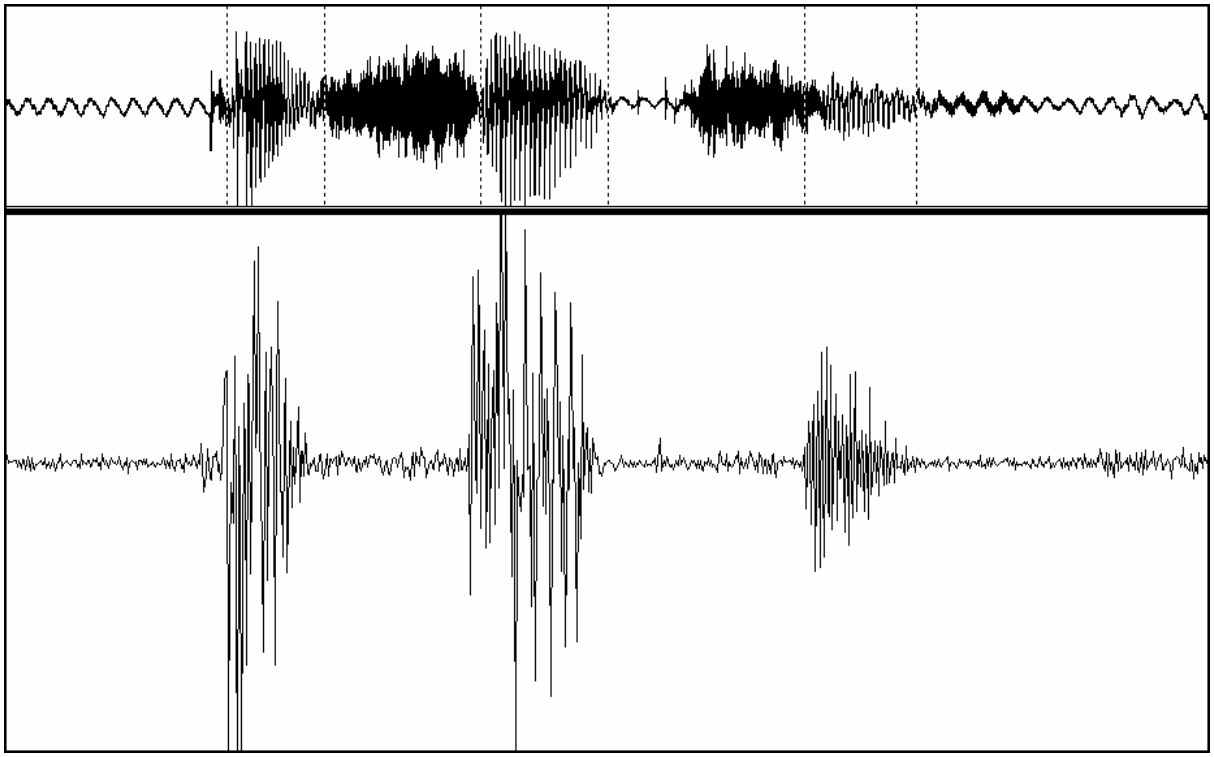


Figure 3. Segmentation of a word *кошачий*, made by the announcer - woman. Above – amplitude-time representation of a signal, below – factors of detailed elaboration of the 6-th level.

Segmentation of signal parts containing vowels, sonorous concordants and fricatives

The following criterion for a choice of the most informative level of decomposition has been received by the experimental way. For j level of decomposition, since the 3-rd one, the inequality implementation is checked:

$$\frac{E}{N} < \frac{E_j}{N_j}, \tag{2}$$

where N_j - amount of detailed elaboration factors at j level, more than 0,5;

$$E = \sum_{i=0}^{N-1} s_{0,i}^2; \qquad E_j = \sum_{i=0}^{\frac{N}{2^j}-1} d_{j,i}^2 .$$

The first level for which the condition (2) can be implemented is the most informative.

The behavior of detailed elaboration factors was analyzed in the following way: at the chosen level of decomposition j for a segment, which is not related to a sibilant (pause), the numerical sequence $\{e_{ij}\}_{i=1}^{l}$ was built:

$$e_{ij} = 10 \lg \sum_{k=0}^{n_j-1} d_{j,i+k}^2 ,$$

where i - the number of a sliding window, $n_j = \frac{n}{2^j}$ - the size of a sliding window at j level, n - the size of a window in an initial signal (512 readouts), l - amount of windows in an examined segment.

Further averaging sequence on 3 values was carried out.

Borders of prospective segments were put down between windows with numbers i and $i+1$, for which $\left|e_{i+1,j} - e_{i,j}\right| \geq 3.5$.

Lacks of algorithm work are those:

1. In stressed and unstressed *-а-* the superfluous segment can be allocated;
2. In some cases two vowel sounds, standing close are not distinguished, for example, *-оа-*, *-ио-* in words *коала, миллион*; the characteristic ending *-ия-, for example, квалификация, аппроксимация*, also is not segmented, that is explained, as *-ия-* is graded and sounds, as unstressed *-а-*.
3. Resonant sounds in combination with unstressed vowel are not divided among themselves.

In figure 4 the result of work of algorithm with a word *акселерация* after separation of sibilants and pauses is shown. Apparently from figure, the segment 1 corresponds to unstressed *-а-*. The pause is precisely enough separated from the speech. The segment 2 contains sounds *-к-* and *-с-*. In the 3-d, unstressed *-е-* and *-л-* are combined, that can be explained to some degree: the voice sound is reduced, has short duration and loses its qualities. The second unstressed sound *-е-* is well separated by borders of the 4-th segment; obviously, it is connected with the fact that it is to the first pretonic vowel. The segment 5 corresponds to *-р-*, the 6-th – to stressed *-а-*. The 7-th segment contains sound *-ц-*, the 8-th segment comprises the unstressed ending *-ия-*.

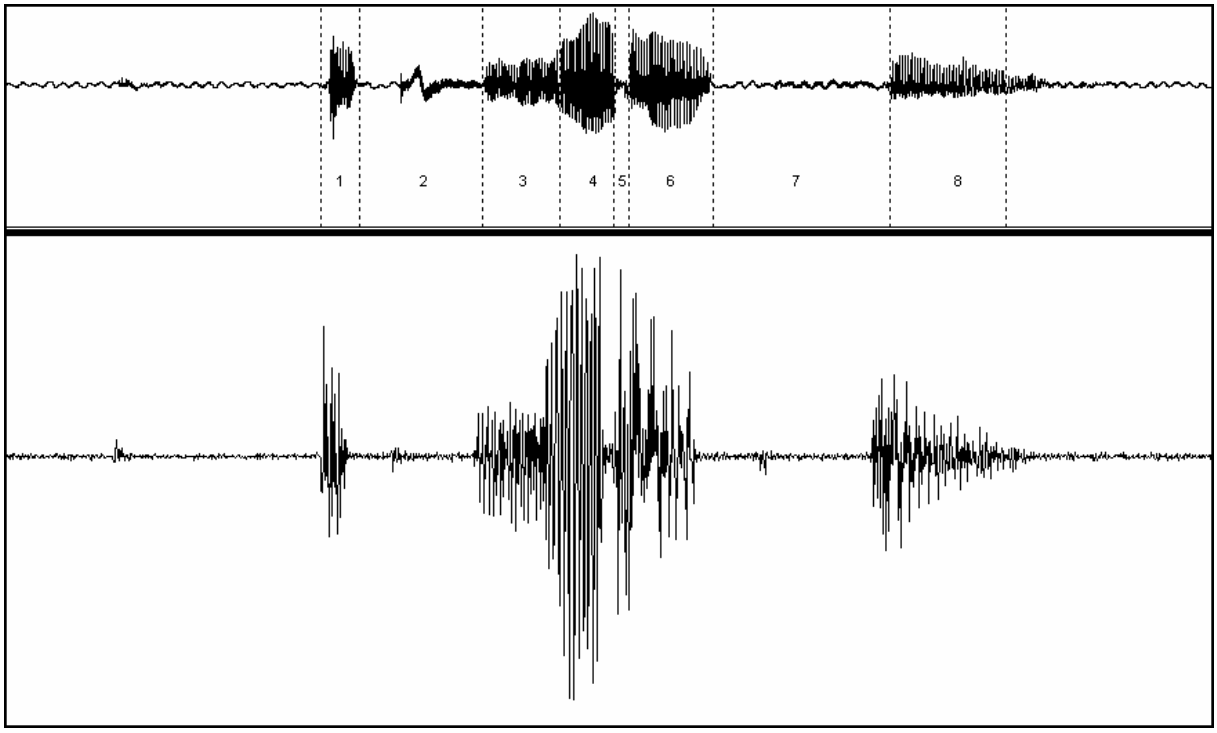


Figure 4. Segmentation of a word *акселерация* after separation of sibilants and pauses. Above – amplitude-time representation of a signal, below – factors of detailed elaboration of the 6-th level.

Conclusion

In the work the method with wavelet-transformation use that segments SS in two stages is described. Firstly, sibilants and pauses that are identified among them are allocated. For this stage simplicity of realization, independence of announcer, low sensitivity noise are characteristic. At the second stage the rest parts of a signal are segmented. Results of its performance are less reliable, such mistakes as occurrence of superfluous borders in vowels and non separation of resonant *-л-, -м-, -н-* from *-и-, -е-* in some cases are characteristic. Thus, there is a necessity of more detailed research of behavior of wavelet-factors at all levels of scaling in view of relative amplitude of a site, frequency attributes, duration of a segment and similar attributes of the nearest next sites.

Literature

1. Dremin I.M., Ivanov O.V., Nechitajlo V.A. Wavelets and their Use. // Successes of Physical Sciences, v. 171, №5 p. 465-500, 2002.
2. Astafjeva N.M. Wavelet-analysis: Bases of the Theory and Examples of Application. // Successes of Physical Sciences, v 166, №11 p. 1145-1170, 1996.
3. Detection of Change of Properties of Signals and Dynamic Systems. Under M.Bassvil, A.Banvenist's edition. Moscow, "Mir", 1989.
4. Walker J. Fourier Analysis and Wavelet Analysis. // Notices of the AMS, vol. 44№6, p. 658-670, 1997
5. Daubechies I. Ten lectures on wavelets. // Philadelfphia: SIAM, 1991

Authors information

Ermolenko Tatyana – Institute of Artificial Intelligence, B.Hmelnitsky avenue, 84, Donetsk - 83050, Ukraine e-mail: etv@iai.donetsk.ua

METHODS OF COLOR IMAGES PROCESSION FOR FURTHER IDENTIFICATION OF THE OBJECT

Gariachevskaja I.V., Kuziomin A.Ya.

Abstract*: The offered paper deals with the problems of color images preliminary procession. Among these are: interference control (local ones and noise) and extraction of the object from the background on the stage preceding the process of contours extraction. It was considered for a long time that execution of smoothing in segmentation through the boundary extraction is inadmissible, but the described methods and the obtained results evidence about expedience of using the noise control methods.*

Keyword*: NOICE REDUCTION, FILTERING, SEGMENTATION, EDGE DETECTION*

Introduction

A number of the images' procession problems are connected with a search of specific form objects on a complex multiobject image. Among these problems are, for example, detection and recognition of the target on the television observation systems, detection and tracking of the transport means movement in the real conditions in the real time, recognition of symbols randomly located on the sheet of paper (for example, letters and conventional signs on the geographical maps) and many others. The key features of this type of problems is: the presence of a small number of standard images (a car silhouette, a detail profile); an essential characteristics of the object are its form and not linear dimensions, position in space etc. To perform recognition i.e. comparison of standard images with the object's boundaries, it is necessary to perform transition to the figures' contour images. The given work is devoted to the problems of segmentation, noise filtration, the object extraction from the background, definition of boundaries for the subsequent identification of the object. The problems of recognition will not be considered directly.

Review

While on the subject of filtering it should be noted that the investigations being performed are directed to the choice of smoothing methods used on the stage preceding the contours extraction. The main problem is to preserve the structure of the objects and to avoid colors distortion.

The simplest noise suppression algorithm is the sliding averaging filter. A mask of some dimension is generally chosen for smoothing and the smoothed image is built in the form of convolution of the initial image with a mask, for example, as follows:

$$F_1 = \frac{1}{9}\begin{pmatrix}1 & 1 & 1\\ 1 & 1 & 1\\ 1 & 1 & 1\end{pmatrix} \quad F_2 = \frac{1}{10}\begin{pmatrix}1 & 1 & 1\\ 1 & 2 & 1\\ 1 & 1 & 1\end{pmatrix} \quad F_3 = \frac{1}{16}\begin{pmatrix}1 & 2 & 1\\ 2 & 4 & 2\\ 1 & 2 & 1\end{pmatrix}$$

It is easy to note that each of the masks meets the main requirement to the masks of the smoothing filters, namely: the central element of each mask is not less than any of the rest and the sum of all elements taking into account the coefficient is equal to 1. All the elements making part of the mask are positive numbers.

Gauss filter is one of the most popular filters used as a component of many algorithms for boundaries' extraction. Significant noise suppression takes place in its operation, but the image structure changes greatly. In particular, thin lines (particularly those ones whose thickness is two pixels) and sharp corners are distorted significantly.

To suppress the local noise the median filters are used, the algorithm of sorting and ordering of massive elements (one dimensional or two-dimensional) either by non-decrease or lack of increase. The meaning of the color found in the central place of the window as a result of permutation is called the median. Just this meaning substitutes the color of the filter raster being at the given moment under the central element of the median filter. The given filter is intended for suppression of the local noise, but with small dimensions of the window the pulse noise suppression takes place (in this case the meaning of the window central point color is often not taken into account). One of the greatest shortages of the median filter is the sharp corners of the image structure "rounding off"; furthermore, the lines whose width doesn't exceed one half the window width disappear, this is really unacceptable with further extraction of the contours.

All the above methods of noise suppression cannot be used in the case of further boundary extraction as the preliminary noise suppression doesn't emphasize the objects' boundary on the image but fuzzify them. But the contour extraction from the initial image gives many redundant contours; solution can be found in using the smoothing filters intended for noise removal with the image structure preservation.

The SUSAN filter (S.M. Smith and J.M. Brady) was developed for structure preserving noise reduction. The main object used in the SUSAN filter is a round mask with the central pixel named the "nucleus". The points whose brightness is approximately equal to the nucleus brightness form the so-called "Univalue Segment Assimilating Nucleus" or USAN. The concept of some point association with the local area having the similar brightness is the basis of the SUSAN method. This region includes much important information on the structure of the image around some point. From the dimension and location of the SUSAN region the boundaries (one- or two-dimensional characteristics) and corners, intersections (two-dimensional characteristics) can be defined. As can be noted, the dimension of the region is a maximal one when the nucleus of the mask is located on the smooth part of the image, it is equal to one-half the area of the mask when the nucleus is located on the smooth boundary and is a minimal one inside the corner.

The SUSAN noise suppression algorithm as well as some available technologies of image filtering preserves the image structure at the cost of smoothing with only those neighbors that make part of the "homogeneous region" with the central pixel. The SUSAN filter calculates the mean value of all the points in the area inside the USAN. Generally the central point is excluded from consideration, this allows to suppress weak pulse deviation of brightness. It is evident that the USAN includes the greatest number of the suitable neighbors for finding mean value not affecting any points of independent area. By this means the image structure is preserved.

In respect of eliminating the limits fuzzing on the image it is clear that on the fuzzion boundaries the points will be contracted into the area where their values are closer. Thus, the SUSAN filter can improve significantly the quality of the image not changing its structure. Another peculiarity is that using the filter iteratively it is possible to achieve more powerful noise suppression without the image quality loss (generally more powerful noise suppression corresponds to greater smoothing of the initial image structure). Moreover, the level of smoothing

doesn't affect the characters' localization, this advantageously differentiates it from other filters, for example, median one. The fact that the suppression of the pulse noise with the given filter is hampered (in the case of the pulse noise emergence the USAN area is equal to 1 i.e. it includes only the nucleus and that is why no variations with this point will be referred to its shortages).

Furthermore, the investigation was directed to obtaining the image gradient, it is also called the spatial differentiation or, more simply, the contour extraction with different methods. To detect and exclude the contours on the image (obtaining of the gradient image) the differential operators are used as a rule not higher than of the second order [4]. In the general form such operator is written in the following way:

$$a_{11}\frac{\partial^2 f(x,y)}{\partial x^2}+a_{12}\frac{\partial^2 f(x,y)}{\partial x\partial y}+a_{22}\frac{\partial^2 f(x,y)}{\partial y^2}+a_{10}\frac{\partial f(x,y)}{\partial x}+$$

$$+a_{01}\frac{\partial f(x,y)}{\partial y}+a_{00}f(x,y)=z(x,y),$$

where $a_{ik}; i,k=0,1,2$ – are prescribed coefficients.

If, for example, we assume $a_{11}=a_{22}=1$ in this expression, and the rest coefficients equal to zero then we'll receive Laplace operator which is widely used in image procession.

In the majority of the similar cases the contours exclusion the masks with the dimensions (3*3) are used as the most optimal ones. Such widely used operators as the operators of Roberts, Kirsh, Sobel Previtte, the differential operator and others functioned according to one and the same principle. When extracting the contour with the given methods the convolution of the initial image by each mask separately is performed. After this the vector norm is defined Depending on what method of the vector norm finding is chose the output image is finally formed.

Thus, for example, one of the most widespread masks [2] was applied to the Roberts operator

$$H_1^{(1)}=\begin{pmatrix}0&0&0\\0&1&0\\0&0&-1\end{pmatrix},\quad H_1^{(2)}=\begin{pmatrix}0&0&0\\0&0&-1\\0&1&0\end{pmatrix};$$

the convolution of the initial image is performed by each mask separately using the formula: $d_1(i,j)=B(i,j)*H_1^1$; $d_2(i,j)=B(i,j)*H_1^2$.

After the vector norm is defined using one of three possible ways:

- geometric sum of the components d_1 and d_2 is found;
- the components with the maximum module $|d_1|$ and $|d_2|$ are chosen;
- the algebraic sum $|d_1|+|d_2|$ is found;

The output image is formed depending on what way of finding the vector norm was chosen. To improve the final results a number of procedures is performed such as preliminary smoothing, making the obtained contour contrast, etc.

The SUSAN [5] operational principles differ significantly. The main object used in the SUSAN filter is a round mask with a central pixel named a "nucleus". The points whose brightness is approximately equal (similar) to the brightness of the nucleus make up the so called "Univalue Segment Assimilating Nucleus" or the USAN. The concept of association of some point with a local area having such brightness is the SUSAN method basis. This area contains many important information about the structure of the image around some point From the dimension and location of the SUSAN area the boundaries (one-dimensional characteristics) and corners, intersections (two-dimensional characteristics) can be defined. As can be noted the dimension of the area is a maximal one when the center of the mask is on the plane part of the image, it is equal to one-half the mask area when the nucleus is on the plane boundary and it is minimal inside the corner.

$$c(\vec{r},\vec{r}_0)=\begin{cases}1, \text{если } |I(\vec{r})-I(\vec{r}_0)|\le t\\0, \text{если } |I(\vec{r})-I(\vec{r}_0)|> t\end{cases},$$

where $\vec{r}_0$ – is the position in the nucleus of two-dimensional image, $\vec{r}_0$ – is the position of any other point within the mask, $I(\vec{r})$ – is the brightness of the point, t – is the brightness threshold which for ordinary monochromatic images is ±27 gradations, and c – is the comparison result. The comparisons are performed for every point of the mask and then the total value of n is calculated using the formula $n(\vec{r}_0) = \sum_{\vec{r}} c(\vec{r}, \vec{r}_0)$.

This total value is the area of the USAN region. t parameter defines the minimal contrast of characteristics and the maximal depth of the ignored noise.

Then n is compared to the fixed threshold g (geometric threshold), whish is set on $3n_{\max}/4$, where $n_{\max}$ – is the maximal value taken by n. The preliminary result is obtained using the following rule

$$R(\vec{r}_0) = \begin{cases} g - n(\vec{r}_0), \text{если } n(\vec{r}_0) < g \\ 0, \quad \text{иначе} \end{cases}.$$

The principle of the SUSAN consists in the following: a greater result corresponds to a smaller region.

When the boundaries are found in the absence of noise the geometric threshold is unnecessary. But to obtain the optimal noise suppression the threshold is set on $3n_{\max}/4$. The use of the threshold g will not result in the screening of the right limits for the following reasons. If the staircase boundary is detected the value n will be either less or equal to $n_{\max}/2$ from any side of the boundary. Thus, the boundary will not be rejected. If the edge is not an ideal staircase boundary and its form is fuzzy, n will be even less, that is why the risk of rejection of the limit is lower.

The cited algorithm gives good results, but more sensitive expression for *c*, is $c(\vec{r}, \vec{r}_0) = e^{-\left(\frac{I(\vec{r}) - I(\vec{r}_0)}{t}\right)^6}$.

This expression makes it possible to consider the pixels' brightness variation more flexibly.

The results received when using this method are more satisfactory, less details of the background are observed as contours. But with suppression of the whole background this method failed.

Realization

The above filters feature significant shortages, this doesn't allow to use them in the initial form in the general case. The problem of noise suppression can be solved through integration of their best properties.

The SUSAN filter was chosen for the basis of the developed method of noise suppression on the color images as the most preserving the structure of the initial image. But the given filter is not intended for pulse noise elimination that is why the following solution was offered:

- An image is processed with the SUSAN filter;
- In the case when the USAN region is less than some threshold ξ, the two-dimensional median filter with a window dimension of 3 by 3 square and the center in the SUSAN nucleus point is used.

The obtained method offers all merits of the SUSAN filter, in particular, it preserves accurately the image structure (boundaries of the homogeneous regions, lines and their intersections, corners), in this case both the form and properties distribution are preserved. The filter suppresses the pulse noise well, using it iteratively it is possible to achieve the noise level close to zero. In this case it doesn't inherit shortages of the median filter such as smoothing of corners and thin lines' elimination

Investigations showed that extracting the edges directly from the color image we often have not only the contour of the object and its details but a great number of the background details' contours as well, particularly if it is a complex background. Solution of such a problem was found. An algorithm allowing to perform segmentation and to extract the whole object from the background was developed.

The algorithm's operation is based on the concept of the hue. As the input image is presented in the RGB system it is necessary to transform it into the HSB format. Further on the hue component is extracted and a histogram

$H(h)$ is built with it, where h – is a hue. In this case the background hue will correspond to the maximum on the histogram, and the hue of the object that should be localized will correspond to the second maximum $M = H(m)$. Then it is necessary to choose the range of the hue $[t_1, t_2]$ values, where all the points of the localized object are included. Values of t_1 and t_2 are calculated by the formulas

$$t_1 = \min(m - b_1, m_1)$$

$$t_2 = \min(m + b_2, m_2)$$

where b_1 and b_2 – is the maximum spread in values of the hue, m_1 – is the maximal value of the hue not exceeding m, where $H(m1) = M * k$ ($M * k$ – is a number of points sufficient for the dimension of the area occupied by them can be a part of the object's monochromatic region). The parameters $b_1 = b_2 = 15,\ k = 0.347$ acceptable for the majority of images were established as a result of investigations.

The following step of the algorithm is separation of the image region, its points background is in the $[t_1, t_2]$ range established in the previous step.

The obtained image includes the local noise regions, which are eliminated with the median filter.

Then it is necessary to divide the image plane into rectangular segments. All the segments of the initial image, which non-empty segments on the image obtained in the previous stage correspond to, remain without variation; the rest segments are removed

The obtained region is the most possible place of the localized object emergence.

Having sharp boundaries of the object it is possible to separate then the contours of the received image including only the object itself and insignificant noise using any of the known methods.

Experimental investigations

This exemplar is meant to be a model for manuscript format. Please make your manuscript look as much like this exemplar as possible. Except for formatting and the inclusion of an abstract, authors have complete freedom to structure their papers as they wish; the section headings need not be as given in this exemplar. In the case of serious deviations from the format, the paper cannot be published.

The considered experiments as well as the research work were directed to solution of two problems: to carry out filtering and to extract the object from the background with further extraction of the contours which in their tern will be used further on as evidences for further objects identification.

Investigations were performed both with the test problems and real images.

Fig.1 shows the initial image, the red car on a complex background and results of different filters operation work as well as the received contours.

Both filters decreased a number of lines. It is difficult to define visually which of them functions better and preserves better the image structure But if the procedure of smoothing is performed several times (8 times for the SUSAN filter, 6 times for smoothing filter) even a visual estimate testifies that the image appears fuzzy after the smoothing filter whereas after repeated use of the SUSAN filter the object's image stands out sharply against fuzzy background. The results are given in Fig. 2 a) and b). The contours extracted subsequently appear quite differently. After the smoothing filter the greater part of the contours is lost. The results are given in Fig. 2 c) and d).

Let us consider one example more. The results of the automatic object extraction algorithm functioning and consequent extraction of the contour are given in Fig.3.

In spite of the fact that part of the details of the object was last (wheels, headlights…) it is possible to indicate that this algorithms gives satisfactory results owing to its functioning. The contour of the car body is a clear closed contour and the contour of the car body is one of the main standards making it possible to classify a car. The absence of the background will increase the object recognition rate.

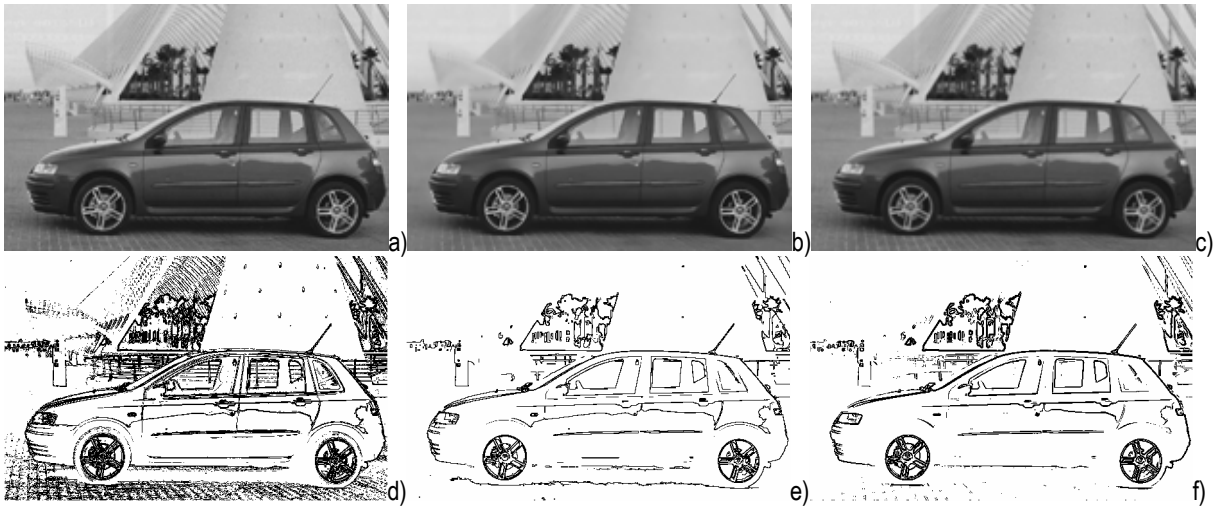


Fig.1.a) initial image; b) image smoothed with the SUSAN filter; c) image received after smoothing filter; d) contours separated from the real image; e) contours separated from the image smoothed with the SUSAN filter; f) contours separated after the smoothing filter

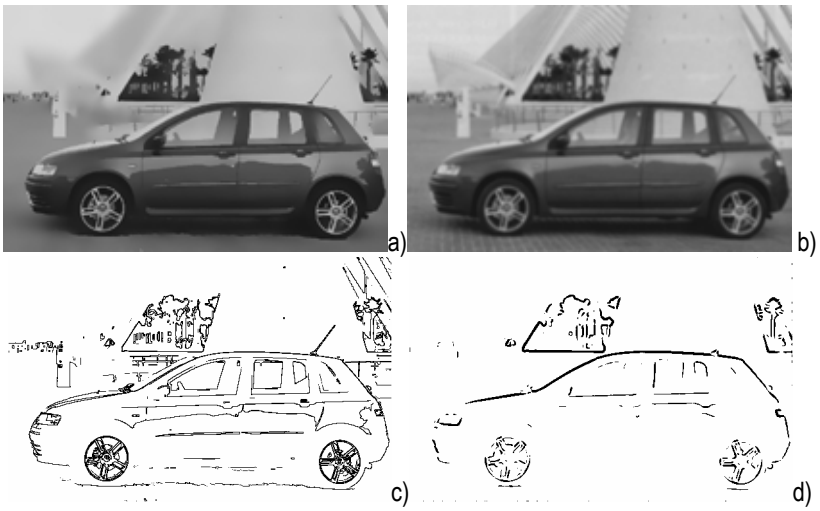


Fig. 2. a) the image after 8-fold use of the SUSAN filter; b) the image after 6-fold use of the smoothing filter; c) the contour obtained after using the SUSAN; d) the contours obtained after using the smoothing filter.

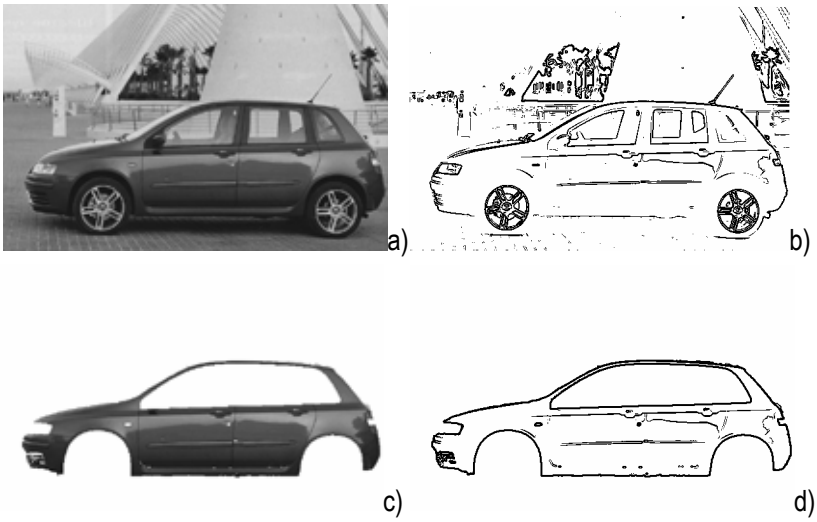


Fig. 3 Results of obtaining image gradient with the offered algorithm. The initial image (a); the contours extracted from the initial image (b); the object's localized region (c); extracted region contours (d).

Conclusion

The performed research showed that the considered methods produce “soft” smoothing preserving the image structure. Only uniform parts of the image are subjected to smoothing, edges and corners of the objects remain clear and not fuzzy. Application of the given method is possible both to the problems of the preliminary image procession preceding the process of segmentation through limits extraction and to the increase in the image quality, for example, photos.

The objects’ extraction by the color hue gives a good result both for the problems where it is necessary to extract an object from the background and for the problems where it is necessary to pass to the space of indications, to get the objects’ contours. The absence of noise in the form of the background will make it possible to perform the recognition quicker.

There are some shortages here: the given method functions only if one sufficiently big object (as compared to the background details) is on the image. The biggest region with one color is extracted automatically.

Bibliography

[1] Panchenko D.S., Putiatin E.P. Comparative analysis of images segmentation methods. “Radio Electronics and Informatics”, 2002

[2] Gariachevskaja I.V., Kuziomin A.Ya. Extraction of objects’ contours on the color image. “Radio Electronics and Informatics”

[3] Putiatin E.P., Averin S.I. Image procession in robotics. – M.: Mashinostrojenie, 1990.-320p.

[4] Yanshin V.V. Analysis and procession of images: principles and algorithms.- M.: Mashinostrojenie. 1995. –112 p.

[5] S.M. Smith, J.M. Brady SUSAN – A New Approach to Low Level Image Processing. 1995.

Author information

Gariachevskaja I.V. – KhNURE, 14 Lenina Ave., Kharkov, Ukraine, 61166; e-mail: ga_irisha@mail.ru

Kuziomin A.Ya. – KhNURE, Prof.; Director of Information Marketing Department; 14 Lenina Ave., Kharkov, Ukraine, 61166; e-mail: kuzy@kture.kharkov.ua

ROC CURVES WITHIN THE FRAMEWORK OF NEURAL NETWORK ASSEMBLY MEMORY MODEL: SOME ANALYTIC RESULTS

P. M. Gopych

***Abstract**: On the base of convolutional (Hamming) version of recent neural network assembly memory model (NNAMM) for intact two-layer autoassociative Hopfield networks maximum-likelihood receiver operating characteristics (ROCs) have been obtained analytically. A method of taking explicitly into account a priory probabilities of alternative hypotheses on the structure of information initiated memory retrieval and modified ROCs (mROCs, unconditional probability of correct recall vs. false alarm) are introduced. The comparison of observed and calculated ROCs and mROCs demonstrates that they coincide quantitatively and in this way intensities of cues used in appropriate experiments may be estimated. It is found that basic ROC properties (one of experimental findings underpinning dual-process models of recognition memory) can be explained within one-factor NNAMM.*

***Keywords**: ROC, mROC, memory, neural network, cue index, recall, recognition, signal detection theory.*

1. Introduction

Receiver operating characteristics (ROCs or ROC curves) are widely used in classic signal detection theory for providing the performance of linear Fisher or Euclidian classifiers for different values of their thresholds and plot the probability of correct detection of a noisy signal as a function of the probability of its false detection or false alarm [1]. Usually, it is assumed that distributions of initial patterns (vectors) conditioned on the presence or absence of sought-after signal are Gaussians with the same variances and a specific distance between them; for both hypotheses their prior probabilities are chosen being equal, ½. In particular, in neuroscience such ROCs are used in data analysis where encoding and processing of sensory information in single and multiple neuronal spike trains are studied [2]. Last years a method for deriving ROCs by means of human memory testing was developed but before the present there exists no computer memory model which might be able to reproduce empirical ROCs neither qualitatively no quantitatively [3]. For this reason understanding of observed ROC curves is respected as one of the most important unresolved problems in the field of computer memory modeling [4].

In contrast to abstract computer models, neurobiology models directly address the question on functional nature and neuroanatomical substrates of different kinds of memory. For example, now recognition memory is hotly debated within dual-process models (DPMs) which are viewed recognition as consisting of two components, recollection and familiarity [3,5,6]. Recollection is thought of as an event where a person recalls as particular stimulus (e.g., a human face) as episode where it was early encountered and familiarity respects to the person's experience (or feeling) that particular stimulus was early encountered but without specific memory about where, when, or why it was happened. It is claimed [3] that DPMs are supported by many results of cognitive, neuropsychological, and neuroimaging memory studies but in spite of long history of research even basic properties of DPMs are ambiguously defined and rather often even their basic terms are used by different authors in different ways [3]. Additionally, DPMs are not specified at computational level because most computer models consider recognition as one- not as two-factor process (although see [6]). On the other hand, none of computer models does not describe all body of recognition memory traits (e.g., ROCs) and for this reason their separate inferences which are not consistent with predictions of DPMs cannot be viewed as convincing arguments against them.

In present work using convolutional (Hamming) version of neural network assembly memory model (NNAMM) [7,8] analytic formulae for maximum-likelihood ROC calculations are derived and shown that ROCs, as one of experimental findings underpinning DPMs, can be explained within NNAMM without assumption that recognition memory is a dual process. A method of taking explicitly into account prior probabilities of hypotheses on the structure of information initiated memory retrieval is proposed; on this base modified ROCs (mROCs, unconditional probability of correct recall vs. false alarm probability) and overall probability of memory trace recall/recognition were introduced. It has been found that comparison of calculated and observed ROCs (or mROCs) provides a method for extraction from empirical ROCs (or mROCs) of intensities of cues used in appropriate experiments.

2. Some NNAMM Backgrounds

According to NNAMM (see ref. 8 for details), components of initial ternary vectors take their values from the set –1,0,1 and most of these values are 0s (that is so called sparse coding). After initial data preprocessing ternary vectors are transformed into binary feature vectors with components –1 or 1 (that is so called dense coding). Actually, feature vectors are quasibinary ones because their spinlike (–1,1) components cannot be shifted to other (0,1) binary representation and they could manifest (although do not manifest) their third, zero, components. Below only quasibinary vectors are considered but for short the preposition "quasi" will be omitted.

Neural network (NN) assembly memory is constructed from interconnected (associated) and equal in rights assembly memory units (AMUs) and basic properties of assembly memory in a whole depend on the properties of its components, AMUs. AMU has original architecture and consists of regular Hopfield two-layer autoassociative NN as its central element, N-channel time-gate, additional reference memory, and two nested feedback loops [8].

NN related to particular AMU is subserved by binary vectors mentioned. We refer to such an N-dimensional arbitrary vector as x. If x represents information stored or that should be stored in AMU we term it x_0. Random vector or binary noise x_r we define as x with components –1 or 1 randomly chosen with uniform probability, ½.

Damaged reference vector $x(d)$ is defined as x_0 with its damage degree d. The components $x_i(d)$ of $x(d)$ are defined as

$$x_i(d) = \begin{cases} x_0^i, & if \quad u_i = 0, \\ x_r^i, & if \quad u_i = 1 \end{cases} \quad i = 1,..,N \tag{1}$$

where u_i are marks whose magnitudes 0 or 1 are chosen randomly with uniform probability and fixed d:

$$d = \sum u_i / N, \quad i = 1,..,N. \tag{2}$$

If the number of marks $u_i = 1$ is m then $d = m/N$; $0 \le d \le 1$; $x(0) = x_0$ and $x(1) = x_r$. Damage degree d is a fraction of noise in vector $x(d)$ while intensity of cue or cue index $q = 1 - d$ is a fraction of correct, undamaged information about x_0 in $x(d)$ [7,8]. The data coded in such a way naturally arise when line or half-tone images are binarized to solve a very important problem of local feature discrimination from smooth background and noise [9].

Expressions 1 and 2 define an original data coding procedure [7]. To design appropriate data decoding rules we explore two-layer auto-associative NN with N cells in its entrance (or exit) layer, entrance and exit cells connected by "all-to-all" rule, and McCalloch-Pitts model neurons with rectangular response and triggering threshold θ.

Following ref. 10 for perfectly learned intact Hopfield NN, the elements w_{ij} of *synapse matrix w* are defined as

$$w_{ij} = \eta x_0^i x_0^j \tag{3}$$

where $i,j = 1,..,N$; $\eta > 0$ is a *learning parameter* (below it is supposed that $\eta = 1$); x_0^i, x_0^j are the components of reference vector x_0 (all w_{ij} may differ from each other in sign only). It is needed to stress that NN with synapse matrix w is learned to remember *only one* memory trace x_0 and we *deliberately* reject the available possibility of storing other traces in the same NN. Also we posit that an input vector x_{in} is decoded successfully if learned NN transforms x_{in} into output vector $x_{out} = x_0$ [7,8,9].

The transformation algorithm is the following. For the jth neuron of the NN exit layer an *input signal* h_j is given by

$$h_j = \sum w_{ij} v_i + s_j \tag{4}$$

where v_i is an *output signal* from the ith neuron of the NN entrance layer; $s_j = 0$. The signal v_j for the jth neuron of NN exit layer (the jth component of x_{out}) is calculated according to the model neuron's response function (signum function or 1 bit quantifier) as

$$v_j = \begin{cases} +1, & if \quad h_j > \theta \\ -1, & if \quad h_j \le \theta \end{cases} \tag{5}$$

where for $h_j = \theta$ the value $v_j = -1$ was arbitrary assigned.

3. Convolutional and Hamming Versions of NNAMM

If $h_i = x_{in}^i$ then from Expression 5 follows that $v_i = x_{in}^i$. From this fact and Equations 3 and 4 for jth exit layer neuron we have: $h_j = \sum w_{ij} x_{in}^i = \eta x_0^j \sum x_0^i x_{in}^i = \eta x_0^j Q$ where $Q = \sum x_0^i x_{in}^i$ is a convolution of vectors x_0 and x_{in} ($-N \le Q \le N$). The substitution of $h_j = \eta x_0^j Q$ into Expression 5 gives that $x_{out} = x_0$ and reference vector is successfully decoded if $Q > \theta$ (if $\eta \ne 1$ then $Q > \theta/\eta$). Hence, above NN algorithm is equivalent to the convolutional decoding algorithm although in present form it is valid only for perfectly learned intact NNs (see details in ref. 8).

Since for each x_{in} exists such a vector $x(d)$ that $x_{in} = x(d)$, inequality $Q > \theta$ can be written as a function of $d = m/N$ and as a result

$$Q(d) = \sum_{i=1}^{N} x_0^i x_i(d) = \sum_{i=1}^{N-m} (x_0^i)^2 + \sum_{i=1}^{m} x_0^i x_r^i = N - m + (m - 2k) = N - 2k > \theta \tag{6}$$

where the dimension of all vectors x, the number of noise components of $x(d)$, the number of corresponding bits of $x(d)$ and x_0 which always coincide, and the number of corresponding bits of particular x_r and x_0 which currently differ are N, m, $N - m$, and k, respectively; θ is threshold value of Q or model neuron's triggering threshold.

It is easily to obtain directly that $Q = N - 2D$ and $D = (N - Q)/2$ where D is a Hamming distance between x_0 and $x(d)$ (Hamming distance is a number of corresponding bits of x_0 and $x(d)$ which are different, $0 \le D \le N$). Since between D and Q there is an univocal correspondence, along with inequality $Q > \theta$ the inequality $D < (N - \theta)/2$ is also valid (cf. Inequality 6 were $k = D$). Moreover, $Q(d)$ can merely be interpreted as an expression for computation of Hamming distance D. That means that the convolutional (Hamming) decoding algorithm or

Hamming classifier directly discriminates the patterns $x_{in} = x(d)$ which are more close to x_0 than a given Hamming distance between them [8]. Hence, for data coding described in Section 2 above NN, convolutional, and Hamming distance algorithms are equivalent. In addition, they all are maximum-likelihood ones and optimal in that sense [8].

4. Conditional Recall/Recognition Probabilities and ROCs

The basic idea of NNAMM is to build a NN memory model from simple objects defined within coding/decoding approach introduced. For this purpose in Sections 2 and 3 it is simply enough instead of coding and decoding to say about encoding and retrieval, respectively [8]. In this way NNAMM was formulated and fundamental recall/recognition properties of NNAMM's memory unit containing respective Hopfield NN as its central element were numerically studied by multiple computations [7,8]. But convolutional (Hamming) version of encoding/retrieval algorithm gives a chance to obtain analytical maximum-likelihood formulae for this aim.

Below we derive a formula for the probability $P(m,N,\theta)$ of correct recall/recognition of memory trace x_0 stored in perfectly learned intact NN under condition that data pattern $x(d)$ initiated many-step memory trace retrieval [8] is actually x_0 with damage degree $d = m/N$ (a method for calculation of the same probability by multiple computations and respective examples for $\theta = 0$ see in ref. 7,8). For this purpose it is needed to find the number $T(m,N,\theta)$ of vectors $x(d)$ for which Inequality 6 is valid and the total number of all possible different vectors $x(d)$. Since $x(d)$ contains m randomly permutated components with randomly chosen magnitudes –1 or 1 (the probability of their choice is ½), the latter equals $2^m C^N_m$. To find $T(m,N,\theta)$ we take into account the fact that for each k the number of vectors $x(d)$ satisfying Inequality 6 is $C^N_m C^m_k$ where C^N_m is the number of random permutations of m (from N) noise components of $x(d)$ and C^m_k is the number of random permutations of k (from m) components of $x(d)$ which have other sign than corresponding components of reference vector x_0. Consequently, $T(m,N,\theta) = C^N_m \sum C^m_k$ where the summation is made over $k = 0,1,..,kmax$ (k is Hamming distance between particular $x(d)$ and x_0). The probability $P(m,N,\theta)$ is computed by dividing $T(m,N,\theta)$ by $2^m C^N_m$, i.e.

$$P(m,N,\theta) = \sum_{k=0}^{k\max} C_k^m \,/\, 2^m \tag{7}$$

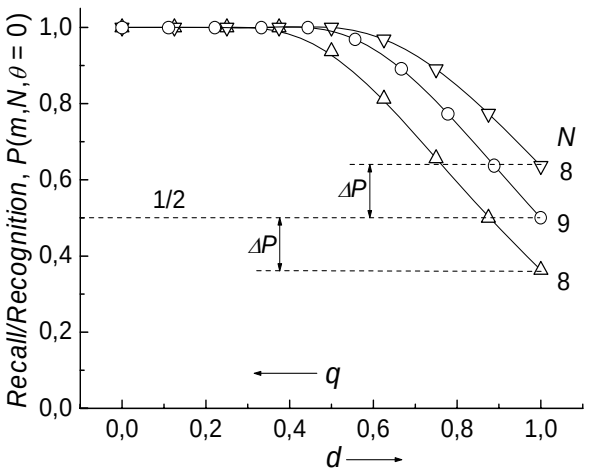
where $kmax$ is defined by Inequality 6 and signum function specified by Equation 5

$$k\max = \begin{cases} (N-\theta-1)/2, & if \quad N \quad is \quad odd \\ (N-\theta)/2-1, & if \quad N \quad is \quad even. \end{cases} \tag{8}$$

Since $0 \le kmax \le m \le N$, if N is odd then $-(N+1) \le \theta \le N-1$ and if N is even then $-(N+2) \le \theta \le N-2$.

Let us consider two important special cases, $P(m,N,\theta) = 1$ and $m = N$, $\theta = 0$:

- Since $\sum C^{kmax}_k = 2^{kmax}$ ($k = 0,1,..,kmax$), from Equation 7 follows that for any N $P(m,N,\theta) = 1$ while $kmax \le m$.
- Since $\sum C^N_k = 2^N$ ($k = 0,1,..,N$), for $m = N$ and $\theta = 0$ if N is odd then $P(m = N,N,\theta = 0) = 2^{N-1}/2^N = ½$. Since additionly $C^m_k = C^m_{m-k}$, if N is even then the sum $S = \sum C^N_k$ ($k = 0,1,..,N/2 - 1$) is defined by equation $2S + C^N_{N/2} = 2^N$ ($C^N_{N/2}$ is the number of events $Q = 0$). Thus, $P(m = N,N,\theta = 0) = ½ - \Delta P(N)$, $\Delta P(N) = C^N_{N/2}/2^{N+1} \sim 0.4/\sqrt{N}$ (here Stirling's formula was used). The facts that $\Delta P(N) < 0$ and the minus sign was assigned to ones in Expression 8 are coursed by the choice of signum function's form. If in Equation 5 for $h_j = \theta$ the value $v_j = +1$ is assigned then $\Delta P(N) > 0$ and the plus sign before ones in expression for $kmax$ should be chosen.



For odd and even N and for different choice of signum function, probabilities $P(m,N,\theta = 0)$ are shown in Figure 1 (as in ref. 7,8 to underline discrete character of NNAMM results small values of N are taken for example).

Figure 1. *Conditional probability P(m,N,θ) of free recall (d = 1), cued recall (0 < d < 1), and recognition (d = 0) calculated according to Equation 7 for triggering threshold θ = 0 and perfectly learned intact NNs with N = 9 (open circles) and N = 8 (triangles) versus damage degree of memory trace x_0 d = m/N or intensity of cue q = 1 – m/N. If N = 9 (N is odd) then free*

recall (false alarm) probability equals ½ (dashed line); if N = 8 (N is even) then free recall probability is less (or more) than ½ on the value of ΔP(8) = 35/256. If N = 9 and m ≤ 4, if N =8 and m ≤ 4 (or m ≤ 5 for other signum function's form) then P(m,N,0) = 1 (see text for details).

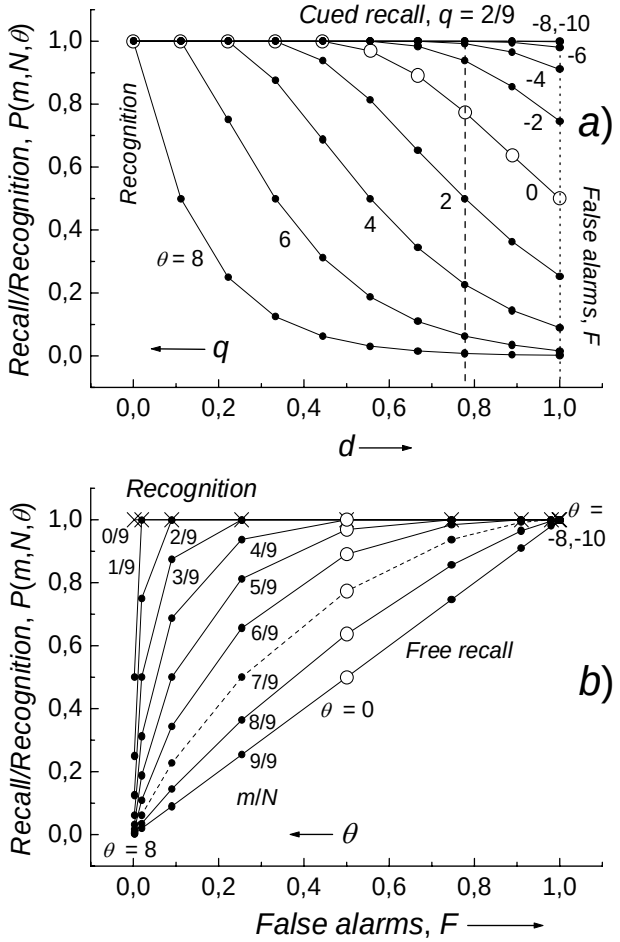


Figure 2 shows the families of curves calculated according to Equation 7; they represent in different form probabilities $P(m,N,\theta)$ for perfectly learned intact NN memory unit with odd N and all possible values of $d = m/N$ and θ.

Figure 2. *Data preparation for ROC plot (a) and ROCs for NN memory unit with N = 9 (b).* ***a)*** *Conditional probabilities P(m,N,θ) versus d = m/N (q = 1 – m/N) and θ. In all Figures open circles denote probabilities P(m,N,0) for usually used value of triggering threshold θ = 0 (i.e., here and in Figure 1 open-circle lines are the same); dashed line connects the values of cued recall (q = 2/9) probabilities for different θ; free recall (q = 0) probabilities are situated along dotted line; for all curves their left-most points are the same, P(0,N,θ) = 1 (that is recognition). The curves demonstrate relations between the thresholds – (N – 1) ≤ θ ≤ N – 1 and false alarm (or free recall) probabilities F = P(N,N,θ); the number of curves is N + 1. Since 0 < F ≤ 1, the value F = 0 is impossible. Right-most point of each curve represents for each θ appropriate value of false alarm F needed to plot ROCs.* ***b)*** *Conditional probabilities P(m,N,θ) versus the false alarm F (or triggering threshold θ) and parameter m/N, i.e. a family of ROC curves. The values of F used for the plot lie along dotted line in panel a). ROC points related to the same value of m/N (q or d) are connected with straight lines; the number of points along each ROC curve is N + 1. The more the value of cue the more the curvature of respective ROC and the more the value of probability P(m,N,θ = 8) for ROC left-most points. Linear ROCs respect to free recall (q = 0, d = 1) and recognition (q = 1, d = 0). Crosses denote recognition probabilities for different θ, P(0,N,θ).*

5. Unconditional Recall/Recognition Probabilities, mROCs, and Overall Probabilities

In Section 4 conditional recall/recognition probabilities were discussed. But it is *a priory* unknown whether initial pattern $x(d)$ is a sample of noise (hypothesis H_0) or memory trace x_0 damaged by noise (hypothesis H_1). To obtain unconditional probabilities of false and correct recall/recognition of the trace x_0 stored in NN memory unit, we use famous Bayes' formular:

$$p_{FR}(m/N,F) = 1/(1+\kappa\frac{P(m/N,F)}{F}), \quad p_{CR}(m/N,F) = 1/(1+\kappa^{-1}\frac{F}{P(m/N,F)}) \tag{9}$$

where $p_{FR}(m/N,F)$ and $p_{CR}(m/N,F)$ reflect unconditional false recall/recognition (FR) and correct recall/recognition (CR) probabilities; $p_{FR} + p_{CR} = 1$; $\kappa = P(H_0)/P(H_1)$; $P(H_0)$ and $P(H_1)$ are prior probabilities of hypotheses H_0 and H_1, respectively. Since $P(H_0)$ and $P(H_1)$ are usually unknown, in most cases it is postulated that $\kappa = 1$.

Here we pay attention to the fact of changing designations: because between F and θ there is a univocal correspondence (see Figure 2a) in Equation 9 and below instead of θ we write F; because all probabilities depend on m and N as on m/N (see Figures 1 and 2) we write these parameters in the latter form.

Data coding approach in ref. 6 introduced allows to find κ in explicit form directly. Indeed, by definition a pattern $x(d)$ contains a fraction $d = m/N$ of noise components and a fraction $q = 1 - m/N$ of undamaged components of x_0 (see Section 2). Hence, d and q may be interpreted as the probabilities $P(H_0)$ and $P(H_1)$ of hypotheses H_0 and H_1, respectively. That means that in Equation 9 within NNAMM approach, κ is given by

$$\kappa = P(H_0)/P(H_1) = d/q = m/(N-m). \tag{10}$$

If $m = N$ then according to Equation 10 κ does not exists and in this special case we posit that $P(H_0) = 1$ and $P(H_1) = 0$, if $m = 0$ then κ^{-1} does not exists and in this special case we posit that $P(H_0) = 0$ and $P(H_1) = 1$. Both propositions are in full concordance with the fact that the former is a case of pure noise and the latter is a case of undamaged memory trace x_0; as it was expected, $P(H_0) + P(H_1) = 1$.

Equations 9,10 provide unconditional probabilities $p_{FR}(m/N,F)$ and $p_{CR}(m/N,F)$ as functions of false alarm F (or model neurons triggering threshold θ) and for this reason for fixed m/N we refer to particular $p_{CR}(m/N,F)$ as modified ROC curve, mROC curve, or mROC. Let us define

$$P_{FR}(m/N) = \sum p_{FR}(m/N,F)/(N+1), \quad P_{CR}(m/N) = \sum p_{CR}(m/N,F)/(N+1) \tag{11}$$

where $P_{FR}(m/N)$ and $P_{CR}(m/N)$ reflect overall, do not depending on F, unconditional FR and CR probabilities of memory trace x_0 stored in perfectly learned NN; summations are made over all $0 < F \le 1$; $p_{FR}(m/N,F)$, $p_{CR}(m/N,F)$ are calculated according to Equation 9; as it was expected, $P_{FR} + P_{CR} = 1$. Due to the use of κ in the form of Equation 10, the values of P_{FR} and P_{CR} may be from the range $0 \le P_{FR} \le 1$ and $0 \le P_{CR} \le 1$ instead of the usual case $0 < P_{FR} \le ½$ and $½ \le P_{CR} < 1$ if it is supposed that $0 < \kappa < \infty$ and $\kappa = 1$. Figure 3 illustrates these claims.

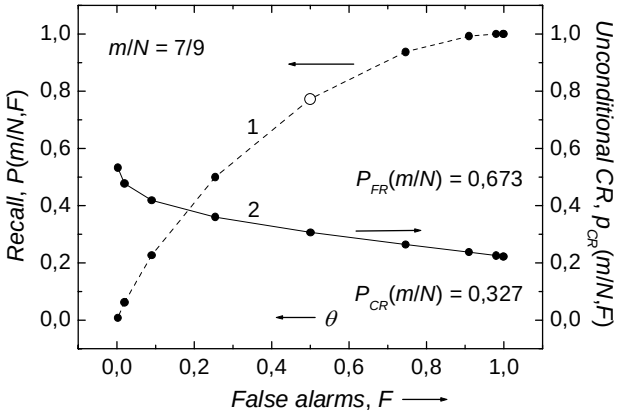


Figure 3. *ROC curve (curve 1, left-hand scale) and mROC curve (curve 2, right-hand scale). ROC points connected with dashed lines are the same as in Figure 2 (d = 7/9, q = 2/9). mROC curve is a plot of unconditional CR probability $p_{CR}(m/N,F)$ as a function of false alarm F. mROC points were calculated according to Equations 9,10; areas above and below curve 2 reflect overall probabilities $P_{FR}(m/N)$ and $P_{CR}(m/N)$, respectively; they are calculated under Equations 11 (that are their simple estimations).*

6. Comparison with Experiments

In Figure 4 NNAMM numerical predictions (calculated ROC curves) are compared with ROC curves observed in item recognition or similar tests. In different panels typical examples of empirical many-point and two-point ROCs are examined, estimated empirical data were adopted from ref. 3. As one can see, good quantitative agreement between theory and experiment is achieved (cue index q is a fit parameter). Thus, the comparison of empirical and model ROCs may be viewed as a method for estimation of specific value of the intensity of cue available for specific memory system in a process of recall or recognition under specific conditions of specific experiment.

As Figure 4 demonstrates, there is no problem to reproduce within NNAMM available empirical ROCs as qualitatively as quantitatively and comparison of calculated and empirical ROCs may be successfully used for the value of intensity of cue, q, estimation. Since for empirical many-point-confidence-scale ROCs the value of cue changes along the curve (see Figure 4a), as the model's predictions as the details of experimental protocols demand scrutiny.

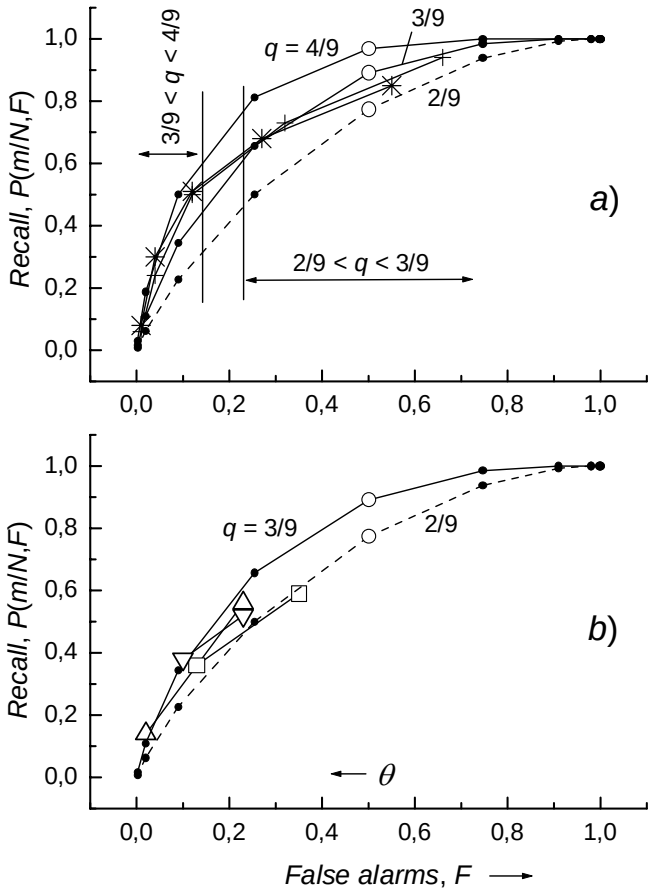


Figure 4. *Theoretical and empirical ROCs. For each calculated ROC, respective value of q is shown. Here and in Figures 2 and 3 the dashed-line curve is the same.* ***a)*** *Examination of observed ROCs derived using 5-point-confidence-scale experiments. Original results are from ref. 11 and 12, the first 3 and last 2 points of empirical ROCs are consistent with the assumption that 3/9 < q < 4/9 and 2/9 < q < 3/9, respectively.* ***b)*** *The same for ROCs derived using 2-point-confidence-scale experiments. Original results are from ref. 13 and 14, they are consistent with 2/9 < q < 3/9.*

Many experiments were also performed where subjects are required to recall as an item itself as other information related to it (e.g., remember/know experiment, associative recognition test, and process-dissociation procedure) [3]. That means that in such experiments those memory events could be selected where subjects are able as target item to retrieve as to assess respective prior probabilities of initial hypotheses on the structure of information initiated retrieval (i.e., whether series of vectors $x(d)$ reflect damaged target item, hypothesis H_1, or lure item, hypothesis H_0). Hence, empirical results obtained using such experimental paradigms could provide unconditional recall probabilities $p_{CR}(m/N,F)$ introduced in Section 5. This assumption is examined in Figure 5.

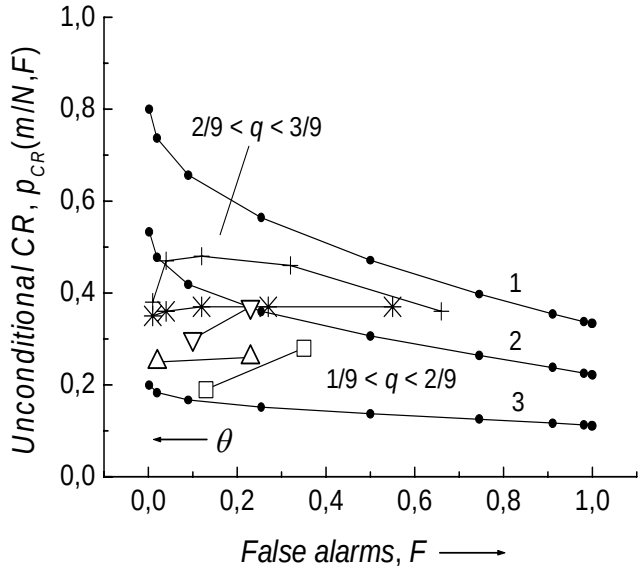


Figure 5. *Theoretical and empirical mROCs. Curves 1,2 and 3 reflect $p_{CR}(m/N,F)$ calculated under Equations 9,10 with cue indices q = 1 – m/N = 3/9, 2/9, and 1/9, respectively. Each observed mROC curve designated in the same signs as ROC curve in Figure 4 was adopted from the same reference [3,11-14].*

As Figure 5 demonstrates, theoretical mROCs provide good quantitative description of observed mROCs [3,11-14] and their comparison may also be viewed as a method for estimation of specific values of the intensity of cue for specific experiments. For example, empirical 2-point mROCs [13,14] are consistence with the assumption that 1/9 < q < 2/9 and for empirical 5-point mROCs [11,12] the value of q changes along the curves from 1/9 < q < 2/9 to 2/9 < q < 3/9.

Comparison between the values of q estimated using respective ROCs (Figure 4) and mROCs (Figure 5) shows that they are similar but not always coincide. Indeed, an analysis of ROCs (Figure 4b) and mROCs observed in experiments [13,14] gives inconsistent results ($2/9 < q < 3/9$ and $1/9 < q < 2/9$, respectively) while the same

analysis of experiments [11,12] gives consistent results if only 2 right-most ROC and mROC points are considered ($2/9 < q < 3/9$, Figures 4a and 5) and inconsistent results if some left-most ROC and mROC points are taken into account ($3/9 < q < 4/9$ and $1/9 < q < 2/9$, respectively). To explain these features, additional analysis as the model's prediction as the details of experiments is needed.

7. Discussion

The properties of empirical ROC curves have been used as one of four basic arguments in favour of DPMs of recognition memory. For example, as Figures 4 and 5 demonstrate empirical ROCs derived from item and associative recognition tests (and from some other similar experiments) are essentially different [3]. ROCs respected to item recognition tests are curvilinear with changing shape across measurement conditions; they can be approximated by a two-factor formula related to signal detection theory and containing recollection and familiarity as stochastically independent fit parameters. For this reason, it is claimed that "at least two separate memory components are needed to account for recognition performance" [3, p.442]. This idea was realized as a two-factor parameterization of empirical ROCs: $P_i = R + (1 - R)\Phi(d'/2 - c_i) + F_i - \Phi(d'/2 - c_i)$ where P_i, F_i, R, d', c_i, and Φ reflect correct recall probability (a counterpart to probability $P(m/N,F)$ defined by Equation 7), false alarm, recollection, familiarity, response criterion, and item distribution (Gaussian), respectively. The fit of this equation to observed ROC curve provides estimations of recollection (R) and familiarity (d'). Since ROCs observed in item recognition tests (Figure 4) are well fitted by this formula and ROCs observed in associative recognition tests (Figure 5) are not, it is suggested that the former can be described by a signal detection theory while the latter can be not [3].

Our NNAMM is based on a version of original binary signal detection theory (Sections 2-5, ref. 7-9] and for intact perfectly learned memory unit is actually a one-factor computer model; this factor is an amount of undamaged information about memory trace x_0 (the value of cue index, q) containing in vectors $x(d)$ initiated many-step memory retrieval [8]. It is essentially that such a one-factor approach on a common ground successfully describes different types of memory including free recall ($q = 0$), cued recall ($0 < q < 1$) , and recognition ($q = 1$) [7,8] and for this reason there is no need to introduce any new type of memory, like recollection or familiarity, for example ("recollection" and "familiarity" of DPMs in our terms are approximately equivalent to recall and recognition, respectively). By definition, all acts of recall or recognition of particular item are different in time processes and, consequently, they are stochastically independent and do not run in parallel. According to NNAMM, recognition ("familiarity") is a one-step process of testing selected memory unit *without* the use of cues stored in other related memory units [8]. Such a process respects in general to item recognition test of so called semantic memory. Recall is a many-step process of testing selected memory unit *with* the use of cues stored in other (one or more) associated memory units [8]. Such a process respects in general to associative recognition test of so called episodic memory (about relations between semantic and episodic memories see ref. 15, for example). Empirical ROCs observed in item recognition tests and mROCs observed in associative recognition tests are successfully described (Section 6) within our NNAMM based on our binary signal detection theory.

Since all basic properties of empirical ROCs (and mROCs) have been qualitatively and even quantitatively reproduced within one-factor NNAMM, they might be excluded from the list of findings underpinning DPMs of recognition memory. On the ground of our recent and previous [16,17] results it is naturally to anticipate that other items of this list (different speeds of response for recollection and familiarity, their different electrophysiological correlates, and different extents of their disruption by certain brain injuries) are also consistent with NNAMM.

8. Conclusion

For the first time a method for theoretical describing of empirical ROC curves is proposed within a computer memory model. For this purpose a convolutional (Hamming) version of our NNAMM based on our original binary signal detection theory [7,8] was used. Analytical formulae for maximum-likelihood calculation of conditional and unconditional probabilities of false/correct recall/recognition of memory trace stored in intact perfectly learned NN memory unit were found. In particular, a method of taking explicitly into account *a priory* probabilities of alternative hypotheses on the structure of information initiated memory retrieval, i.e. vectors $x(d)$, and a method for estimation of overall probabilities are proposed. Using derived analytical formulae empirical ROCs obtained in item recognition tests and empirical mROCs obtained in associative recognition tests were described and the values of intensity of cue for some specific experiments were quantitatively estimated; thus, the comparison of

theoretical and empirical ROCs is a method proposed here to find cue indices for specific experiments. It is shown that ROCs might be excluded from the list of empirical findings underpinning popular DPMs of recognition memory.

I am grateful to HINARI (Health Internetwork Access to Research Initiative) for free on-line access to recent journal full-text articles and my family and my friends for their help and support.

Bibliography

[1] D.Green & J.Swets. Signal Detection Theory and Psychophysics. New York, Wiley, 1966.

[2] W. Metzner, C. Koch, R.Wessel, & F.Gabbiani. Feature Extraction by Burst-Like Spike Pattern in Multiple Sensory Maps. Journal of Neuroscience, 1998, 18(6), 2283-2300.

[3] A.P.Yonalinas. The Nature of Recollection and Familiarity: A Review of 30 Years of Research. Journal of Memory and Language, 2002, 46, 441-517.

[4] B.B.Murdock. Classical Learning Theory and Neural Networks. The Handbook of Brain Theory and Neural Networks. Cambridge, MIT Press, 1995, 189-192.

[5] J.R.Manns, R.O.Hopkins, J.M.Reed, & E.G.Kitchener, L.R.Squire. Recognition Memory and the Human Hippocampus. Neuron, 2003, 37(1), 171-180.

[6] K.A.Norman & R.C.O'Reilly. Modeling Hippocampal and Neocortical Contributions to Recognition Memory: A Complementary Learning Systems Approach. Psychological Review, in the press.

[7] P.M.Gopych. Determination of Memory Performance. JINR Rapid Communications. 1999, 4[96]-99, 61-68 (in Russian).

[8] P.M.Gopych. A Neural Network Assembly Memory Model with Maximum-Likelihood Recall and Recognition Properties. Physics of Particles & Nuclei, in the press. See also Vth International Congress on Mathematical Modeling, Dubna, Russia, September 30 – October 6, 2002, Book of Abstracts, vol.2, p.91.

[9] P.M.Gopych. Identification of Peaks in Line Spectra Using the Algorithm Imitating the Neural Network Operation. Instruments & Experimental Techniques, 1998, 41(3), 341-346.

[10] J.J.Hopfield & D.W.Tank. Computing with Neural Circuits: a Model. Science, 1986, 233, 625-633.

[11] A.P.Yonalinas. Receiver-Operating Characteristics in Recognition Memory: Evidence for a Dual-Process Model. Journal of Experimental Psychology: Learning, Memory, and Cognition, 1994, 20(6), 1341-1354.

[12] A.P.Yonalinas. Consciousness, Control, and Confidence: The 3 Cs of Recognition Memory. Journal of Experimental Psychology: General, 2001, 130(3), 361-379.

[13] F. Strack & J.Foerster. Reporting Recollective Experiences: Direct Access to Memory Systems? Psychological Science, 1995, 6(6), 352-358.

[14] E.Hirshman & A.Henzler. The Role of Decision Processes in Conscious Recollection. Psychological Science, 1998, 9(1), 61-65.

[15] E.Tulving, H.J.Markovitsch. Episodic and Declarative Memory: Role of the Hyppocampus. Hyppocampus, 1998, 8, 198-204.

[16] P.M.Gopych. Three-Stage Quantitative Neural Network Model of the Tip-Of-the-Tongue Phenomenon. Proceedings of the IXth Internetional Conference KDS-2001, St-Petersburg, Russia, June 19-22, 2001, p.158-165 (in Russian). See also http://arXiv.org/abs/cs.CL/0103002, http://arXiv.org/abs/cs.CL/0107012, http://arXiv.org/abs/cs.CL/0204008.

[17] P.M.Gopych. Computer Modeling of Feelings and Emotions: a Quantitative Neural Network Model of the Feeling-Of-Knowing. Kharkiv University Bulletin, Series Psychology, 2002, no.550(1), 54-58 (in Russian). See also http://arXiv.org/abs/cs.AI/0206008.

Author information

Petro M. Gopych – Kharkiv National University; Svoboda Sq., 4, Kharkiv, 61077, Ukraine; e-mail: pmg@kharkov.com.

КОНТЕКСТНО – ЗАВИСИМОЕ РАСПОЗНАВАНИЕ РЕЧИ ДЛЯ ТЕМАТИЧЕСКИХ ТЕКСТОВ

Т. Хашан

Аннотация: *в статье описан статистичесикй метод контекстно – зависимого распознавания тематичесих текстов, позволяющий улучшить качество систем распознавания речи. Основываясь на данных о совместном вхождении конкретных слов в контекстную последовательность (глубина связи – не более 5 слов) с указанным взаимным расположением, строится группа гипотез о появлении в данном контексте слова из списка слов-кандидатов на распознавание и выбирается наиболее вероятная из них. Описана программная реализация данного подхода.*

Ключевые слова*: вероятностный подход, контекстно – зависимое распознавание.*

1. Введение. Обзор контекстно-зависимого распознавания

При построении систем распознавания речи с большим словарем ни одна из существующих на сегодняшний день процедур контекстно-независимого распознавания не удовлетворяет разработчиков в связи с недопустимо низкой вероятностью правильного распознавания, снижающейся с увеличением объема словаря. Отдельную проблему составляет распознавание омонимов и слов, близких по звучанию. Очевидно, что в подобных системах должен присутствовать модуль контекстно-зависимого распознавания, позволяющий корректировать результаты распознавания в соответствии с контекстом. В наиболее общем виде процесс функционирования системы контекстно-зависимого распознавания речи может быть представлен в виде схемы, изображенной на Рис. 1.

Ниже рассматривается статистический подход к реализации контекстно-зависимого распознавания для тематических текстов.

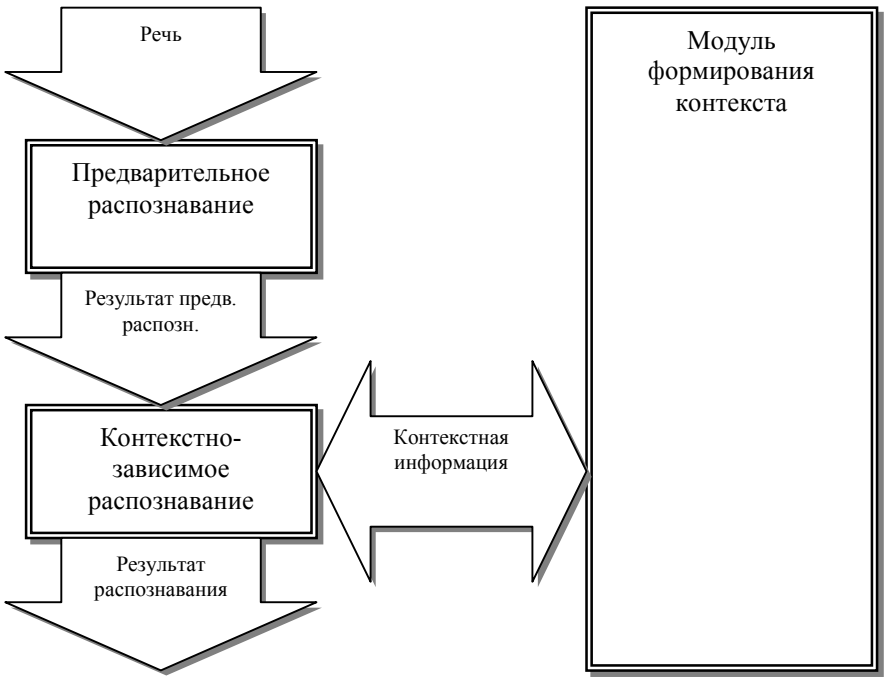


Рисунок 1. Функционирование системы контекстно-зависимого распознавания

2. Контекстно-зависимое распознавание речи для тематических текстов

Наиболее перспективными с точки зрения контекстно-зависимого распознавания являются тексты со сравнительно жестко заданной тематикой (например, математические, медицинские и т.п.). Как показывает практика, подобные тексты содержат значительное количество устойчивых фраз и оборотов, специфичных для выбранной тематики, что значительно повышает эффективность работы модуля контекстно-зависимого распознавания.

2.1. Процедура контекстно – зависимого распознавания

Пусть словарь системы состоит из M слов. Словарь состоит из слов W_i, i=1,...,M, обязательным элементом словаря является пустое слово. В процессе распознавания формируется последовательность $\Re = \{W_{i_1}, W_{i_2}, W_{i_3}, ..., W_{i_k}\}$, эту последовательность мы и будем далее называть *контекстом*; величина K является *объемом контекста*; k-е слово контекста будем обозначать $\Re^k$; номер слова W в словаре будем обозначать N_w.

Распознаванию подлежит очередное произнесенное слово X. Распознавание заключается в последовательной проверке гипотез

$$H_i : X = W_i, i = \overline{1, M}$$

и выборе наиболее вероятной из них.

Для упрощения дальнейшего изложения отвлечемся от конкретного содержания процедуры предварительного распознавания и будем считать, что результатом предварительного распознавания является ряд оценок вероятностей выдвинутых гипотез без учета контекста, т.е.

$$P_i^0 = P(H_i), i = \overline{1, M} \qquad (1)$$

при этом подразумевается, что распознаваемое слово присутствует в словаре и выдвинутые гипотезы образуют полную группу событий, так что

$$P_i^0 \geq 0, \qquad \sum_{i=1}^{M} P_i^0 = 1$$

Задача контекстно–зависимого распознавания заключается в том, чтобы уточнить полученные в результате предварительного распознавания вероятностные оценки, используя информацию о текущем контексте, т.е. построить оценки

$$\widetilde{P}_i = P(H_i \mid \Re)$$

2.2. Контекстная переоценка вероятностей по результатам распознавания

Будем предполагать, что для последовательностей слов из текстов по определенной тематике существуют определенные вероятности совместного вхождения конкретных слов в контекстную последовательность с указанным взаимным расположением, т.е. существуют и определены

$$p_{ijt} = P(\Re^{k-t} = W_j \mid \Re^k = W_i)$$
$$i, j = \overline{1, M}, t = 1, T \qquad (2)$$

где T – глубина контекстных связей.

Тогда, в соответствии с формулой Бейеса (для полной системы гипотез $\{H_i\}_{i=1}^{M}$ в соответствии с формулой Бейеса имеет место соотношение $P(H_i \mid B) = \frac{P(H_i)P(B \mid H_i)}{\sum_{j=1}^{M} P(H_j)P(B \mid H_j)}$), можно использовать

следующую последовательность вычислений для переоценки полученных в результате предварительного распознавания вероятностей по контексту объемом K:

$$P_i^t = \frac{P_i^{t-1} p_{iN_{\Re K-t}t}}{\sum_{l=1}^{M} P_l^{t-1} p_{lN_{\Re K-t}t}}, t = \overline{1,T}$$
$$\widetilde{P}_i = P_i^T \qquad (3)$$

2.3. Построение словаря. Оценка условных вероятностей совместного появления слов

Наиболее целесообразным методом построения словаря и оценки условных вероятностей (2) является автоматический анализ значительного объема существующих текстов по выбранной тематике. В ходе этого анализа выполняются следующие действия:

1. Формируется словарь используемых слов.
2. Определяется количество вхождений каждого слова C_i и количество совместного появления пар слов W_i и W_j на расстоянии t – $C_{ijt.}$
3. В качестве условных вероятностей (2) принимаются их статистические оценки $p_{ijt} = \frac{C_{ijt}}{C_i}$
4. Словам, находящимся от начала фразы на расстоянии меньше используемой глубины, предшествует автоматически включаемое в словарь *пустое слово.*

2.4. Формирование контекста

В ходе функционирования системы контекстно-зависимого распознавания контекст непрерывно изменяется с каждым вновь распознанным словом. Основными рекомендациями по формированию контекста являются следующие:

1. Пользователю системы должна быть предоставлена возможность корректировки контекста в случае неверного распознавания
2. В тех случаях, когда объем контекста меньше глубины контекстных связей, контекст дополняется слева до необходимого объема пустым словом, входящим в состав словаря и имеющим соответствующие условные вероятности
3. При формировании контекста должны учитываться более крупные лингвистические единицы. Контекст должен опустошаться в начале предложений, обособленных предложений в составе сложных предложений, причастных и деепричастных оборотов и т.п.

Реализация указанных рекомендаций увеличивает эффективность контекстно-зависимого распознавания.

Пример программной реализации

Рассмотрим реализацию указанного подхода на примере имитатора контекстно – зависимого распознавания. В указанной программе результаты предварительного распознавания формируются пользователем, что позволяет отвлечься от конкретного содержания процедуры предварительного распознавания и изучить поведение процедуры контекстно-зависимого распознавания в различных ситуациях.

Рассмотрим конкретный пример. Пусть обработке подлежат тексты задач по геометрии и в качестве контекста фигурируют слова «Основанием пирамиды служит …». Будем считать, что после произнесения слова «прямоугольный» и работы процедуры предварительного распознавания, равновероятно распознавание слов «прямоугольный», «прямоугольные», «прямоугольном». В проанализированном тексте встречаются фразы: «Основанием пирамиды служит прямоугольный треугольник…», «Основанием пирамиды служит многоугольник…», «Основанием пирамиды служит параллелограмм…» и т.п.

Тогда после учета контекста и переоценки вероятностей гипотез, последние распределятся следующим образом:

ПРЯМОУГОЛЬНЫЙ	–	93,8%	
ПРЯМОУГОЛЬНИК	–	4,1%	
РАВНОБЕДРЕННЫЙ		–	1%
ОСТАЛЬНЫЕ		–	менее 1%

Из приведенного примера видно, что для правильного распознавания оказалось достаточным, чтобы слово было включено в достаточно большой список равновероятных кандидатов. Таким образом, результаты окончательного распознавания гораздо слабее зависят от качества предварительного распознавания, требуя от него лишь в общем очертить набор похожих слов.

Рабочий кадр, соответствующий описанному примеру, представлен на рисунке 1.

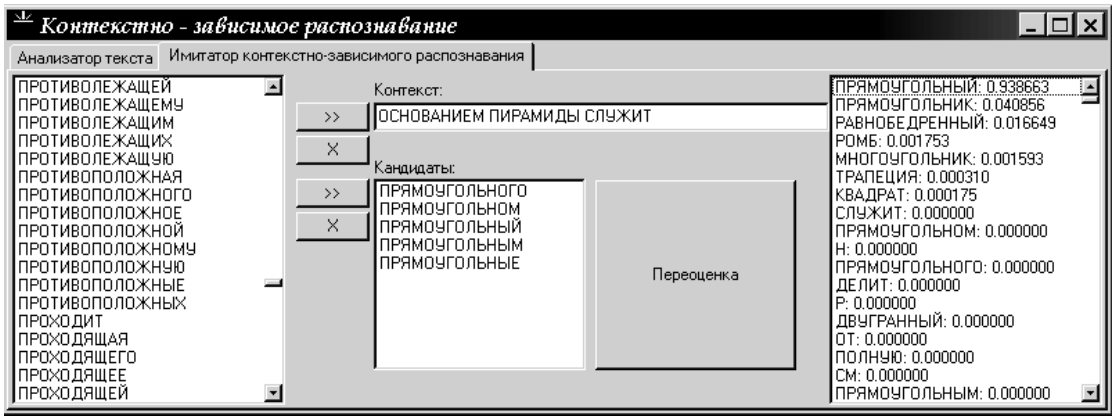


Рисунок 2. Иллюстрация к примеру

Заключение

В работе предложен статистический подход к контекстно-зависимому распознаванию тематических текстов. Его применение позволяет улучшить качество работы систем распознавания речи, предназначенных для работы с тематическими текстами или текстами, содержащими статистически связанные элементы (фразы и обороты). Рассматриваемый подход может быть применен для модификации любой системы распознавания речи, допускающей вероятностное толкование результатов распознавания (распознаватели на основе СММ, нейросетей, сравнения с эталоном и т.д.).

Литература

1. Гмурман В. Е. Теория вероятностей и математическая статистика // Москва, «Высшая школа», 1998г.
2. А.М.Андреев, Д.В.Березкин, А.В.Брик. Лингвистический процессор для информационно-поисковой системы // Компьютерная хроника , 1998. № 11
3. А.М.Андреев, Д.В.Березкин, А.В.Брик, Ю.А.Кантонистов. Вероятностный синтаксический анализатор для информационно – поисковой системы. // Компьютерная хроника , 1998. № 11
4. C.D.Manning, R.Carpenter. Probabilistic Parsing Using Left Corner Language Models. 1997. // Опубликовано на сервере www.xxx.lang.gov/cmp-lg.
5. R.M.Losee. Learning Syntactic Rules and Tags with Genetic Algorithms for Information Retrieval and Filtering: An Empirical Basis for Grammatical Rules. // Information Processing & Management, 1995. // Опубликовано на сервере www.xxx.lang.gov/cmp.lg.
6. D.M.Magerman. Natural Language Parsing as Statistical Pattern Recognition. // A dissertation submitted to the department of computer science at the committee on graduate studies of Stanford University, 1994. // Опубликовано на сервере www.xxx.lang.gov/cmp.lg.

Информация об авторах

Хашан Татьяна – Институт искусственного интеллекта, пр. Б. Хмельницкого, 84, Донецк-83050, Украина; e-mail: tsk@iai.donetsk.ua

ALGORITHM OF CLUSTERIZATION OF MASS-USED MICROROBOTS

I.A.Kaliaev

Abstract: *The problem of microrobots "cloud" clusterization into the groups (clusters), which are intended for solution of target problems set, put before "cloud", is examined. Algorithm of "cloud" clusterization, based on strategy of collective decision making, is suggested. The results of experimental investigations of developed algorithm on program model are shown.*

Keywords*: microrobots "cloud", clusterization, collective decision-making, distributed control system.*

Introduction

Big number of researches /1,2/ is devoted to a problem of creation of the separate microrobot. At the same time, the separate microrobot has as a rule the extremely limited opportunities. Therefore effective functioning of microrobots is possible only in the case of their mass application. As an example of such mass microrobots application may be so called "the smart dust" when "cloud" of the elementary microrobots is used for the solution of some set of target tasks. If for the solution of each target task is necessary some number of microrobots from the "cloud" then a problem of distribution (clusterization) of "cloud" on such groups (clusters) which are capable to solve all tasks put before "cloud" is appeared.

Problem formalization

Formally this problem can be formulated as follows. Let there is the "cloud" containing N of the elementary microrobots R_j $(j = 1,N)$. We shall suppose, that each microrobot R_j of he "cloud" possesses the following opportunities:

- to determine coordinates x_j^T, y_j^T, z_j^T of its current position in the environment;
- to control its own movement to a point with the defined coordinates;
- to exchange by information with all other microrobots of the "cloud".

Suppose, that M targets from set $P = [p_i, i = 1,M]$ are put before the "cloud" of microrobots and each target is defined by coordinates x_i^p, y_i^p, z_i^p of its position in the environment. The target p_i $(i = 1,M)$ can be solved (provided), if it will be achieved by group, containing not less than $n_i < N$ microrobots. We shall consider, that the information about coordinates x_i^p, y_i^p, z_i^p $(i = 1,M)$ of all targets from set P and number of the microrobots necessary for their providing (fig. 1) is available to each microrobot of "cloud".

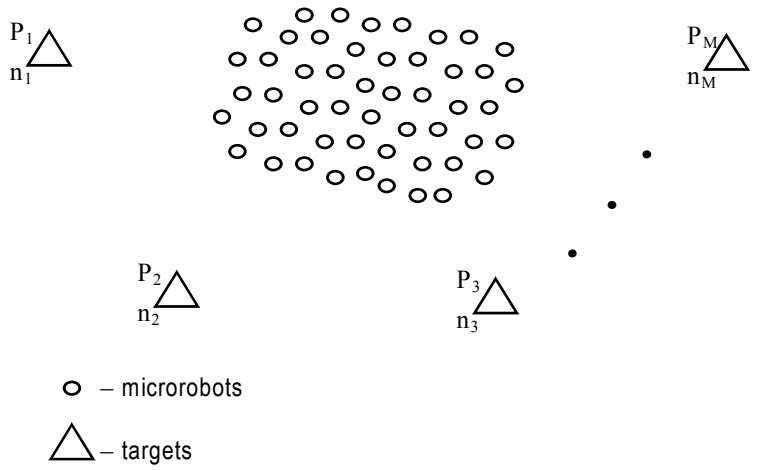


Fig. 1

Algorithm of "cloud" clusterization

It is possible to suggest the following algorithm of microrobots "cloud" clusterization on groups, each of which will provide some target from set *P*. For this purpose it is necessary to organize in memory of each microrobot two one-dimensional arrays C_1 and C_2, the number of elements in which is equal $M+1$, where *M*- number of the targets put before the "cloud". Each element $c_i^1 \in C_1$ $(i=1,M)$ stores the information of number n_i of the microrobots necessary for providing of *i-th* target $p_i \in P$, and the element c_{M+1}^1 is equal to whole number *N* of microrobots in the "cloud". On the other side the element c_i^2 $(i=1,M)$ of the array C_2 stores the information of number of microrobots from the "cloud" which have chosen the target p_i in a present moment of time, and the element c_{M+1}^2 defines number of the "reserve" microrobots which have not chosen any target from set *P*.

The basic idea of suggested algorithm consists in the organization of consecutive procedure of decision-making (target distribution) among microrobots of the "cloud". Before the beginning of this procedure all elements of array C_2 for all microrobots R_j $(j=1,N)$ should be nulled. At first the decision makes the robot R_j with minimal number (all microrobots of "cloud" should be preliminary numbered, for example, from their priorities). As the target it chooses target $p_i \in P$, the distance up to which is minimal, i.e. the quantity

$$L=\sqrt{(x_i^p-x_j^T)^2+(y_i^p-y_j^T)^2+(z_i^p-z_j^T)^2} \quad (1)$$

is minimal,

where x_i^p, y_i^p, z_i^p - coordinates of the target $p_i \in P$;

x_j^T, y_j^T, z_j^T - coordinates of the robot R_j current position (for the first robot *j=1*).

Further, the microrobot R_j realizes check - how much robots have already chosen this target to a present moment of time. For this purpose it reads out from the memory *i-th* element c_i^2 of the array C_2 and compares it with *i-th* element c_i^1 of the array C_1. If

$$c_i^2 < c_i^1, \quad (2)$$

that it means, that number of the microrobots which have chosen *i-th* target, is not enough for its providing. In this case the microrobot R_j chooses this target as its own target. We should notice, as before the beginning of procedure of clusterization all elements of the array C_2 are nulled, for the first robot R_1 the condition (2) will be satisfied for any $i \in 1,M$.

If

$c_2^i \geq c_1^i$,

it means, that *i-th* target is already provided. In this case the microrobot R_j should attempt to choose another target of the set *P*. For this purpose among the remaining part of the targets set P/p_i it finds, according to expression (1), nearest target p_k $(k \in 1,M)$ and then analyzes its providing. If

$c_k^2 < c_k^1$,

it means, that this target is not provided yet. Therefore the microrobot R_j chooses it as the own target. Otherwise it again tries to choose next nearest target among the remaining part of the targets set $P/p_i \cup p_k$.

Process continue until the robot R_j will not choose the target for itself. If all targets from set *P* are already provided then the robot R_j will be included in "reserve" *(M+1)-th* cluster because a condition

$c_{M+1}^2 < c_{M+1}^1 = N$

will be satisfied always.

After finishing of the decision-making procedure robot R_j transmits the information message of its choice to all other robots of "cloud", i.e. messages number *i* of the target chosen by it. On the basis of this information all robots of "cloud" increase value of corresponding *i-th* element of the array C_2 on unit. Besides the robot R_j transmits the information of coordinates of the chosen target to its executive devices for fulfilling of movement in the direction of chosen target position. It is necessary to note, that the coordinates of "reserve" cluster position may be defined in any point of the environment , for example, in middle distance from all targets of the set *P*.

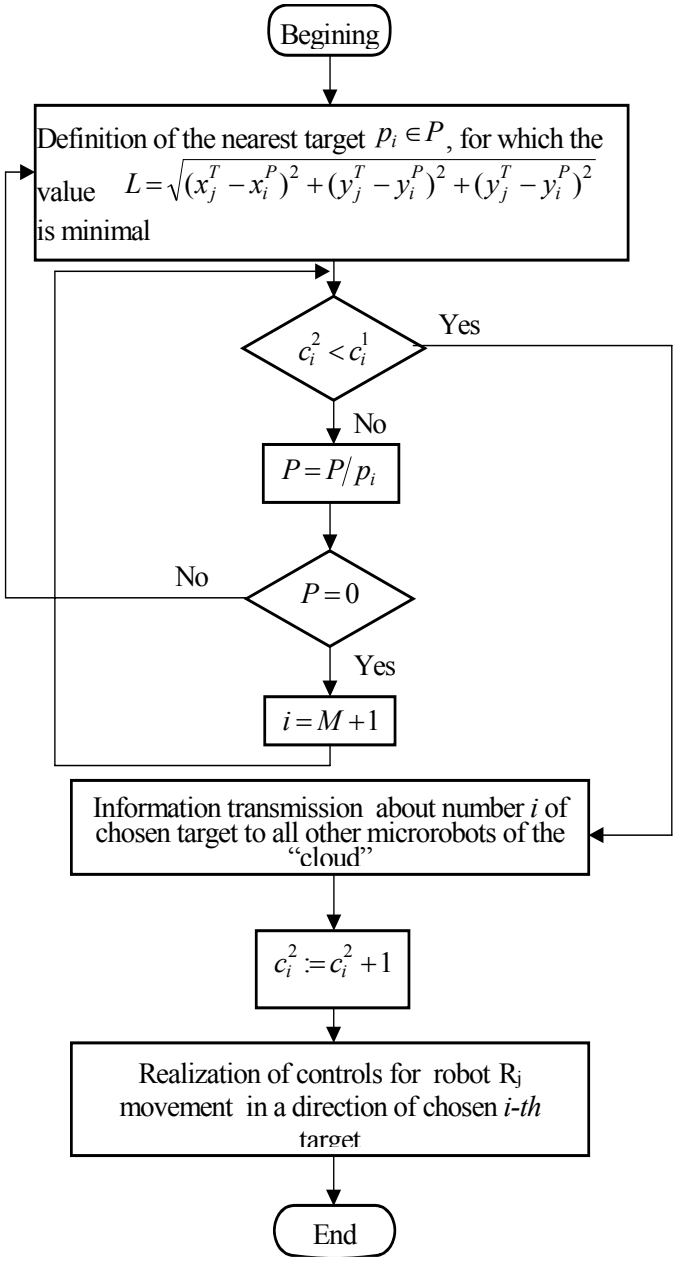


Fig. 2

Structure of microrobot's control system

The graph-scheme of the suggested above decision-making algorithm of *j-th* microrobot of "cloud" is shown on fig. 2 and the structure of a control system of the separate microrobot, corresponding to this algorithm, - on fig. 3.

Here CB is the computing block, realizing algorithm of decision-making, GPS - the receiver of global satellite navigation with which help coordinates of the current position of the robot R_j in environment are determined, BIE - the block of the information exchange serving for communication of the microrobot R_j with other robots of "cloud", ED - the executive devices realizing of the microrobot R_j movement in a direction of the chosen target, "Initial data" - the initial information about targets coordinates and number of the microrobots necessary for their providing, transmitted from some command centre simultaneously to all microrobots of "cloud".

After finishing of decision-making procedure by the microrobot R_j similar procedure is realized by microrobot R_{j+1}, then R_{j+2}, etc. down to the robot R_N. As the result all microrobots of "cloud" will be distributed into $(M+1)$ cluster, each of which will be directed on providing of the corresponding target from set *P* (fig. 4). If

$$\sum_{i=1}^{M} n_i < N,$$

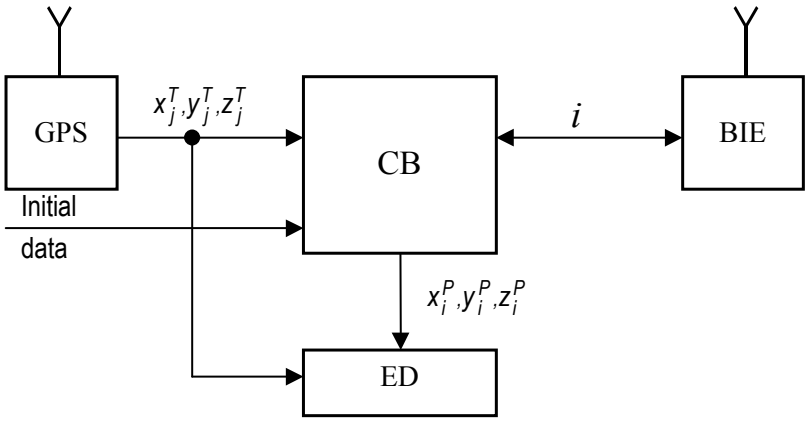


Fig. 3

i.e. number of the microrobots, required for providing all targets from set *P,* is less than the whole number of microrobots in the "cloud", then $(M+1)$-th "reserve" cluster will be formed (fig. 4). If $\sum_{i=1}^{M} n_i \geq N$ then all microrobots of "cloud" will be distributed on the targets from set *P*, and that is why in this case $(M+1)$-th "reserve" cluster will be empty. The whole time *T* of procedure of microrobots "cloud" clusterization will take

$T = N \cdot \tau$,

where *N* - number of microrobots in "cloud";

τ - time given to *j*-th $(j = 1, N)$ microrobot of "cloud" for realization of algorithm of decision-making.

Process of "cloud" clusterization should repeat periodically anew, that will allow "cloud" to adapt operatively to the situation changes in the environment. For example, if some microrobot has broken down (or has been destroyed by the opponent) this fact will be taken into account by "cloud" in the next cycle of clusterization. Really, if some microrobot R_j has broken down, in the time interval given for its decision-making it will not transmit the information about number of the chosen target to other microrobots of "cloud". Hence value of the element c_i^2 of array C_2 (where *i* – number of the target which was chosen by robot R_j in the previous cycle of clusterization) at all microrobots of the "cloud" will decrease on unit. As the result given "vacancy" will be filled by the microrobot from a "reserve" cluster or by the microrobot, which is the located closely to the given target but not including earlier in *i*-th cluster.

Similarly, if coordinates of targets position have changed, it also will be taken into account in the next cycle of clusterization. In this case the list of microrobots, including in the each cluster, can be changed essentially

because the distance *L* between separate microrobots of "cloud" and the targets changes, but the whole number of microrobots in each cluster will satisfy to the requirement of providing of the corresponding targets from set *P* again (certainly, in the case if $\sum_{i=1}^{M} n_i \le N$).

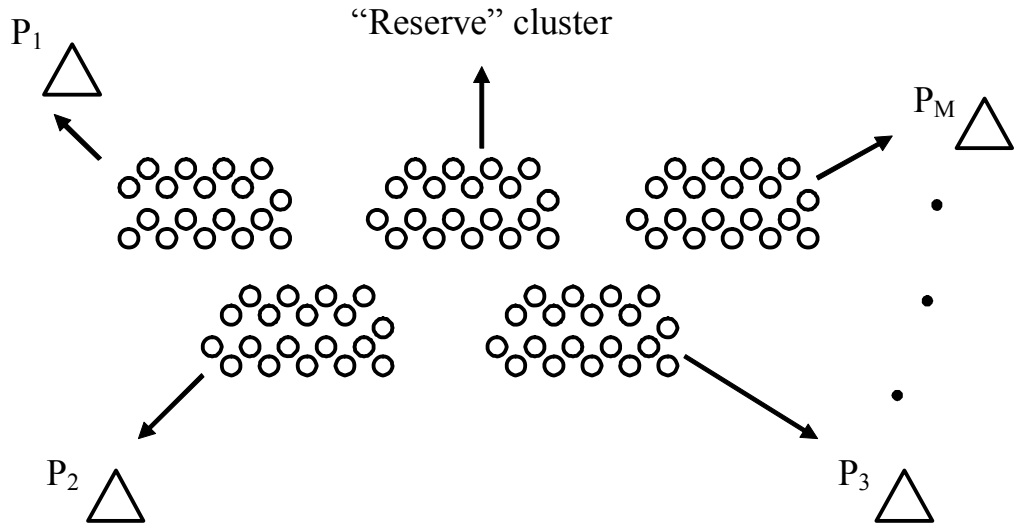


Fig. 4

Experimental investigations

For experimental testing of suggested algorithm of microrobots "cloud" clusterization the special program model has been developed and created. The program model provides the following opportunities:

- arbitrary disposition of the targets from set *P* ;
- dynamic changing of targets positions during "cloud" function;
- destroying some microrobots of "cloud" during its function.

Conclusion

Experiments with program model have shown efficiency of the suggested approach. At first, the algorithm of decision-making executed by each *j*-th $(j = 1, N)$ microrobot of "cloud" is very simple from the computing point of view that provides an opportunity of its realization in real time by means of the elementary microprocessor, placed microrobot board. At second, periodic realization of clusterization procedure allows "cloud" to adapt operatively to the situation changes, such as change of number of microrobots in the "cloud" and change of position of the targets in the environment.

Bibliography

1. L.J. Bocharov, P.P. Maltsev. State and perspectives of development of microelectromechanical systems abroad // Microsystem techniques, № 1, 1999. – pp. 3-6.
2. Rubtsov I.V., Nesterov V.E., Rubtsov V.I. Modern foreign military micro- and mini-robotics // Microsystem techniques, 2000, № 1. – pp. 36-42.

Author information

Igor Kaliaev - Scientific Research Institute of Multiprocessor Computing Systems Taganrog State University of Radioengineering, director, Chekhov Street, 2, Taganrog, 347928, Russia, e-mail: kaliaev@mvs.tsure.ru

METHOD OF COLLECTIVE CONTROL OF THE OBJECTS GROUP

I.A. Kaliaev

Abstract: *The problem of interconnected actions control of objects group, solved common (group) task in conditions of beforehand unknown situation, is examined. The principles of organization and function of group distributed control systems, realized iterative procedure of collective action optimization, are suggested. The results of experimental investigations of suggested approach on program models are shown.*

Keywords: *objects group, group control, iterative procedure, collective decision-making, distributed control systems.*

Introduction

Big number of researches is devoted to a problem of creation of intelligent robots. At the same time, as a result of these researches it became understandable, that the separate robot can be used only for the solution of a narrow circle of a problems. Effective application of robots is possible only at their group interaction when the solution of some complex problem is realized by group of robots [1-3].

Further under the group of robots we shall understand group of some automatic objects which should coordinate their actions for achievement of the common (group) target. The basic complexity here consists in development of methods and algorithms of group control providing coordination of objects actions in the group for optimum solution of the task put before them.

Problem formalization

Formally the problem of group control can be presented as follows.

Let some group of n objects $R_j \ \ (j=\overline{1,n})$ functions in some environment E and states of each object of group are described by a vector-function $\mathbf{S}_j(t)=\langle s_{1j}, s_{2j},..., s_{mj}\rangle$ and states of environment - by vector-function $E(t)=\langle e_1, e_2,..., e_W\rangle$. Besides we shall assume that each object R_j of group can realize some actions described by a vector-function $D_j(t)=\langle d_{1j}, d_{2j},..., d_{Hj}\rangle$ with which help he can change as its own states as a states of environment. Generally these changes are determined by systems of the equations

$$\begin{aligned}\frac{d\mathbf{S}_j}{dt}&=f_j(\mathbf{S}_1,\mathbf{S}_2,...,\mathbf{S}_N,\mathbf{A}_1,\mathbf{A}_2,...,\mathbf{A}_N,\mathbf{E}) \ \ j=\overline{1,n}\\ \frac{d\mathbf{E}}{dt}&=f^*(\mathbf{S}_1,\mathbf{S}_2,...,\mathbf{S}_N,\mathbf{A}_1,\mathbf{A}_2,...,\mathbf{A}_N,\mathbf{E})\end{aligned} \tag{1}$$

Some limitations can be imposed on states S_j of objects $R_j \ \ (j=\overline{1,n})$ and their actions $\mathbf{A}_j$. In common case these limitations are represented by systems of inequalities

$$\alpha(\mathbf{S}_1,\mathbf{S}_2,...,\mathbf{S}_N,\mathbf{E})\le 0 \tag{2}$$

and

$$\beta(\mathbf{A}_1,\mathbf{A}_2,...,\mathbf{A}_N,\mathbf{S}_1,\mathbf{S}_2,...,\mathbf{S}_N,\mathbf{E})\le 0 \tag{3}$$

which should satisfy allowable states of objects of the group and their allowable actions.

Taking into account the designations, entered above, generally the problem of control of objects group consists in definition of such vector-functions of actions $A_j(t)$ for each object $R_i \ \ (j=\overline{1,n})$ of group on an interval of time $[t_0, t_к]$, satisfying systems (1) and limitations (2) and (3) and realizing the extremum of some functional

$$Y = F(A_1, A_2, ..., A_N, S_1, S_2, ..., S_N, E) \quad (4)$$

defining target of objects group functioning in the environment.

If any beforehand unknown forces, capable to change the vector-functions $\mathbf{S}_j(t)$ $(j = \overline{1,n})$ of object states and vector-function $E(t)$ of environment state do not exist in the environment, then the solution of the problem formulated above does not represent basic difficulties. Really, in this case the problem can be solved beforehand (prior to the beginning of actions realization) with the help of classical methods of the control theory, notwithstanding how much time will occupy such solution. Realization of the vector- functions of actions $A_j(t)$ $\left(j = \overline{1,n}\right)$, received as a result of such solution, then can be executed by control systems of the separate objects, included in the group, by a principle of program control. However it is understandable, that any failure which has occurred during realization of beforehand planned actions, will lead to impossibility of achievement of the target put before group of objects.

If objects of group should function in conditions of beforehand unknown changes of a situation in the environment in which beforehand unknown forces operate, i.e. vector- functions $E(t)$ and $\mathbf{S}_j(t)$ $\left(j = \overline{1,n}\right)$ can be changed unpredictable, such approach cannot be used at all. Really, in this case to determine vector functions of actions $A_j(t)$ $\left(j = \overline{1,n}\right)$ on all interval of time $[t_0, t_к]$ has not any sense, because as a result beforehand unforeseen changes of the states of environment $E(t)$ and objects $\mathbf{S}_j(t)$ $\left(j = \overline{1,n}\right)$ these actions may be not optimum from the point of view of an extremum of functional (4), or unrealizable at all according to condition (3). Therefore in this case control by group of objects can be executed by a principle of feedback when in current moment of time t_0 only initial values $\mathbf{A}_j^0 = \mathbf{A}_j(t_0)$ of vector functions $A_j(t)$ $\left(j = \overline{1,n}\right)$ of the actions, directed on optimum achievement of the group target from the current situation, are determined. After realization of these actions procedure of decision-making about new current actions $\mathbf{A}_j^0$ $\left(j = \overline{1,n}\right)$ should repeat anew taking into account new states of objects of group and environment. Process of a choice of the current actions and their realization should repeat periodically down to achievement of the group target, i.e. achievement of an extremum of functional (4). It is necessary to note, that at realization of such approach the problem of definition of the current actions of objects group should be solved during period of time, essentially smaller then time of change of states of objects and environment. Otherwise received actions will not be adequate to the current situation and their realization will not change situation “in the direction” the group target. In other words in this case the solution of the problem of definition of the current actions of objects, including in the group, should be carried out in real time of change of a situation in system “object – environment”.

Centralized organization of group control system

It is possible to suggest two approaches to a problem of group control of the objects functioning in conditions of beforehand unknown situation. The first approach is based on presentation of objects group as uniform object of control. In this case the problem of control by group of objects can be formulated as a problem of optimum control [4].

Problem 1. To find vector-functions of actions $\mathbf{A}_j(t)$ $(j = \overline{1,n})$ of objects in the group on interval of time $[t_0, t_к]$ at which the extremum of target functional (4) is achieved under connections (1) and limitations (2) and (3), where $S_j(t_0) = S_j^0$ $(j = \overline{1,n})$ and $E(t_0) = E^0$, $\mathbf{S}_j^0$ - the current state of object R_j, $\mathbf{E}^0$ - the current state of environment. As the current action A_j^0 of object R_j $(j = \overline{1,n})$ to accept the action determined by initial value of a vector-unction $\mathbf{A}_j(t)$, i.e. $\mathbf{A}_j^0 = \mathbf{A}_j(t_0)$.

Easier speaking, it is necessary to solve in current moment of time t_0 the problem of control by group of objects formulated above (i.e. to find vector functions of actions of all objects of group for achievement of the group target on all interval $[t_0, t_k]$) and after that to accept as the current actions $\mathbf{A}_j^0$ of objects R_j $\left(j = \overline{1,n}\right)$ the initial

values of these functions, i.e. values $A_j(t_0)$. After realizing of these actions process of the group decision-making is necessary to repeat anew taking into account new states of objects and environment and so on till the group target will achieve.

The structure of a control system of objects group, responding such setting of a problem, can be presented as following (fig. 1).

Here each object $R_j \ \left(j = \overline{1,n}\right)$ of group should constantly transmit in central processor unit (CPU) the information about its current state $\mathbf{S}_j^0$ and the current state $\boldsymbol{E}^0$ of environment, surrounding it. On the basis of this information, the central processor unit solves the Problem 1, formulated above, and determines the current actions $A_j^0 = A_j(t_0)$ of objects in group, directed on optimum achievement of the group target from the current situation. These current actions are passed to executive devices of objects for realization. After that the cycle of system work repeats anew in view of the new information about states of objects and environment and so on down to achievement of the group target.

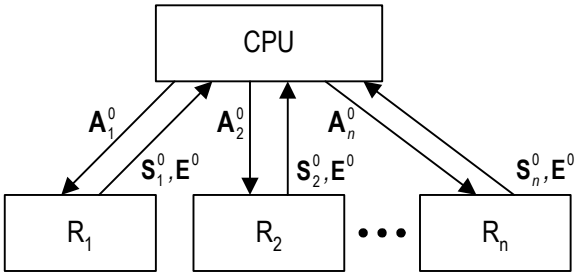


Fig. 1

However such centralized organization of system of group control has a number of essential lacks which complicate its practical use.

First, as the number of objects in group can be great, dimension of the Problem 1, solved by CPU, will be very big. At the same time, as the decision of this problem should be carried out in real time of situation change then CPU should have super high performance.

The second lack is a complexity of the organization of information exchanges between objects and CPU. Really, if objects function on significant distance from stationary CPU then it is necessary to have powerful receiver-transmitter on their board, that can be inconvenient.

The third important lack is a low survivability of system. Really, failure of CPU automatically leads to failure of all groups of objects as a whole.

Iterative procedure of collective decision-making

It is possible to offer other approach to a problem of the group control, devoided the mentioned above lacks. This approach is based on the strategy of control, inherent to collective of people, solving common problem without of the commander. Further, such strategy of control we shall name collective control in contrast to the centralized group control, considered above. Difference of a collective principle of control from group control can be shown evidently by the example of chess and football. In chess one central unit (person) operates by actions of all figures, subordinated to him. Thus he tries to solve a multivariate problem of optimization of actions of all figures for achievement of the group target – checkmate to the opponent – that demands, as a rule, big intellectual and time expenses.

Completely other strategy of group control is used in football. Here each player of a team basically plays in itself, nobody commands him, and he beforehand does not know the further actions of players of the team. At the same time, each player of a football team on the basis of the information about current situation on a field and the

current actions of other players thus chooses the current action to reach the general collective purpose - to score a goal to the opponent and do not miss a ball in the own gate. As the football player does not try to solve a problem of optimization of actions of all other players of the team, and solves only individual problem of optimization of its own actions for achievement of the command target in the current situation then the complexity of solved problem becomes relatively low and that is why its solution can be carried out in real time of change of a situation in the field.

Similar strategy of behaviour can be used and for control of object group, functioning in conditions of beforehand unknown situation. The main principle here consists in following: each object of group should solve individual problem of optimization only its own actions in the current situation, trying in the same time to bring in the maximal contribution in achievement of the common (group) target. Further, to distinguish the group, using the such strategy, we shall name its collective of objects. Formally the problem of control of separate object, included in such collective, can be formulated as follows.

Problem 2. To find a vector- function of actions $\mathbf{A}_j(t)$ of object R_j $(j=\overline{1,n})$ on interval of time $[t_0, t_0+\Delta t]$ which give the extreme increment of target functional (4)

$$\Delta Y = Y(t_0+\Delta t) - Y(t_0)$$

under connections (1), limitations (2) and (3) and fixed values $\mathbf{E}(t)=\mathbf{E}^0$, $\mathbf{S}_i(t)=\mathbf{S}_i^0$ and $\mathbf{A}_i(t)=\mathbf{A}_i^0$ $(i=1,2,...,j-1,j+1,...,n)$, in the right parts (1), (2), (3), where $\mathbf{E}^0$- the current state of environment, $\mathbf{S}_i^0$ and $\mathbf{A}_i^0$ $(i=1,2,...,j-1,j+1,...,n)$ - the current states and actions of other objects of collective at the moment of time t_0. As the current action $\mathbf{A}_j^0$ of object R_j $(j=\overline{1,n})$ to accept the action, determined by initial value of a vector function $\mathbf{A}_j(t)$, i.e. $\mathbf{A}_j^0=\mathbf{A}_j(t_0)$.

In other words, the problem consists in a choice of such current action which would be directed on achievement of the common collective target under fixed current states and actions of other objects of collective, and fixed state of environment. The dimension of such problem will be, at least, in *n* time less in comparison with the Problem 1 of centralized control because for definition of a vector -function $\mathbf{A}_j(t)$ at any $j\in\overline{1,n}$ it is necessary to take into account not n, but only one *j*-th equation from the first system (1), i.e. the number of optimized parameters is reduced in n time, where n - number of objects in collective.

If the interval of time Δt is small enough, it is possible to take, that values of vector functions $\mathbf{S}_j(t)$, $\mathbf{A}_j(t)$ $(j=\overline{1,n})$ and $E(t)$ during this period are constant. Then the Problem 2 can be essentially simplified and presented as follow.

Problem 2*. To find as the current action $\mathbf{A}_j^0$ of object R_j $(j=\overline{1,n})$ such action A_j^{k+1} which satisfies to limitations

$$\beta\left(\mathbf{A}_1^{k+1},...,\mathbf{A}_{j-1}^{k+1},\mathbf{A}_j^{k+1},\mathbf{A}_{j+1}^{k},.....,\mathbf{A}_n^{k},\mathbf{S}_1^0,\mathbf{S}_2^0,....,\mathbf{S}_n^0,\mathbf{E}^0\right)\le 0 \tag{5}$$

$$\alpha\left(\mathbf{S}_{1j}^{k+1},\mathbf{S}_{2j}^{k+1},...,\mathbf{S}_{nj}^{k+1},\mathbf{E}_j^{k+1}\right)\le 0 \tag{6}$$

where $\mathbf{S}_{ij}^{k+1}=\mathbf{S}_i^0+\Delta\mathbf{S}_{ij}^{k+1}$ $(j=\overline{1,n})$ and $\mathbf{E}_j^{k+1}=\mathbf{E}^0+\Delta\mathbf{E}_j^{k+1}$,

$$\begin{aligned}\Delta\mathbf{S}_{ij}^{k+1} &= f_i\left(\mathbf{S}_1^0,...,\mathbf{S}_n^0,\mathbf{A}_1^{k+1},\mathbf{A}_2^{k+1},...,\mathbf{A}_{j-1}^{k+1},\mathbf{A}_j^{k+1},\mathbf{A}_{j+1}^{k},...,\mathbf{A}_n^{k},\mathbf{E}^0\right)\Delta t;\quad i=\overline{1,n}\\ \Delta\mathbf{E}_j^{k+1} &= f^*\left(\mathbf{S}_1^0,...,\mathbf{S}_n^0,\mathbf{A}_1^{k+1},\mathbf{A}_2^{k+1},...,\mathbf{A}_{j-1}^{k+1},\mathbf{A}_j^{k+1},\mathbf{A}_{j+1}^{k},...,\mathbf{A}_n^{k},\mathbf{E}^0\right)\Delta t,\end{aligned} \tag{7}$$

and also gives an extreme increment of target functional (4), i.e. gives extreme value of equation

$$\begin{aligned}&\Delta Y_j^{k+1}=Y_j^{k+1}-Y_{j-1}^{k+1}=\\ &F\left(\mathbf{S}_{1j}^{k+1},...,\mathbf{S}_{nj}^{k+1},\mathbf{A}_1^{k+1},...,\mathbf{A}_{j-1}^{k+1},\mathbf{A}_j^{k+1},\mathbf{A}_{j+1}^{k},...,\mathbf{A}_n^{k},\mathbf{E}_j^{k+1}\right)-\\ &-F\left(\mathbf{S}_{1j-1}^{k},...,\mathbf{S}_{nj-1}^{k},\mathbf{A}_1^{k+1},...,\mathbf{A}_{j-1}^{k+1},\mathbf{A}_j^{k},\mathbf{A}_{j+1}^{k},...,\mathbf{A}_n^{k},\mathbf{E}_{j-1}^{k}\right).\end{aligned} \tag{8}$$

Here $k = 0,1,2,3,...$ – number of an iterative cycle; values $\Delta\mathbf{E}_j^{k+1}$ and $\Delta\mathbf{S}_{ij}^{k+1}$ $(i, j = \overline{1,n})$ define according to (1) changes of the current states of environment and objects of collective on an interval of time Δt as a result of realization of actions $\mathbf{A}_1^{k+1},...,\mathbf{A}_{j-1}^{k+1},\mathbf{A}_j^{k+1},\mathbf{A}_{j+1}^{k},...,\mathbf{A}_n^{k}$ by them; Y_{j-1}^{k+1} - value of target functional, received as a result of realization of actions $\mathbf{A}_1^{k+1},...,\mathbf{A}_{j-1}^{k+1},\mathbf{A}_j^{k},\mathbf{A}_{j+1}^{k},...,\mathbf{A}_n^{k}$ by objects of collective; Y_j^{k+1} - value of target functional, received if object R_j will realize new action $\mathbf{A}_j^{k+1}$; $\mathbf{A}_j^{k}$ $(j = \overline{1,n})$ - the action, chosen by object R_j in *k*-th cycle of iteration; $\mathbf{A}_j^{k+1}$ - the action, chosen by object R_j in *(k+1)*-th iterative cycle; $\mathbf{A}_j^{0}$ $(j = \overline{1,n})$ - initial actions of objects of the collective at the start moment of iterative process.

In other words, the object R_j $(j = \overline{1,n})$ at the moment t_0 should choose as the current action $\mathbf{A}_j^{0}$ such action $\mathbf{A}_j^{k+1}$ allowable in the current situation (condition (5)) which realization does not lead to invalid situation (condition (6)) and gives an extreme increment of target functional (4), i.e. extreme value of equation (8) under condition that other objects of collective realize the actions $\mathbf{A}_1^{k+1},...,\mathbf{A}_{j-1}^{k+1},\mathbf{A}_{j+1}^{k},...,\mathbf{A}_n^{k}$.

Here it is necessary to take into account one important circumstance. According to (5) - (8) action $\mathbf{A}_j^{k+1}$ of each object R_j should be taken into account by all other objects of collective at a choice of there new current actions. Therefore if object R_j choose new current action $\mathbf{A}_j^{k+1}$ by the solution of discrete optimization Problems 2* then all other objects of collective should also try to choose new actions, which are optimum from the point of view of achievement of the group target, because their old choice did not take into account new action of object R_j, i.e. all other object of collective should solve the Problem 2* anew in view of the new action, chosen by object R_j. After that the object R_j should solve the Problem 2* anew, also trying to choose the optimum current action in view of changes in actions of other objects of collective and so on while any changes in actions of objects of collective will not give an essential increment of target functional (4).

The given problem, obviously, is possible to solve by the organization of iterative procedure of optimization of the collective decision-making which essence consists in the following. First the object R_1 on the basis of the information on the current states and actions of other objects of collective solves the formulated above Problem 2* and chooses as its current action such action $\mathbf{A}_j^{k+1}$ (where $k = 0$ - number of an iterative cycle, $j = 1$) which satisfies to limitations (5) and (6) and gives extreme value of equation (8).

The information about the action $\mathbf{A}_1^{k+1}$, chosen by object R_1, is transmitted to all other objects of collective. After that similarly makes decision object R_2, taking into account new action $\mathbf{A}_1^{k+1}$ of object R_1 and old actions $\mathbf{A}_3^{k},...,\mathbf{A}_n^{k}$ of all other objects of collective. In other words, the object R_2 solves the Problem 2* and chooses such action $\mathbf{A}_2^{k+1}$, allowable in the current situation, which satisfies to limitations (5) and (6) and gives an extremum of equation (8). The information about new action $\mathbf{A}_2^{k+1}$ of object R_2 also is transmitted to all objects of collective. Further, object R_3 similarly makes the choice about current action, taking into account new actions, chosen by objects R_1 and R_2 and the previous actions, chosen by all other objects, trying to find such allowable action $\mathbf{A}_3^{k+1}$, satisfying to limitations (5) and (6), which gives extreme value of equation (8). The action $\mathbf{A}_3^{k+1}$, chosen by object R_3, is transmitted to all other objects of collective. Further, the object R_4 makes the decision, then object R_5, and so on till the object R_n. After that the iterative cycle of optimization repeats anew, i.e. the object R_1 again solves the Problem 2*, truing to choose new current action $\mathbf{A}_1^{k+2}$ in view of the actions $\mathbf{A}_j^{k+1}$ $(j = \overline{2,n})$, chosen by other objects in the previous cycle, and so on. The decision will be achieved, if $Y^{k+1} = Y^{k}$, where $Y^{k+1} = Y_n^{k+1}$ и $Y^{k} = Y_n^{k}$ (see Problem 2*), i.e. if values of target functional, received in the

end of (*k*+1)-th and *k-th* cycles of iterative process, will coincide. The actions $\mathbf{A}_j^{k+1}$ $(j=\overline{1,n})$, received as a result of realization of iterative procedure, described above, are accepted as the current actions $\mathbf{A}_j^0$ $(j=\overline{1,n})$ of objects of collective, directed on optimum achievement of the group target in the current situation, i.e. $\mathbf{A}_j^0=\mathbf{A}_j^{k+1}$ $(j=\overline{1,n})$.

Distributed organization of collective control system

The scheme of the organization of the distributed control system by group of the objects, realizing iterative procedure of the collective decision optimization, is shown on Fig.2.

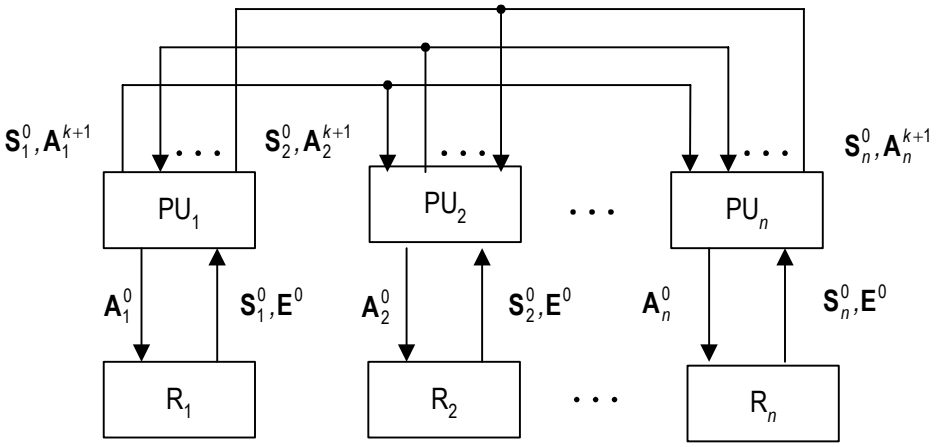


Fig. 2

Here each object R_j $(j=\overline{1,n})$ of collective possesses its own processor unit PU_j, which is connected to processor units of other objects by means of information channels, through which the information about current states $\mathbf{S}_i^0$ of other objects R_i $(i=\overline{1,n},\ i\neq j)$ and actions $\mathbf{A}_i^{k+1}$ $(k=0,1,2,3,\ldots)$, chosen by them during realization of iterative procedure are transmitted. Processor unit of each object can be constructed by following scheme (fig. 3). Here CB – the computing block, BIT – the block of information transmission, BIR – the block of information reception, BDCS – the block of definition of the current state, SB – the sensor block.

As a whole the distributed control system of object collective should work as follows. In the beginning the initial data, necessary for calculation of expressions (5) - (8), are transmitted to processor units PU_j $(j=\overline{1,n})$ of all objects.

Further iterative process of optimization of the collective decision is started. In everyone *(k*+1)-th $(k=0,1,2,\ldots)$ cycle of this process by means of the block of information reception BIR each object R_j $(j=\overline{1,n})$ of collective receive the information about current actions $\mathbf{A}_1^{k+1},\mathbf{A}_2^{k+1},\ldots,\mathbf{A}_{j-1}^{k+1},\mathbf{A}_{j+1}^{k},\ldots,\mathbf{A}_n^{k}$, chosen by all other objects of collective in the current moment of time, and also the information about their current states $\mathbf{S}_1^0,\mathbf{S}_2^0,\ldots,\mathbf{S}_{j-1}^0,\mathbf{S}_{j+1}^0,\ldots,\mathbf{S}_n^0$. This information is transmitted further in computing block CB. Besides the information about current state $\mathbf{S}_j^0$ of the object R_j from BDCS and the current state $\mathbf{E}^0$ of environment from sensor block SB is passed to computing block CB too. On the basis of all this information CB calculates value ΔY_j^{k+1} by means of equations (7) and (8) for every possible allowable actions of object R_j in the current situation (i.e. the actions, satisfying limitations (5) and (6)) and as new action $\mathbf{A}_j^{k+1}$ chooses that action for which value ΔY_j^{k+1} is

extreme. The information about new action $\mathbf{A}_j^{k+1}$, chosen by object R_j, is transmitted to all other objects of collective with the help of the block of information transmission BIT.

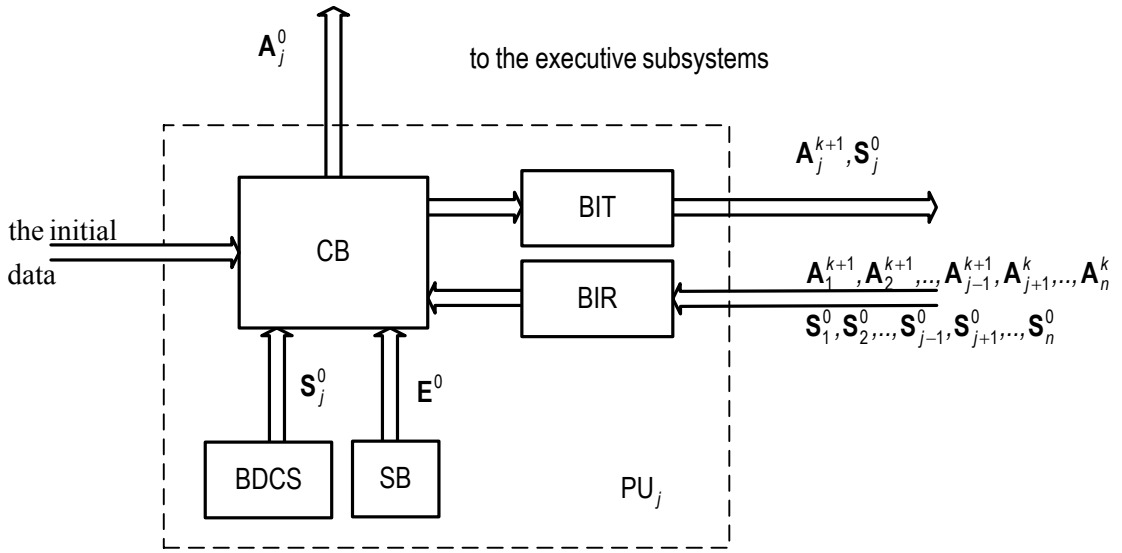


Fig. 3

After that iterative process of optimization proceeds further until value Y^{k+1} of target functional, received in *(k*+1*)*-th a cycle of iteration, will not coincide with the value Y^k, received in *k*-th cycle. The actions $\mathbf{A}_j^{k+1}$ $(j=\overline{1,n})$, received in processor unit PU_j $(j=\overline{1,n})$ in the end of iterative process, are accepted as the current actions $\mathbf{A}_j^0$ of objects of collective and are given out on their executive subsystems for realization.

Further, procedure of optimization of the collective decision repeats anew taking into account new state of environment and objects (for example, the part of objects can be destroyed as a result of counteraction of the opponent), and so on while the collective target will not be reached.

It can be proofed that iterative process, described above, is asymptotically steady in sense of Lyapunov definition under condition that values target functional (4) are limited [5].

The distributed organization of a control system of object collective, realizing iterative procedure of optimization of the collective decision, possesses a lot of advantages in comparison with the centralized organization (see fig. 1), namely:

1. The requirements to performance of processor unit PU_j $(j=\overline{1,n})$ of the distributed system considerably less in comparison with CPU in the centralized system because the dimension of a problem, solved by PU_j, in *n* time is less than dimension of the problem solved by CPU. It in turn allows to realize processor unit PU_j as the compact microprocessor device, placed directly onboard of the object R_j.

2. Requirements to a subsystem of information exchanges are essentially reduced in comparison with the centralized control system. Really, as the volume of the information, transmitted between processor units PU_j $(j=\overline{1,n})$, is insignificant and besides objects of collective should work, as a rule, on small distance from each other then for the organization of information exchanges between objects low-power receiver-transmitter can be used.

3. High survivability of system is provided because failure of separate object or its processor unit does not lead to failure of all collective of objects as a whole. Moreover, at failure of separate object the given circumstance will be taken into account by all other objects of collective in the next iterative cycle of optimization of the collective decision because the failed object will not transmit the information about its current state $\mathbf{S}_j^0$ and the chosen action $\mathbf{A}_j^{k+1}$.

However it is understandable, that all these advantages are reached by refusal of a guarantee of reception of a global extremum of target functional because by using of strategy of collective control only the current actions $\mathbf{A}_j^0$ of objects are determined. At the same time, as we have assumed above, the collective of objects functions in conditions of beforehand unknown situation, in general, it is not meaningful to search the actions $\mathbf{A}_j(t)$ $(j=\overline{1,n})$ of objects on all interval $[t_0, t_к]$ because beforehand unpredictable changes of situation can lead to that these actions will be not optimum or unrealizable.

Experimental investigation

The opportunities of using of the suggested method of collective control by group of objects on model problems such as a problem of distribution of the targets between objects of collective and a problem of control by group of the lifts, serving a high-altitude building, are shown in detail in the paper.

For examination of serviceability and efficiency of the suggested approach to a problem of group control two program models, simulating group interaction of objects, have been developed, namely:

- the program model of the game in virtual football of two teams, each of which aspires to score the maximal number of goals to opponent, trying to pass minimum number of goals in its own gate;
- the program model of military operations between two mixed divisions, each of which aspires to do the maximal losses to the opponent at an allowable level of own losses . Here it is supposed, that each sides has the various types of the combat technique, having certain characteristics, such as fire power, protection, mobility, rate of fire, etc., which are taken into account at optimization of collective actions.

Conclusion

The carried out experimental researches with program models have shown efficiency of the suggested approach to a problem of group control. First, due to simplicity of iterative method of optimization of collective actions time of the decision making is essentially less in comparison with other algorithms. It allows to group, using this approach, to be adapted more operatively to current situation and to accept adequate actions. Secondly, the decision about current actions of objects of the group, received with the help of the suggested method, in most cases is close to optimum. At last, thirdly, comparison of results, received with the help of the suggested method in program models, and by person in identical test situations, shows, that the person solves the same problem not only much more slowly, but also much worse from the point of view of achievement of an extremum of target functional.

Bibliography

1. V.I. Gorodetskii. Multiagent systems: modern state of investigations and perspectives // Artifical intelligence news, № 1, 1996.
2. A.V. Timofeev. Neural Multi-Agent Control of Robotic Systems // Proc. of Inter. Conf. of Informatics and Control, St.-Petersburg, Vol.2, N3, 1997. pp.537-542.
3. V. Stroev. Systems with artifical intelligence used in land forces // Foreign military review, N3, 1997. pp.27-30.
4. Reference book of automatic control theory / Edited by A.A. Krasovisrii, Moscow (Nauka), 1987.
5. I.A. Kaliaev, A.R. Gayduk, S.G. Kapustian. Distributed systems of robots collectives action planning. – Moscow (Yanus-K), 2002. 291 c.

Author information

Igor Kaliaev - Scientific Research Institute of Multiprocessor Computing Systems Taganrog State University of Radioengineering, director, Chekhov Street, 2, Taganrog, 347928, Russia, e-mail: kaliaev@mvs.tsure.ru

APPLICATION OF THE SUFFICIENCY PRINCIPLE IN ACCELERATION OF NEURAL NETWORKS TRAINING

Krissilov V.A., Krissilov A.D., Oleshko D.N.

Abstract: *One of the problems in AI tasks solving by neurocomputing methods is a considerable training time. This problem especially appears when it is needed to reach high quality in forecast reliability or pattern recognition. Some formalised ways for increasing of networks' training speed without loosing of precision are proposed here. The offered approaches are based on the Sufficiency Principle, which is formal representation of the aim of a concrete task and conditions (limitations) of their solving [1]. This is development of the concept that includes the formal aims' description to the context of such AI tasks as classification, pattern recognition, estimation etc.*

Keywords: *neural networks*

Introduction

Nowadays developers have a lot of different models of neural networks and algorithms of their training [2, 3] for disposal. Though the scientific researches are permanently carried on in this field, the theory of neural networks is still feebly formalised. However, even now two stages of creation of artificial neural systems could be defined: structural and parametric synthesis. At the first stage, developer has to do the following: choose the model for the network, define its structure and choose the algorithm for its training. The parametric synthesis includes training processes of the created network and verification of the obtained results. Then, depending on verification results, there can be a necessity of return to one of the stages of structural or parametric synthesis. Thus, becomes obvious that creation of the neural system is an iterative process.

Feeble formalisation of these stages results in necessity for the developer of the neural system to solve a number of problems. E.g., at the structural synthesis stage, in case of solving a non-standard task, it is necessary to spend a lot of time for choosing the corresponding model for the network, choosing its structure and training method. The problem of the parametrical synthesis is a considerable training time. If real tasks are being solved without any simplification, then duration of training process for created network could be too long. However, some tasks require to spend for training as less time as it is possible, e.g., real-time tasks.

The aim of the given article is to offer possible methods to reduce the training time for neural networks with back propagation training algorithm. As such methods are offered: control of procedures of modification and evaluation of weight coefficients, reorganisation of objects in recognition classes. Two possible ways for solving this problem were offered in [4]. The first one was based on choosing the particular functional base for the network. The second method controlled the value of the step of weights modification, considering it from the point of view of a centrifugal force and, adjusting it so that its vector was always directed on an optimum of the set of weights.

In this paper the given problem is considered from the point of view of overtraining the network. In most cases a neural network is trained, while its error will not become equal to zero. It can result in inadmissible spending of time. Though, for most tasks it is *enough* for this error not to exceed some defined value.

Sometimes the level of sufficiency is determined by conditions of the task and required result. However, in most cases this process flows past at an intuitive level and the guided principle is not sufficiently fixed by us. Actually this moment is one of most important in solving similar problems, and optimal value of the varied parameter can depend on many basic values and limitations of the task. Thus, there is a necessity for formalising the given principle, in further – the Sufficiency Principle (SP).

Using SP for training neural networks

Let's consider training of the multilayer back propagation neural network within the frames of solving the classification problem.

Three kinds of errors can be picked out in the training process. Let's name them Elementary Error, Local Error and Global Error. The Elementary Error is the error of a single neuron of the network, for neurons from the output layer it can be evaluated as follows:

$$e_i = Y_i - A_i \tag{1}$$

where Y_i – standard value, A_i – neuron activation level.

The Local Error is the common average error for all neurons of output layer on a single iteration.

$$E_{Li} = \sqrt{\frac{\sum_{k=1}^{m} e_k^2}{m}} \tag{2}$$

where m – number of neurons in the output layer of the network, i – number of training process iteration.

The Global Error is obtained as it is shown below:

$$E = \sqrt{\frac{\sum_{i=1}^{n} E_{L_i}^2}{n}} \tag{3}$$

where n – number of training sets in the training sample.

The neural network is considered to be ideally trained if its Global Error is equal to zero [5]. However, usually it is difficult to train the net to such level and sometimes it is even impossible. These hardships are connected with presence in the training sample of similar training sets. Thus, the more of such sets are in the sample, the harder will be to train the net.

The essence of the SP is the rejection from attempt to reach the Ideal in solving the concrete task. Considering the training process from the point of view of SP and Local Error, it is possible to say, that complete recognition ($E_L = 0$) is not always necessary. Usually, in order to refer some object to a specific class, the Local Error just shouldn't exceed some defined δ.

Thus, in the frames of errors considered above, three kinds of applying the SP are represented below. The first one offers to accept the error of a single neuron equal to zero if its Elementary Error lies within some boundaries ($e_i \le \delta_e$; δ_e – elementary sufficiency parameter). The second considers the Local Error of the neural network. If E_{Li} is less or equal to δ_{EL} (δ_{EL} – local sufficiency parameter), then procedure of recounting the weights won't be applied for this training iteration. And the last one offers to stop the training process after the Global Error of the network will reach value of some δ_E (global sufficiency parameter).

The minimal value of each δ depends on kind of the training sample. Let's consider the following characteristics of the sample: its completeness heterogeneity, and contradictoriness. The completeness is characterised by provision of classes with training sets. The number of training sets for each class should be in 3 – 5 times more, than number of its *features* used in the set [6]. Let's evaluate the value of completeness as follows:

$$F_{TS} = \frac{N_F}{N} * 100 \ \% \tag{4}$$

where N_F – number of classes satisfying to the condition mentioned above; N – number of all classes.

The heterogeneity shows how uniformly the sets are distributed among classes. In order to obtain its value let's take the number of training sets for the i-th class $[C_i]$. Then the mean deviation of this value on sample for the given class is:

$$\overline{\Delta}_{C_i} = \sqrt{\frac{\sum_{k=1}^{N} ([C_i] - [C_k])^2}{N - 1}} \ ; \quad k \neq i \tag{5}$$

Let's evaluate the average of distribution for $\overline{\Delta}_{C_i}$ and $[C_i]$, on condition that values are equiprobable:

$$R_{\Delta} = \frac{\sum_{k=1}^{N} \overline{\Delta}_{C_i}}{N}; \qquad R_C = \frac{\sum_{k=1}^{N} [C_k]}{N} \tag{6}$$

Then heterogeneity can be evaluated as following:

$$H_{TS} = \frac{R_{\Delta}}{R_C} \tag{7}$$

Contradictoriness is a rate of conflicting sets in the training sample. Conflicting sets have the same features, but distributed to different classes. Thus, contradictoriness can be obtained as following:

$$I_{TS} = \frac{N_I}{N} \tag{8}$$

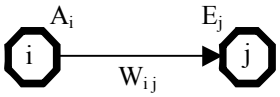
where N_I – number of conflicting sets.

It is obvious, that the lower contradictoriness and heterogeneity, the more narrow can be intervals δ.

Proposed procedures allow to reduce the number of idle changes of weight coefficients. Thus they speed up an approximation of the weights' set to its optimum.

Adjusting the step of weights modification

In original the expression for changing weights between neurons i and j is as following [7]:



Pic. 1

$$W_{ij}^{t+1} = W_{ij}^{t} + \alpha * E_j * A_i^t \tag{9}$$

where E_j – the error of the j-th neuron;

A_i – the activation level of the i-th neuron;

α – the step of weights modification.

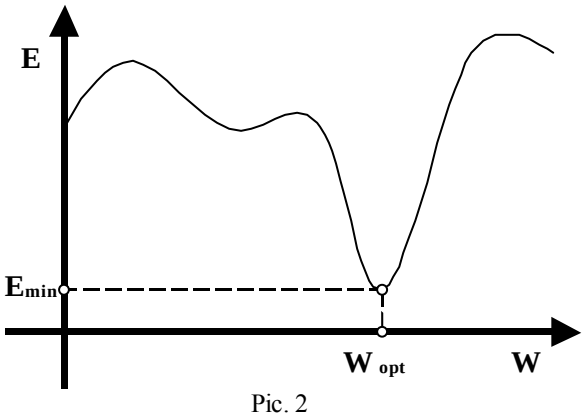
In (9) α is a constant value. However, it is obviously, that if α will be too small, then training will last too long. On the other hand, if α is big, then when the network comes near the minimum point of the error function $E = f(W)$ (E – the Global Error; W – the set of weights) (Pic. 2), it won't be able to reach it. The network will continuously oscillate around this point re-counting its weights and only making worse its characteristics.

Thus it is necessary to manage the value of α. It is obvious, that if W^{opt} should be reached for the minimal number of iterations, then some average value of α is not acceptable.

Then, at the beginning of the training process some maximum value for α should be set. It will provide a quick approximation to the area of W^{opt}. During the approximation the value of α should be gradually decreased.

$$\alpha_0 = \alpha_{\max}; \quad \alpha_{t+1} = \alpha_t - \partial\alpha \tag{10}$$

where $\partial\alpha$ – is decrement of the α.



Pic. 2

The offered method of dynamical adjusting the step of weights modification allows to keep the speed of error's decreasing on a sufficient and satisfactory level.

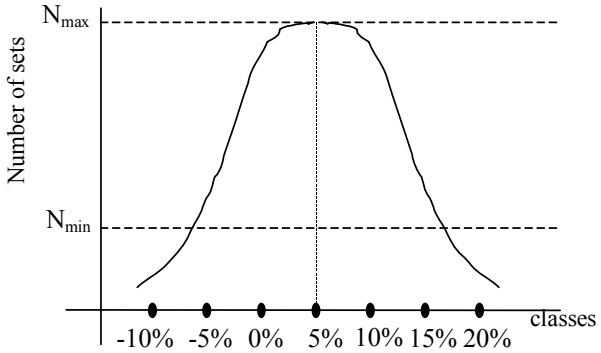
Reorganization of recognition classes.

There are number of AI tasks which suppose of possibility of reorganising objects between classes and classes themselves, e.g. creating the forecast based on analysis of time series. This provides two ways for acceleration of the training process in this case. The aim for these ways is to perfect the training sample's characteristics.

There are number of AI tasks which suppose of possibility of reorganizing objects between classes and classes themselves, e.g. creating the forecast based on analysis of time series. This provides two ways for acceleration of the training process in this case. The aim for these ways is to perfect the training sample's characteristics.

Let's consider reducing the number of recognition classes. It is known that the smaller a neural network is, the quicker will be its training. For back propagation neural network its structure is defined by created training sample: the number of recognition classes uniquely defines the number of neurons in the output layer. Thus, reducing the number of classes results in decreasing the size of the network.

However, there is a big number of real tasks, where such losses in precision of classification are inadmissible. Thus, this method can be applied only for tasks without tight restrictions on precision.



Pic. 3

It is offered to reduce the number of classes by their combining. In order to find classes for combining, it is necessary to analyse the completeness and heterogeneity of the training sample. If the number of training sets for some class doesn't satisfy the completeness condition, or it is greatly less than in other classes, then recognition of this class by the network will be difficult. For example, results obtained after analysis of the training sample can be the classical normal distribution looking as it is shown on Pic. 3. In order to decrease the heterogeneity of the training sample, classes with number of sets lower than some N_{min} should be taken and then neighbouring classes should be combined. Then the number of training sets will get over the barrier of $N_{\min}$ and network will be able to train qualitatively and quickly. However, it will also results in reducing the precision in solving the given task. Thus, it is necessary to adjust, using SP, the number of classes recognised by the network with its size.

Further, let's consider the contradictory training sample. In such a sample classes have both:

objects with low dispersion and located close to the standard of this class – Rules, and objects remote from the standard and located somewhere near the class' boundaries –Eliminations.

Also in the sample there can be classes with high dispersion inside, for which it is impossible to find the standard. Eliminations and Fuzzy classes increase the contradictoriness of the training sample, essentially slow down the training and sometime make it ever impossible. Presence of such elements in the sample can indicates that subsetting of the objects' space on classes was wrong. The solving of this problem is moving Eliminations to other classes and/or forming new classes with lower dispersion.

Thus, the training speed of the network can be increased either by reducing the number of recognition classes, or by moving objects among classes and by forming new classes. The second way increases the training speed by perfecting the training sample, and the first one also by reducing the size of the network.

Conclusions

Thus, three ways of accelerating of the training process for back propagation neural network were considered in this paper.

The first way is based on the analysis of networks' errors. Three levels of errors were described: Elementary Error, Local Error and Global Error. Depending on the kind of the analysed error, different algorithms and software procedures of their implementation were created for obtaining values of the network's weights.

The second way consists in dynamic adjusting the step for changing values of the network's weights. The aim of this method is a minimisation of number of training iterations by reducing the inconsistent adjustments of weights.

The third way considers the reorganization of objects in recognition classes as the way of perfecting characteristics of the training sample: completeness, contradictoriness and heterogeneity.

All proposed ways were applied in forecast and pattern recognition tasks and have brought positive results. They have shown ability to decrease the number of iterations of the training process.

As the test case the task of forecasting the residuals on the bank accounts was solved. The training time was about 30 – 40 hours that was two times less in comparison with original methods.

Applying of them has allowed creating the forecast (for two weeks horizon) with mean-root-square error not greater than 4%.

Bibliography

[1] Krissilov, V.A., Krissilov, A.D. "High-Quality Decision Making by Aim-Oriented Modeling", Proc. of 19-th International Conference of NAFIPS, Atlanta, GA, 2000, pp.241-245

[2] Krissilov, V.A., Oleshko, D.N., Trootnev, A.V. "Applying of neural networks in tasks of intellectual analysis of information", Review of Odessa State Polytechnic University, Vol.2 (8), 1999, pp. 134-139

[3] Dayhoff J. Neural network architectures, New-York: Van Nostrand reinhold, 1991.

[4] Patrick, P. "Minimisation methods for training feed forward Neural Networks", NEUARAL NETWORKS, 1994, Volume 7, Number 1, pp. 1-11

[5] Fausett, L. Fundamentals of Neural Networks. New York: Prentice Hall, 1994.

[6] Bishop C. Neural Networks for Pattern Recognition, Oxford: University Press, 1995.

[7] Patterson D. Artificial Neural Networks, Singapore: Prentice Hall, 1996.

Author information

Victor A. Krissilov – Ph. D., Head of Chair "System Software Design" of Odessa Polytechnic University, App.36., 20 Deribasovskaya Str, Odessa-26, 65026, Ukraine; E-mail: VictorK@OL405.paco.net

Anatoly D. Krissilov – Ph. D., Senior Researcher, Institute for Market Problems and Econo-ecological Research of Ukrainian National Academy of Science 29 Francuzskij Blvrd., Odessa-44, 65044, Ukraine.

Dmitry N. Oleshko – post graduate student of Odessa Polytechnic University; E-mail: boss@ic.ospu.odessa.ua

COMPARATIVE STUDY OF MODULAR CLASSIFIERS AND ITS TRAINING

M. Kussul, A. Galinskaya

***Abstract**: In this paper we present the experimental comparative investigation of modular neural networks intended to solve classification problem in a complex feature space. Several modular architectures are investigated as well as three methods of modular networks training. Simple cooperative training method is introduced. All experiments are carried out with real world data obtained for multi-sensor car safety system. It is*

stated that use of task decomposition based on separation of input space by sensors has advantages over other task decomposition methods for viewed type of problems. Comparison of individual and cooperative training of modular classifiers shows that the methods give similar classification rate. Individual training gives better repeatability whereas cooperative training significantly reduce training time.

***Keywords**: Modular neural network, cooperative training, classifiers*

Introduction

Modular approach is well known in different fields of science and engineering. When saying about neural networks such terms as modular networks and multi-net systems are usually used. We will not make a difference between these two terms and suppose that a system, which consists of more than one module and uses at least one algorithm of artificial neural networks, can be called a modular neural network.

Of late years the usage of modular neural network classifiers for solving complex classification problems increases. The advantages of using modular classifiers over individual neural networks are: possibility to divide a complex task into subtasks and select the most appropriate algorithms to solve the subtasks, reduction of resulted network complexity, training time decrease, possibility to capture discontinuous input-output functions and others. Modular classifiers preferences are discussed in [1].

Modular approach to classification problems is especially effective in the case of complex feature space. Complex feature or input space is often formed by signals from a set of sensors, by use of preprocessing algorithms for the images processing, audio stream in speech recognition and so on. Application of modular classifiers raises three major questions that could be formulated as:

1. which modular architecture is appropriate for the task,
2. what kinds of modules are most effective in the task solution and
3. how to train the modular classifiers.

There are no trivial answers to these questions for particular applied problem solution. Known results in this field did not provide any reliable guidelines to choose directions or at least to introduce limitations so as to indicate the potential area of best solutions. So the investigation should include an experimental validation of the various modular network configurations effectiveness and the ways of their training organization with different data sets.

The main goal of the investigation described in this paper is to analyze and compare training methods for modular neural network classifiers. The classifiers are constructed to solve applied classification problem of car airbag deployment. Some of existing choices of modular neural networks and selected approaches are discussed in section “Modular classifiers”. Section “Statement of the problem” is dedicated to introduce the task, which is solved. Section “Architecture selection and network training” consists of the comparative study of different modular architectures and applied training methods.

Modular classifiers

Modular neural network architectures. Usually four modular neural network architectures or methods of network combining are distinguished. The methods could be called, following Sharkey [2], as:

- *Gathered:* the decision made by the final module is built from a combination of partial solutions suggested by subordinate modules. This approach seems to be most powerful and widely used. Actually, Sharkey has used term “Cooperative” for this method of combining modular components, but we would like to preserve term “cooperative” to call training methods in which modules are trained together in any sense.
- *Competitive:* the gate module or tree of gate modules chooses one of solutions generated by its subordinates [3].
- *Sequential:* the modules are joined in such a way as to form a chain. One of the most interesting chain combinations is a self-organized network using for automatic clustering and then supervised learning network using for the final classification [4].
- *Supervisory:* the final module controls the operation of its subordinates, in particular when those are being trained. Adaptive critic networks can be viewed as an example of the supervisory architectures [5].

It should be noticed that the term “Gathered” here is used in the narrow sense. Stated distinction does not cover modular architectures that can also be named “Gathered” and in which the component modules have an influence on each other during data processing. For example, Associative Projective Neural Networks (APNN) [6]

or modular networks intended to operate with time series and those include feedback or lateral connections between modules.

In this investigation simple gathered architectures of modular networks were used in which subordinate modules accept inputs of a modular network and fusion modules are used to produce output of modular network.

Selecting components for modular networks. Depending on selected modular architecture and current task, modular network components can be determined on the base of task decomposition, successive approximation of solution and functional partitioning of data processing. In addition to selection of subordinate modules, there are numerous approaches to a problem of output module selection or modules for the modular network. Now task decomposition method is wide spread in applied problems and specifically in the field of solution of classification problems defined in complex feature space.

There are two main approaches to classification task decomposition: automatic and explicit. Explicit decomposition relies on well understanding of the problem. The division into subtasks is made prior to training. Explicit decomposition is useful when the task is well defined. Decomposition can be done, for example, on the base of:

- *Subtasks*, where component modules solve own partial problem;
- *Sources*, where subordinates accept data from different sources, for example, sensors;
- *Ensemble or redundant approach*, where component modules solve the same problem in different manner;
- *Feature space partitioning*, as in the combining of local experts;

The last two decompositions are task independent in general. Of course, this distinction is relative. There are no strong boundaries between these approaches and they can be combined for a modular network. Furthermore, any type of neural network can be used as a component network depending on the task.

In the current investigation we have briefly analyzed all of mentioned decomposition methods and decide on subtask and source decompositions.

Training of modular classifiers. As well as diversity of modular architectures there are various methods of training modular neural network classifiers and its components. Beyond the confines of “Gathered” modularity mentioned above, the next attempts to modular network training can be roughly named:

- *Individual training*, where all component modules are trained separately. Each network has its own error function that defines when the network has to be trained for an input vector or a set of vectors.
- *Error propagation*, where output error of a modular network is propagated back and directly determines training (error function) of subordinate networks. Mainly it means that the component networks will be trained in the same time or for the same input vectors.
- *Cooperative*, where decision on training a component module is made by external error function, which of course can depend on error functions of the module and modular network as a whole. This method represents approach between two extremes above, i.e. the modules trained in arbitrary combination for an input vector.

We have examined each of these three approaches of modular neural classifiers training. Neural networks with supervised learning algorithms were used in current investigation. We have chosen one of possible definition of error on subordinate modules for each of these three training methods. Mostly we have used Multi Layer Perceptron (MLP) with error function:

$$\mathbf{E} = \frac{1}{2}\sum_{\mathbf{i}} (\mathbf{O}_{\mathbf{i}} - \mathbf{Y}_{\mathbf{i}})^2$$

where **Y** is actual output of a network, **O** – required output and **i** is index of output neuron.

Individual training supposes using error signal δ defined for a training algorithm of each component net of the modular classifier. For an output node of MLP with back-propagation training algorithm it is:

$$\delta_{\mathbf{i}} = \begin{cases} -(\mathbf{O}_{\mathbf{i}} - \mathbf{Y}_{\mathbf{i}}) \cdot \mathbf{f}' & \textbf{if} \quad \mathbf{E} > \mathbf{T} \\ \mathbf{0} & \textbf{otherwise} \end{cases}$$

where **f’** denotes derivative of neuron transition function and **T** is threshold.

Error propagation can be organized using probability distribution at the component modules as, for example, in [6]. Alternative way is direct propagation of error. To examine error propagation method we have used hidden layer error of back-propagation algorithm where error is distributed from inputs of fusion module to outputs of subordinate modules:

$$\delta_i^S = \begin{cases} f' \sum_j W_{ij} \cdot \delta_j^F & \text{if} \quad E^F > T^F \\ 0 \quad \text{otherwise} \end{cases}$$

Here index **i** denotes the number of output of subordinate module **S**, **j** – index of neuron on the first hidden layer of the fusion network **F** and **W** are weights of connections of the hidden layer.

Cooperative training allows wide scope of external definition of error for subordinate modules. We have used the simplest external rule, which in the case of MLP component networks, can be written as:

$$\delta_i^S = \begin{cases} -(O_i - Y_i) \cdot f' & \text{if} \quad \left((E^F > T^F) \cap (E^S > T^S)\right) \\ 0 \quad \text{otherwise} \end{cases}$$

Here index **F** denotes some fusion network and index **S** denotes it subordinate.

Statement of the problem

It is known that sometimes a car occupant can be injured or even killed by an air bag. There are dangerous positions of an occupant's head in a car cockpit relatively to air bag placement. The cockpit can be divided onto three zones: dangerous zone in which an occupant can be killed, risk zone where an occupant can be injured and safe zone where air bag can safe life of an occupant. In addition, there are some other cases when air bag should not be deployed. Totally, for the task there are defined three conditional classes (deploy, de-powered deploy and not deploy) for seven conditions of the front seat occupation.

Conditional class to deploy an air bag:

- FFA – face forward adult in the safe zone;
- FFS – faced forward child seat;

Conditional class to partially deploy an air bag:

- ARA – adult at risk zone, air bag still has to be deployed but an occupant head position is close to dangerous zone;
- ARS – child seat situated at risk zone.

Conditional class to forbid air bag deployment:

- OPA – adult out of position;
- OPS – child seat out of position;
- RFS – rear faced child seat.

Automatic system has to define occupied zone and depending on conditional class forbid air bag deployment or not.

The classification task is formulated as follow: using signals from several ultrasonic sensors, define in which zone an occupant's head or child seat is located at each moment and make decision to deploy or not deploy an air bag. So, the current task is formulated as two-class problem. Here is used target setting and data collection of Automotive Technology International (NJ, USA).

The data consist of Training set with 281,000 patterns and independently collected set with 97,000 patterns. For cross-validation purpose the independently collected set is split into Test set, which is used to select classifier components (20,000 randomly selected patterns) and Validation set, which is used once to evaluate obtained modular networks to be sure that the classifier generalize well. Each pattern is marked as belonging to one of three mentioned classes. Input feature space is formed with 95 records from four ultrasonic sensors.

Architecture selection and network training. Comparative investigation of modular classifiers was carried out in two stages. On the first stage the modular architectures were compared. The architectures differ in terms of the task decomposition and the type of fusion (output) module. Second stage is comparative study of modular network training methods described above. The investigation was carried out using MNN CAD software [7].

All experimental results in this paper are presented as average classification error for the mentioned classes.

Preliminary tests show that most appropriate neural network to solve the problem is Multi Layer Perceptron (MLP). The task can be naturally decomposed by the conditional classes and by the sensors. In addition, we examined Ensemble organization of the modular classifier where each subordinate network accepts all data. To train individual MLPs two algorithms were used: Extended Delta-Bar-Delta (EDBD) and Generalized Delta Rule (GDR). These and some more of MLP training algorithms can be found in Haykin [8].

Single classifier was selected from the set of networks. The networks were optimized by architecture and the best performance and selected among two and three layers MLP. The best single classifier was used as a baseline for the investigation. Components of modular networks are used without optimization. The best single classifier has such error rate on the examined sets: 3.27% on Training set; 3.21% on the Test and 3.17% on the Validation set. Total average error rate is 3.22%.

Ensemble networks. The investigations of ensemble-organized nets included an analysis of training and testing results obtained with different modular net options as particular parameters were varied:

- **Init** – Variation of random weights in the course of the net organization. The net's modules had the same architecture (95-15-1, the EDBD training algorithm) and differed only in their initialization.
- **Arch** – Variation of the architecture of particular modules. The net consisted of three modules. The modules had such architectures: 95-15-1, 95-21-1 and 95-20-7-1.
- **Alg** – Variation of the training algorithms of particular modules. The net consisted of three uniform modules with the architecture 95-15-1. The training algorithms involved were EDBD, GDR, and Perceptron (the Hebbian rule).
- **Data** – Variation of data used to train particular modules. The net consisted of two modules one of which received original data and the other took data after a random nonlinear transformation using a randomly initialized neural network of the architecture 95-40-95.

For all the items listed above, another object of study was the dependence of the classification results on the way of combining outputs of particular modules in the fusion module. These fusion modules were used:

- **EDBD** - a neuron trained by the EDBD algorithm;
- **Perc** - a perceptron trained by the Hebbian rule;
- **Avg** - a threshold device that calculated the average value of the modules' responses;
- **Max** - a fusion module for selecting a subordinate module by the maximum of the absolute value of the difference between the obtained response and the threshold one. This method also can be treated as selection by maximum of posterior probability of subordinate networks.

These experiments were carried out with number of modules in ensemble varied from 2 to 15 and the best classifiers were selected. Training results are shown in the Table 1, where first column describes method of ensemble network construction, second column gives the number of nets in ensemble and third one shows type of fusion module.

Table 1. Comparison of ensemble networks (% of errors)

Components	Num	Fusion	Train	Test	Valid	Total
Init	3	EDBD	2.84	3.05	3.00	2.96
		Perc.	2.87	3.09	3.03	3.00
		Avg.	2.87	3.1	2.93	2.97
		Max.	2.87	3.16	2.92	2.98
Arch	3	EDBD	2.84	3.06	2.91	2.94
		Perc.	2.96	3.06	2.89	2.97
		Avg.	2.87	3.1	2.93	2.97
		Max.	2.87	3.16	2.92	2.98
Alg.	3	EDBD	3.28	2.91	2.82	3.00
		Perc.	3.1	3.21	3.03	3.11
		Avg.	7.64	4.23	4.09	5.32
		Max.	7.67	4.27	4.13	5.36
Data	2	EDBD	3.40	3.31	3.16	3.29
		Perc.	3.9	3.06	2.91	3.29
		Avg.	4.04	3.35	3.09	3.49
		Max.	4.04	3.35	3.09	3.49

Comparative results of Table 1 suggest use of neural network as fusion module. MLP was used as a fusion module in later experiments according these results.

Task decomposition networks. All classifiers based on task decomposition used MLP with EDBD training algorithm as a fusion module. The experiments were run with modular nets of two types. The nets differed in the task decomposition methods: separation by sensors and subtask decomposition by conditional classes.

Four sets of subordinate networks were investigated for source-based decomposition of the inputs:

- **Net1** - all possible combinations of the four sensors – 15 subordinate modules;
- **Net2** - a net made of modules that contribute most to the decision of the final module (8 nets were selected on the basis of weight factors of the final module);
- **Net3** - a net with data separation by triples of sensors – 4 subordinate modules, each one using inputs from only three sensors;
- **Net4** - a net with data separation by triples of sensors plus module trained at the input data from all four sensors – 5 subordinate modules;

For subtasks decomposition, three sets of subordinate networks were investigated:

- **Net5** - all conditional class separation options – 6 subordinate modules;
- **Net6** - nets that use coupled combinations of the conditional classes – 3 subordinate modules;
- **Net7** - separation "each conditional class versus two others" - 3 subordinate modules;

Experimental results for task decomposition-based classifiers are shown in the Table 2

Table 2. Task decomposition (% of errors)

Separation by sensors				
	Train	**Test**	**Valid**	**Total**
Net1	3.08	2.77	2.83	2.89
Net2	2.98	2.80	2.89	2.89
Net3	3.11	2.91	2.87	2.96
Net4	2.99	2.83	2.83	2.88
Separation by classes				
Net5	3.02	3.00	2.89	2.97
Net6	3.56	3.03	3.00	3.20
Net7	3.66	3.01	3.06	3.24

According to comparison of investigated architectures, the best performance show multi-net system with subordinate networks trained for different sensors inputs.

Comparison of training methods. In this stage of investigation, three training methods for modular classifiers were compared. They are Individual, Error propagation and Cooperative that are defined in section "Modular networks". The methods were compared for two architectures of modular classifiers Net3 and Net4 (see above).

Table 3. Training methods (% of errors)

Training method	**Net**	**Train**	**Test**	**Valid**	**Total**
Individual	Net3	3.11	2.91	2.87	2.96
	Net4	2.99	2.83	2.83	2.88
Error prop.	Net3	3.33	3.04	3.19	3.19
	Net4	3.68	2.92	3.03	3.21
Cooperative	Net3	2.94	2.88	3.03	2.95
	Net4	2.72	3.09	3.12	2.98

Discussion

Main goal of the comparative investigation was to define most appropriate architectures of modular classifiers and training method to solve classification problem defined in the complex feature space. All experiments were done on the data collected for the car safety system where an occupant free moves in a car cockpit. The data consist of patterns with blocked sensors so about 2% of errors could be eliminated by additional modules intended for rejection of classification for such cases or by switching between networks trained without blocked sensors. Thus, difference in 0.1% of classification error is meaningful for current investigation. However, complete safety system is out of examination under this investigation.

Investigation shows that for well-defined problem, such consider here task, stable and best results give classifiers with sensor partitioning of input space. Relatively similar results were obtained with use of ensemble networks, which can be recommended to use if there is no natural background for task decomposition.

Our experiments show that without special optimization and use of task dependent architectures of neural networks, modular classifiers give advantages over single network classifiers from 10% to 30% depending on variety of input features.

Neural network is acceptable and probably preferable choice for a fusion module of gathered or pyramid-like architecture (they call it also cooperative or modular) of modular classifiers.

Individual and Cooperative training give about the same success rate on the viewed task. Used error propagation method, which is emulated by 5-layer not fully connected MLP, shows worse results. During our experiments, Individual training demonstrates more stable results whereas cooperative training significantly reduce training time.

Bibliography

1. M. Sarkar, "*Modular pattern classifiers: a brief survey*", IEEE International Conference on Systems, Man, and Cybernetics, vol. 4, 2000, 2878-2883.
2. A.J C. Sharkey. *Combining Artificial Neural Nets: Ensemble and Modular Multi-Net Systems*. London: Springer-Verlag, 1999.
3. R. A. Jacobs, M.I. Jordan, S.J. Mowlan, and G.E. Hinton. *Adaptive mixtures of local experts*. Neural Computation, 3:79-97, 1991.
4. S.R. Ray and W.H. Hsu Self-organized-expert modular network for classification of spatiotemporal sequences. Intelligent data analysis. 1998
5. Danil V. Prokhorov, Donald C. Wunsch, *Adaptive Critic Designs*, - IEEE Transactions On Neural Networks, Vol. 8, No. 5: 997-1008, September 1997
6. Kussul, E. M., Rachkovskij, D. A., & Baidyk, T. N. (1991a). *Associative-Projective Neural Networks: architecture, implementation, applications*. In Proceedings of the Fourth International Conference "Neural Networks & their Applications", Nimes, France, Nov. 4-8, 1991 (pp. 463-476).
7. M. Kussul, A. Riznyk, E. Sadovaya, A. Sitchov, Tie-Qi Chen. *"A visual solution to modular neural network system development",* Proceedings of the 2002 International Joint Conference on Neural Networks IJCNN'02, Honolulu, HI, USA, 12-17 May 2002, vol.1
8. S. Haykin, Neural Networks: A Comprehensive Foundation, 2nd ed. Englewood Cliffs, NJ, Prentice-Hall, 1999.

Author information

Michael Kussul – PhD., Institute of Mathematical Machines and Systems, NASU, Glushkova ave. 42, 03680, Kiev, Ukraine. kussul@mnn.kiev.ua

Alla Galinskaya – Institute of Mathematical Machines and Systems, NASU, Glushkova ave. 42, 03680, Kiev, Ukraine. alla@mnn.kiev.ua

MULTI-AGENT SECURITY SYSTEM BASED ON NEURAL NETWORK MODEL OF USER'S BEHAVIOR

N. Kussul, A. Shelestov, A. Sidorenko, V. Pasechnik, S. Skakun, Y. Veremeyenko, N. Levchenko

***Abstract**: It is proposed an agent approach for creation of intelligent intrusion detection system. The system allows to detect known type of attacks and anomalies in user activity and computer system behavior. The system includes different types of intelligent agents. The most important one is user agent based on neural network model of user behavior. Proposed approach is verified by experiments in real intranet of Institute of Physics and Technologies of National Technical University of Ukraine "Kiev Polytechnic Institute.*

***Keywords**: neural network, multi-agent system, network security system, user behaviour model, intrusion detection system.*

Introduction

During last decades information technologies based on the computer networks play an important role in various spheres of human activity. Problems of great importance are entrusted on them, such as keeping, transmission and automation of information processing. The security level of processed information can vary from private and commercial to military and state secret. Herewith the violation of the information confidentiality, integrity and accessibility may cause the damage to its owner and have significant undesirable consequences. Thus the problem of information security is concerned. Many organizations and companies develop security facilities that require significant contributions. On the other hand, the impossibility of creating completely protected system is a well-known fact – it will always contain mistakes and «holes» in its realization.

To protect computer systems such accustomed mechanisms as identification and authentication, mechanisms of the delimitation and restriction of the access to information and cryptographic methods are applied. However they possess following drawbacks:

- exposure from internal users with malicious purposes;
- difficulties in access differentiation caused by information resources globalization, which washes away differences between "own" and "foreign" subjects of the system;
- reduction of productivity and communication difficulties due to mechanisms for access control to the resources, for instance, in e-commerce;
- simplicity of passwords definition by making combinations of simple users' associations.

Therefore logging and audit systems are used along with these mechanisms. Among them are Intrusion Detection Systems (IDS).

The Intrusion Detection Systems

IDS are usually devided to systems detecting already known attacks (misuse detection systems) and anomaly detection systems registering the life cycle deviations of the computer system from its normal (typical) activity. Besides, IDS are subdivided to network-based and host-based types by information source. Herewith they can be as real-time (online), so offline.

Network-based IDS analyze network dataflow, protecting its participants, practically not affecting the productivity of their work. Network-based systems do not use information about processes from separate workstation. In turn, the host-based systems are installed on the separate computers and analyze information from their logging mechanisms.

If IDS is real-time an attack can be registered on the stage of its preparation and warned on the stage of its generation (that is more preferable). In this case there is no need to store large amounts of logged data. However the real-time host-based IDS may vastly influence upon the system productivity.

In contrast there are offline systems, which, as a rule, are activated at night or at any other time, when workstation load is low. Thereby, they do not use system resources, necessary for other tasks. Their drawbacks: to analyse information it is necessary to save sufficient amount of audit-data logged during observation, and reaction on attacks is greatly remitted.

At present a lot of IDS are developed. Among them are: Haystack, GrIDS, NIDES, ASAX, DARPA, EPIC2, snort and others. They have made the significant contribution to development of IDS. These systems are based on different algorithms. The main trends are:

- Building activity graphs (Graph-based Intrusion Detection System – GrIDS) in which nodes represent hosts and edges represent network activity among them. The detection technique is to compare graph to a known pattern of intrusive activity.
- Statistical deviation detection methods (Next Generation Intrusion Detection Expert System – NIDES). These systems are the prime examples of anomaly detection systems.
- Employing an expert evaluations. In this approach more scalability is achieved by hierarhical arrangement of the expert systems (Extensible Prototype for Information Command and Control – EPIC2).

Main drawbacks of the described IDS are:

- high probability of the false positive and false negative warnings;
- primitive mechanisms of determining new, unknown in advance intrusions;
- unstable reaction to distributed attacks;
- need of human expertise during all the working time.

To eliminate such defects new approaches were developed. They allow to build completely or highly automated IDS [1]. These approaches are mainly directed on “intellectuallization” of IDS. Among them:

1. Use of neural networks [2,3], genetic algorithms, utilizing variable-sized Markov chains [4] etc.

2. Systems based on agent approach [1,5].

It is known, that beside 70% of attacks are initiated from the inside of network. It might be as password stealing, so using vulnerabilities of information security and the software. So modern approaches actively use the user behavior model.

Developing IDS it is also necessary to take into account distributed nature of attacks on computer network. All these factors show agents approach to be more preferred for creating the security systems.

Agent paradigm

The agents system is meant to be the system of interacting agents. They are coordinated by general global purpose (the strategy) but autonomous enough to realize their own tasks within the framework of the general strategy (the own tactics).

Importance of transition to agent paradigm is compared with importance of using object-oriented approach. Agent technology can be effectively applied in different areas of information technology, e.g. computer networks, software development, object-oriented programming, artificial intelligence, human-machine interaction etc.

Main advantages of intelligent agent systems are as follows.

Distribution. Functional independence of system parts, ability of solving heterogeneous tasks from all domains.

Intelligence. The ability to adapt to the changing environment.

Scalability. Property that makes possible solving new tasks without bringing significant changes to the system architecture.

System structure and functionality

Integrated network IDS should detect different attack types (known and unknown) and anomaly activities. To meet these requirements it should contain various (rather autonomous) interactive modules. Such architecture can be implemented on the base of agent approach. An important role is played by User Agent that should monitor the user behavior and detect anomalies in its activity. Other types of agents are responsible for other aspects of security.

User Agent. This agent allows to detect anomalies in user activity on base of neural network user model. It predicts user actions on the base of the model and compares them to real activity. But we should take into account that behavior of the same user differs for various operating systems. Consequently, User Agent is developed for each type of operating system available in the network (e.g. Win2000, 98, XP, Free BSD).

Host Agent. Performs system calls processing and detects anomalies and known types of attacks. For example, it allows to detect “Trojan horse” attacks.

Network Agent. Operates at the firewall and analyses the network traffic. The information extracted from packets is used to detect known attacks and anomalies in the network. It may be done utilizing neural network and probability approaches (e.g. Bayesian networks and variable-sized Markov chains).

Server Agents. Group of agents are responsible for the server security.

Controller Agent. Responsible for anomalies analysis and detection of distributed attacks in scale of whole system, initializing agents, interaction with database and between various parts of the system.

The structure of proposed system is shown in Fig. 1.

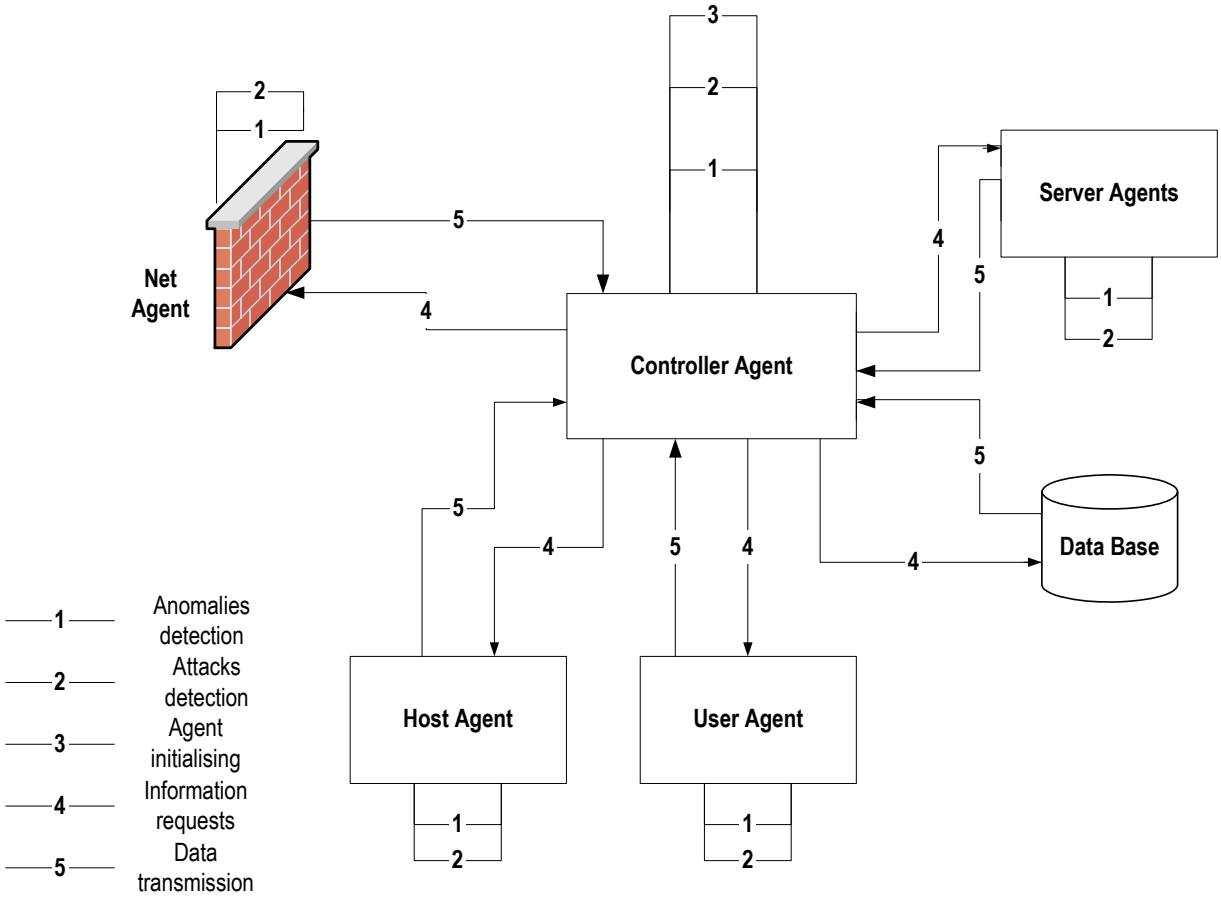


Fig. 1. System structure.

As the user logs on, Controller Agent creates correspondent User Agent and initializes it. During the user session agent controls the user's activity on the base of neural network behavior model. At the same time it picks data for behavior model correction. When the session is finished it sends data for database update. In the case of

anomaly detection User Agent informs Controller Agent about suspicious activity. Host Agents and Server Agents detect system anomalies and known attacks.

Experimental results

Efficiency of suggested approach is confirmed by experimental results. We have built neural network user behavior model for operating system FreeBSD [3]. For this purpose we applied the feed forward neural network that was trained to predict a command by given number of previous ones. The experiments were carried out intranet of Institute of Physics and Technologies of National Technical University of Ukraine "Kiev Polytechnic Institute". About 4000 user behavior models were analyzed. Experimental results confirmed such models to be capable to detect anomalous user activity. Taking into account this experience we propose to spread given approach on other operating systems.

Neural network user behavior model was applied for operating system Windows 98. Initial information for neural network was process sequences run in the system. Neural network was to predict running process by the previous ones. The criterion for the optimal neural network prediction is to distinguish appropriate user from others (Fig. 2).

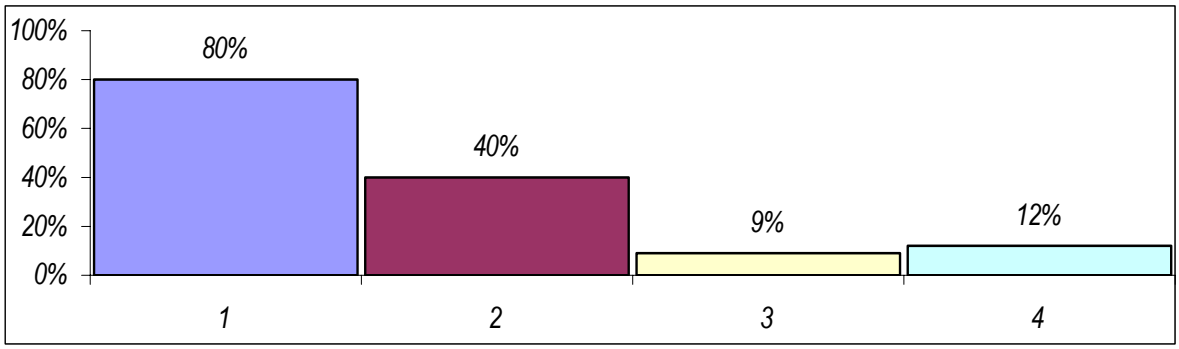


Fig. 2. Indexes of predicted processes for different users.

1 - Index of predicted processes for legal user on training set.

2 - Index of predicted processes for legal user on testing set.

3 - Index of predicted processes for illegal user #1.

4 - Index of predicted processes for illegal user #2.

Prediction errors for process for one session are shown in Fig. 3.

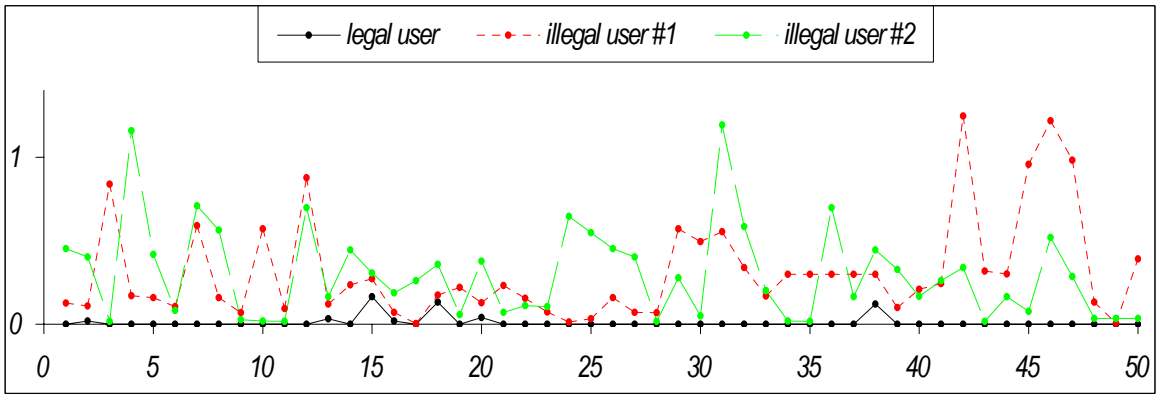


Fig. 3. Prediction errors for processes.

These results show the possibility of neural network to distinguish different users' behavior. Also other experiments were carried out in order to find optimal number of processes (premises) for correct prediction. Best results were achieved by predicting every 6-th command in sequence.

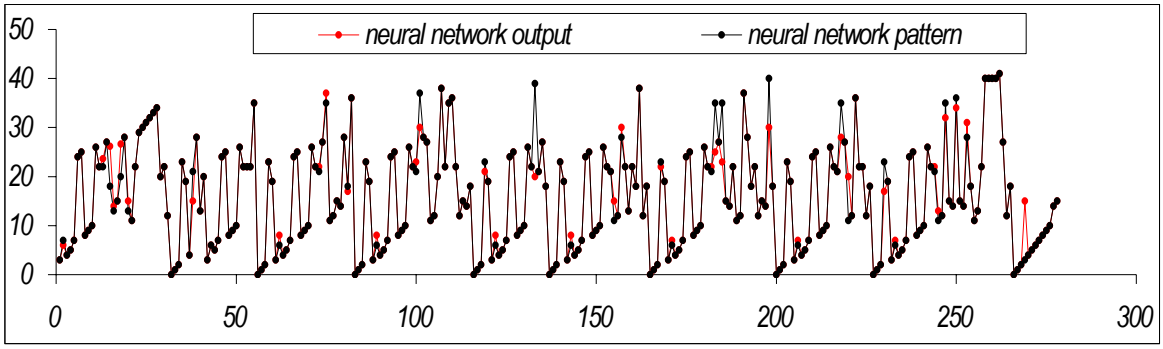


Fig. 3. Predicted process.

Conclusion

The above approach takes advantage of both intellectual methods of intrusion and anomaly detection and multi-agent architecture. The use of neural networks enables detection previously unknown attack types, while agent-based architecture provides features of intelligence and scalability as well as possibility to work in a heterogeneous environment. Currently, research of user behavior model demonstrates effectiveness of such approach.

Bibliography

1. V.Gorodetski, O.Karsaev, A.Khabalov, I.Kotenko, L.Popyack, V.Skormin. Agent-based model of Computer Network Security System: A Case Study. Proceedings of the International Workshop "Mathematical Methods, Models and Architectures for Computer Network Security". Lecture Notes in Computer Science, vol. 2052, Springer Verlag, 2001, pp.39-50.
2. James Cannady, James Mahaffey. The Application of Artificial Neural Networks to Misuse Detection: Initial Results.
3. A.M. Reznik, N.N. Kussul, A.M. Sokolov. Neural network identification of the behavior of the users of computer systems. Cybernetics and computational techniques, 1999, vol.123, pages 70-79.
4. A.M. Sokolov Computer System Intrusion Detection utilizing second-order Markov chain .Proceedings of International Workshop "Artificial Intellegence". vol 1, pages 376-380. (in russian)
5. Jai Sundar Balasubramaniyan, Jose Omar Garcia-Fernandez, David Isacoff, Eugene Spafford, Diego Zamboni. An Architecture for Intrusion Detection using Autonomous Agents http://citeseer.nj.nec.com/balasubramaniyan98architecture.html.

Author information

Natalia Kussul - PhD, Head of Space Information Systems Department, Space Research Institute NASU-NSAU; 40 Glushkov Ave, 03187 Kiev,Ukraine; e-mail: inform@space.is.kiev.ua, nkussul@dialektika.kiev.ua
Andrey Shelestov - PhD, Senior Scientist, Space Research Institute NASU-NSAU; 40 Glushkov Ave, 03187 Kiev,Ukraine; e-mail: inform@space.is.kiev.ua
Anton Sidorenko – Bachelor in Applied Mathematics;, System Developer, Space Research Institute NASU-NSAU; 40 Glushkov Ave, 03187 Kiev,Ukraine; e-mail: inform@space.is.kiev.ua
Vladimir Pasechnik - Bachelor in Applied Mathematics; System Developer, Space Research Institute NASU-NSAU; 40 Glushkov Ave, 03187 Kiev,Ukraine; e-mail: inform@space.is.kiev.ua
Sergey Skakun - Bachelor in Applied Mathematics; System Developer, Space Research Institute NASU-NSAU; 40 Glushkov Ave, 03187 Kiev,Ukraine; e-mail: inform@space.is.kiev.ua
Natalia Levchenko - Bachelor in Applied Mathematics; System Developer, Space Research Institute NASU-NSAU; 40 Glushkov Ave, 03187 Kiev,Ukraine; e-mail: inform@space.is.kiev.ua
Yuri Veremeyenko - Bachelor in Applied Mathematics, Physics and Technology Institute, National Technical University of Ukraine "Kiev Polytechnic Institute"; 37 Peremogy Ave, 03057 Kiev, Ukraine; e-mail: yur@pth.ntu-kpi.kiev.ua

МОДЕЛИ МУЛЬТИ-АГЕНТНОГО ДИАЛОГА И ИНФОРМАЦИОННОГО УПРАВЛЕНИЯ В ГЛОБАЛЬНЫХ ТЕЛЕКОММУНИКАЦИОННЫХ СЕТЯХ

А.В.Тимофеев

Введение

Главной целью управления глобальными телекоммуникационными сетями (ТКС) является быстрый поиск и доставка (транспортировка) необходимой информации пользователям-агентам ТКС с высоким качеством предоставляемых услуг. С этой точки зрения общую задачу управления и мульти-агентной обработки информации в ТКС можно разделить на четыре взаимосвязанных подзадачи:

- управление потоками данных между агентами ТКС с адаптацией к различным видам гетерогенного трафика;
- организация мульти-агентного диалога между агентами ТКС;
- управление телекоммуникационным оборудованием;
- административное управление производительностью и конфигурацией ТКС.

В глобальных ТКС (например, в Internet) управление передачей и распределением (маршрутизацией) потоков данных между агентами должно осуществляться не по жесткой программе, а “в россыпь” по непредсказуемо изменяющимся запросам пользователей или узловых компонентов ТКС.

Традиционный подход к управлению ТКС зачастую не обеспечивает интерактивность в реальном времени (например, речевой диалог без задержек). Другими недостатками традиционного подхода являются неадаптивность (по отношению к изменяющемуся трафику) управления потоками информации, невозможность автоматического предотвращения сетевых конфликтов, распознавания неисправностей и реконфигурирования ТКС без участия человека (администратора или операторов сети).

1. Проблемы управления и мульти-агентного диалога в ТКС

При проектировании и эксплуатации современных ТКС важную роль играют теория и средства управления потоками передаваемых данных. Однако сегодня теория управления ТКС (в отличие, например, от теории автоматического управления движением самолетов, роботов и других подвижных объектов) развита слабо. Поэтому возникает необходимость в постановке, формализации и решении задач управления, обработки и передачи информации в ТКС в режиме диалога в условиях неопределённости.

Специфика ТКС как объекта управления заключается в распределённом характере компонент ТКС и управляемых потоков разнородных (гетерогенных) данных, передаваемых через узлы ТКС по различным маршрутам и каналам связи. Вследствие этого система управления ТКС также должна быть распределённой и иметь многоуровневую иерархическую структуру.

На каждом уровне этой системы возникают специфические цели и задачи управления.

Однако многие из этих целей не формализованы, а задачи не решены (в том смысле, что для этих задач отсутствуют теоретически обоснованные модели “коллективного” диалога и алгоритмы управления). Поэтому в статье значительное внимание уделяется постановке и решению задач организации мульти-агентного диалога и управления потоками данных в условиях неопределённости и сетевых конфликтов.

Неопределённость условий эксплуатации ТКС проявляется в непредсказуемом характере изменения и в гетерогенности сетевого трафика. Неопределённым является также число пользователей-агентов ТКС, которое может значительно изменяться в течение суток. В ТКС могут возникнуть сбои или отказы отдельных компонент, а также разного рода сетевые конфликты. Поэтому необходимы адаптация к трафику, мониторинг и диагностика состояний ТКС, а также классификация и разрешение сетевых конфликтов.

Указанные особенности ТКС требуют исследования и разработки адаптивного подхода к решению задач управления “коллективным” диалогом и потоками данных в ТКС на базе современных интеллектуальных и мульти-агентных технологий. Предлагаемые методы адаптивного и интеллектуального управления обеспечивают адаптацию к непредсказуемо изменяющемуся трафику, адаптивную маршрутизацию потоков данных, мульти-агентную обработку информации, функциональную диагностику ТКС, распознавание и разрешение сетевых конфликтов [3-6].

Широкий класс сложных распределённых ТКС может быть представлен как мульти-агентная система (МАС). При этом в роли агентов ТКС как МАС выступают либо пользователи ТКС, либо узловые компьютеры или локальные ТКС как сегменты ТКС.

Характерными чертами этих агентов ТКС является наличие локальных баз данных и знаний и телекоммуникационных каналов связи для обмена информацией между агентами в процессе совместного использования распределенных информационных ресурсов и обработки передаваемой информации.

Главная особенность мульти-агентной обработки информации и управления заключается в том, что сначала сложная задача декомпозируется (фрагментируется) на ряд локальных задач, решение которых распределяется (распараллеливается) между агентами, а затем результаты решения этих локальных задач агрегируются (интегрируются) и реализуются с помощью телекоммуникационных ресурсов.

Работа МАС обработки информации поддерживается ТКС, реализующей сетевые технологии передачи данных между агентами. Коммуникация между удаленными агентами осуществляется на уровне их локальных баз данных и знаний путём управляемого обмена сообщениями в процессе решения локальных (индивидуальных) или общих (глобальных) задач.

Значительный теоретический и практический интерес представляют две стратегии мульти-агентной обработки и передачи информации:

- с координатором (когда один из агентов отвечает за координацию поведения всех остальных агентов);
- без координатора (когда все агенты “равноправны” и не подчиняются ведущему агенту-координатору).

При мульти-агентном управлении диалогом и потоками данных в ТКС возникает необходимость в разработке методов автоматического предотвращения или разрешения сетевых конфликтов, которые могут возникать между агентами ТКС. В связи с этим большое значение имеют мульти-агентные модели и алгоритмы обработки информации (репликация кода, фрагментация данных, адаптивная маршрутизация и т.п.).

При проектировании систем управления потоками данных в ТКС важную роль играет надёжность используемого оборудования. Надёжность глобальной ТКС тем ниже, чем больше узловых компьютеров входит в состав ТКС. Это объясняется тем, что с увеличением числа узлов ТКС возрастает вероятность выхода из строя одного или нескольких компьютеров. Поэтому возникает необходимость в адаптивном управлении и мульти-агентной обработке информации в ТКС, гарантирующих решение задач при непредсказуемом изменении трафика, сбое или отказе одного или нескольких узловых компьютеров ТКС.

2. Глобальная управляемость и диалоговые возможности ТКС

Основным требованием, предъявляемым к глобальной ТКС, является возможность выполнения ею главной функции - обеспечение реальным и потенциальным пользователям управляемого доступа к распределённым информационным, вычислительным и телекоммуникационным ресурсам всех узловых компьютеров или локальных ТКС, объединённых в глобальную сеть. Для удовлетворения этого требования ТКС как сложный динамический объект информационного управления должна быть глобально управляемой.

В современной теории управления под глобальной управляемостью динамического объекта управления подразумевается существование управления, обеспечивающего перевод этого объекта из любого допустимого начального состояния в любое допустимое конечное (целевое) состояние за конечное время. Если объект управления обладает свойством управляемости, то это означает, что существует один или несколько алгоритмов допустимого управления, гарантирующего достижение цели. В противном случае это вообще невозможно, так как соответствующий алгоритм управления просто не существует.

Применительно к ТКС глобальная управляемость означает возможность доступа с любого узлового компьютера в произвольный момент времени, называемый начальным, к информационным и

телекоммуникационным ресурсам любого другого компьютера ТКС за конечное время. При этом управляемый доступ пользователя к имеющимся распределённым ресурсам обеспечивается автоматическим планированием одного или нескольких коммуникационных маршрутов, связывающих компьютер пользователя с целевым компьютером, который содержит запрашиваемые ресурсы и передаёт их пользователю в виде информационного потока данных.

Однако следует отметить, что в литературе по ТКС (см., например, [1]) под управляемостью сети подразумевается “возможность централизованно контролировать состояние основных элементов сети, выявлять и разрешать проблемы, возникающие при работе сети, выполнять анализ производительности и планировать развитие сети”.

Такое определение управляемости ТКС является слишком общим и расплывчатым. В нём фактически смешаны свойство управляемости ТКС как динамического объекта управления с требованиями к системам управления ТКС, режимам их работы, дефектоустойчивостью ТКС, их производительностью и т.п.

Полезность понятия глобальной управляемости ТКС и необходимость автоматической маршрутизации, управления потоками информации и организации диалога особенно ярко проявляются в сложных корпоративных и глобальных (публичных) сетях. Однако в настоящее время в области управления ТКС существует много нерешённых проблем. Вследствие этого ещё не созданы удобные, многопротокольные и мульти-агентные системы управления ТКС, способные адаптироваться к непредсказуемо изменяющемуся гетерогенному трафику и автоматически разрешать сетевые конфликты.

Многие современные ТКС фактически управляются квалифицированными людьми-администраторами или операторами сетей. Существующие автоматические средства не управляют ТКС, а осуществляют наблюдение (мониторинг) за их работой на основе измерения некоторых важных показателей функционирования ТКС. В действительности они только следят за правильностью функционирования ТКС, формируют и хранят “историю эксплуатации” ТКС и частично контролируют возникающие неисправности.

Существующие системы мониторинга и обработки информации фактически не способны автоматически управлять потоками данных в глобальных ТКС, адаптироваться к изменяющемуся трафику, разрешать сетевые конфликты, диагностировать и устранять неисправности и отказы.

Современные средства автоматизации ТКС обеспечивают управление только отдельными элементами сети, т.е. являются средствами локального управления ТКС. Функции глобального управления, наблюдения и диагностики ТКС возлагается на человека (например, на администратора сети и сетевой персонал).

Важнейшей характеристикой управляемой глобальной ТКС является её производительность. Реальная производительность ТКС существенно зависит от используемой системы управления и может приближаться к определённому пределу, который естественно назвать потенциальной производительностью ТКС.

Производительность глобальных ТКС характеризуется следующими основными показателями [1, 2]: время реакции ТКС, пропускная способность ТКС и задержка передачи потоков данных. Время реакции ТКС - это длительность T интервала времени между начальным моментом t_0 запроса пользователя к ТКС и конечным моментом t_T получения ответа на этот запрос. Значение этого показателя $T=T(\alpha_1, \alpha_2, \ldots)$ зависит от ряда факторов: α_1 - тип службы ТКС, к которой обращается пользователь; α_2 - тип узлового сервера, к которому обращается пользователь; α_3 - текущее состояние элементов ТКС (загруженность сегментов, маршрутизаторов и коммутаторов, через которые проходит запрос, и т.п.); α_4 - время дня, в которое пользователь обращается с запросом в ТК, и т. п.

Время реакции ТКС является глобальной характеристикой производительности ТКС с точки зрения агента-пользователя. Аддитивное разложение времени реакции T на составляющие T_i не интересует пользователя. Ему важен конечный результат - минимальное значение времени реакции глобальной ТКС на его запрос в режиме диалога.

Знание сетевых локальных составляющих T_i времени реакции T позволяет оценить производительность отдельных элементов ТКС, выявить наиболее непроизводительные элементы ТКС и минимизировать

глобальное время реакции T средствами управления или путём модернизации используемого оборудования ТКС.

Таким образом, открывается принципиальная возможность повышения производительности ТКС за счёт оптимизации (по быстродействию) управления. Например, можно минимизировать T_3 за счёт управляемого выбора кратчайших или быстрейших маршрутов “клиент-сервер”, удовлетворяющих информационный запрос пользователя. Для этого можно использовать нейросетевые модели оптимальных маршрутизаторов.

Пропускная способность ТКС V - это объём потока данных, переданных ТКС или её компонентами за единицу времени. Обычно величина V изменяется в битах в секунду или в пакетах в секунду. В отличие от времени реакции ТКС, зависящего, в частности, от пользователя, пропускная способность ТКС V является объективным показателем производительности сети, характеризующим скорость передачи потоков данных между узлами ТКС через различное коммуникационное оборудование. Принято различать среднюю, мгновенную и максимальную пропускную способность ТКС [1].

Глобальная пропускная способность ТКС V зависит от локальных пропускных способностей V_j её компонент, измеряемых, например, между узлами сети или входным и выходным портами маршрутизатора. Вследствие последовательного характера передачи потоков данных различными компонентами ТКС глобальная пропускная способность любого сложного маршрута движения данных в ТКС будет равна минимальной из пропускных способностей элементов этого маршрута. Поэтому для повышения глобальной пропускной способности ТКС необходимо повысить пропускную способность её самых медленных компонент. Это можно сделать, в частности, средствами управления ТКС (например, путём оптимизации маршрутизаторов) по критерию поиска кратчайших или быстрейших коммуникационных маршрутов.

3. Гетерогенность и неопределённость трафика ТКС

Трафик сообщений, передаваемых в телефонных сетях или в сетях кабельного телевидения, значительно отличается от трафика информационных потоков в глобальных ТКС типа Internet. Эти потоки передают не только файлы, БД и т.п., но и мультимедийные данные, представляющие в цифровой форме изображения и речь. Именно поэтому глобальные ТКС всё шире используются для проведения телеконференций, дистанционного обучения и т.п.

Для управляемой передачи мультимедийной информации необходимы не только специальное оборудование, но и новые протоколы и алгоритмы управления, обеспечивающие адаптацию к изменяющемуся мультимедийному трафику. Дело в том, что звуковые колебания и световые волны являются непрерывными процессами. Поэтому для их высококачественного воспроизведения необходимо их измерить, закодировать и синхронно передать так, чтобы не было значительных искажений и запаздываний.

Традиционный трафик данных, передаваемых ТКС, может непредсказуемо изменяться в широких пределах. По существу этот трафик имеет “пульсирующий” характер и заранее неизвестную интенсивность. Например, запрос пользователя в БД удалённого компьютера порождает поток данных между его локальным и удалённым компьютерами, зависящий от многих факторов (редактирование текстов и т.п.), причём его незначительная задержка практически не влияет на качество обслуживания пользователя ТКС.

Однако при изменении трафика в широких пределах качество обслуживания может значительно ухудшиться. Поэтому возникает необходимость в адаптации системы управления ТКС к неизвестному трафику, изменяющемуся в широких пределах непредсказуемым образом.

Другая причина возникновения необходимости в адаптивном управлении связана с тем, что в глобальных ТКС должны передаваться как обычные потоки данных, так и мультимедийная информация. Это означает, что реальный трафик глобальных ТКС обычно является неопределенным и гетерогенным. Поэтому системы управления потоками информации в глобальных ТКС должны адаптироваться к заранее неизвестным особенностям гетерогенного трафика и обеспечивать высокое качество обслуживания пользователей, состав которых также может непредсказуемо изменяться.

4. Децентрализированное управление диалогом типа " клиент – сервер"

Среди компьютерных сетей хронологически первыми появились глобальные ТКС, объединявшие между собой компьютеры, распределенные по разным городам и странам, и обеспечивающие возможность управляемого обмена информацией на большом расстоянии.

Именно при проектировании глобальных ТКС были предложены и отработаны основные идеи, механизмы и технология работы современных компьютерных сетей. К ним относятся принципы многоуровнего построения телекоммуникационных протоколов, технология маршрутизации и коммуникации информационных пакетов и т.п. [1,2]. Эти принципы и технологии нашли воплощение в наиболее популярной ТКС – Internet и продолжают совершенствоваться с учетом новых потребностей информационного общества.

Глобальные ТКС относятся к распределенным компьютерным сетям с децентрализованным управлением локальными центрами (узлами) обработки и передачи информации на базе компьютеров с сетевыми адаптерами, связанных между собой коммуникационными каналами. Для локального управления компьютерами в режиме запроса его ресурсов необходимы программные модули – серверы, постоянно ожидающие и оперативно обслуживающие запросы пользователей глобальной ТКС на доступ к ресурсам.

С другой стороны, для локального управления компьютерами в режиме формирования запросов к ресурсам удаленных компьютеров ТКС необходимы программные модули – клиенты (client), вырабатывающие и передающие запросы в адрес нужного компьютера ТКС.

Пары модулей «клиент - сервер» является важной частью сетевой операционной системы. Они обеспечивают управляемый совместный доступ к определенному типу удаленных информационных ресурсов (например, к файлам).

Множество таких пар (по всем типам ресурсов) образует сетевую службу, управляющую обменом информацией между компьютером – сервером, представляющим свои ресурсы другим компьютерам ТКС, и компьютером – клиентом, потребляющим эти ресурсы. Одни и те же компьютеры ТКС могут одновременно играть роль и сервера, и клиента.

Система управления глобальной ТКС состоит из сетевой операционной системы, системы управления локальными и распределенными ресурсами, называемой сетевой службой, и системы управления информационными потоками, связанной с так называемыми сетевыми приложениями (сетевые БД и БЗ, системы архивирования данных, почтовые системы, системы разрешения сетевых конфликтов и т.п.).

ТКС как объект управления характеризуется топологией физических связей между управляемыми компьютерами ТКС. Топологической моделью глобальной ТКС будем называть граф сетевой конфигурации ТКС, вершинам которого соответствуют компьютеры (или локальные компьютерные сети), а ребрам – физические (электрические) связи между ними. Компьютеры (или локальные сети) глобальной ТКС называются узлами сети.

Оптимизация топологических моделей ТКС обычно (из экономических соображений) осуществляется по критерию минимизации суммарной длины физических каналов связи. Однако для повышения надежности ТКС и балансировки трафика (загрузки) отдельных каналов иногда вводятся избыточные (резервные) каналы связи.

В локальных ТКС с небольшим количеством (несколько десятков) компьютеров топологическая модель однородна и соответствует одной из типовых сетевых конфигураций: «кольцо», «общая шина», «звезда», «ячеистая» и «полносвязная» [1].

Глобальные ТКС создаются на основе соединения удаленных локальных ТКС, которые играют роль большой ТКС. Поэтому глобальные ТКС декомпозируются на однородные локальные ТКС, имеющие одну из типовых конфигураций, а их топологическая модель является смешанной (гетерогенной) .

5. Особенности адаптивного и интеллектуального управления информационными потоками

Сложность современных глобальных ТКС, обеспечивающих доступ к колоссальным информационным и вычислительным ресурсам, такова, что в процессе управления ими центральную роль играют люди-

профессионалы в области информационных и телекоммуникационных технологий. Однако по оценкам специалистов, если такая глобальная ТКС, как Internet, будет расти такими же темпами, как это происходит в начале XXI века, то через 10 лет потребуется 200 млн. человек для управления и обслуживания Internet–пользователей. Поэтому возникает острая необходимость в автоматизации процессов управления и обработки информации в глобальных ТКС по запросам пользователей.

Решение этой задачи наталкивается на трудности, связанные с отсутствием формализованных моделей основных компонентов и процессов, происходящих в в ТКС, неизученностью изменяющихся условий эксплуатации ТКС и т.п. Без преодоления этих трудностей невозможна разработка теории автоматического управления ТКС с адаптацией к изменяющемуся трафику средств автоматизированного проектирования систем управления информационными процессами и коммуникационным оборудованием в ТКС нового поколения.

Предоставление пользователям ТКС распределенных информационных и вычислительных ресурсов происходит в условиях неопределенности и нестационарности, т.е. при недостатке сведений о текущем состоянии ТКС и постоянно изменяющейся информационной среды.

Главными факторами неопределенности и нестационарности являются:

1) неопределенность или непредсказуемые изменения гетерогенного трафика в ТКС;
2) непредсказуемые изменения числа пользователей ТКС;
3) априорная непредсказуемость характера запросов агентов-пользователей, связанная с их текущими локальными интересами;
4) возможность сетевых конфликтов в ТКС и т.п.

В этих реально складывающихся условиях неопределенности и нестационарности возникает необходимость в робастном, адаптивном или интеллектуальном управлении. Дело в том, что именно робастные и адаптивные системы управления способны обеспечивать правильное функционирование ТКС в заранее неопределенных и непредсказуемо изменяющихся условиях за счет сигналов обратной связи о текущем состоянии ТКС и компенсации возникающих факторов неопределенности.

Интеллектуальные системы управления, наследуя свойства робастности и адаптивности, обладают дополнительными функциями искусственного интеллекта. Иначе говоря, они автоматически (т.е. самостоятельно) могут решать некоторые интеллектуальные задачи. Например, интеллектуальные системы управления могут диагностировать состояния ТКС, распознавать сетевые конфликты и обеспечивать их предотвращение или автоматическое разрешение.

ТКС с адаптивным управлением будем называть адаптивным ТКС, а ТКС с интеллектуальным управлением – интеллектуальными ТКС. Такие адаптивные интеллектуальные ТКС ведут себя как самонастраивающиеся «разумные» системы, способные самостоятельно преодолевать трудности и решать возникающие в ТКС проблемы прежде, чем пользователь ТКС узнает о них.

Отличительными чертами таких ТКС, относящихся к новым поколениям компьютерных сетей, являются следующие:

- постоянная доступность и готовность удовлетворять любые (допустимые) запросы пользователей, т.е. способность к диалогу;
- приспособляемость (адаптивность) к факторам неопределенности и способность «разумно» реагировать на непредсказуемые события (изменения трафика или числа пользователей, сетевые конфликты и т.п.);
- автономная реконфигуративность и восстанавливаемость в случае сбоев и отказов (например, при отключении или неисправности каких-то каналов связи или узлов ТКС) или непредсказуемых изменений внешней среды;
- самозащита от возможных угроз и атак, направленных на потерю работоспособности ТКС или её главных компонент.

Поскольку глобальная ТКС состоит из многих локальных компонент, очень важно, чтобы все управляющие компоненты были адаптивным или интеллектуаль-ными. Тогда суперпозиция и сетевая интеграция этих компонент обеспечивает адаптивность и интеллектуальность ТКС в целом. Здесь уместна аналогия локальных и глобальных ТКС нового поколения с человеком или коллективом людей, которые состоят из множества взаимосвязанных саморегулирующихся адаптивных и интеллектуальных подсистем,

поддерживающих не только их «живучесть», но и достижение ими локальных (частных) или глобальных (коллективных) целей и потребностей.

Заключение

В контексте изложенного выше интересна новая концепция IBM, изложенная в 2002 г. в манифесте “Autonomic Computing: IBM’s Perspective on State of Information Technology”. В поисках решения в IBM Research обратили внимание на организацию управления в живой природе. Любой живой организм состоит из множества саморегулирующихся систем и подсистем. Элементарную систему с самоуправлением называют autonomic, а суперпозицию таких систем - autonomic computing. Меморандум построен как призыв к учёным академических организаций и представителям коммерческих компаний к тому, чтобы осознать, что наступили “времена больших перемен”.

Работа выполнена при поддержке гранта Министерства промышленности, науки и технологий РФ № 37.029.11.0027 и гранта Администрации Санкт-Петербурга в сфере научной и научно-технической деятельности № 244.

Список литературы

1. Олифер В.Г., Оливер М.А. Компьютерные сети. Принципы, технологии, протоколы. - Спб.: Питер, 2001. - 672с.
2. Уолренд Дж. Телекоммуникационные и компьютерные сети. Вводный курс.- М.: Постмаркс. - 2001. - 480 с.
3. Timofeev A.V. Intelligent Control Applied to Non-Linear Systems and Neural Networks with Adaptive Architecture.- Journal of Intelligent Control, Neurocomputing and Fuzzy Logic, 1996, pp. 1-18.
4. Timofeev A.V. Intelligent Control and Multi-Agent Navigation. - Proceedings of the IAR Conference and Associated IAR/ICD Workshop (Strasbourg, November 22-23, 2001), pp. 123-127.
5. Тимофеев А.В., Сырцев А.В. Нейросетевые методы оптимальной маршрутизации потоков данных. - Материалы конференции “Региональная информатика-2002” (Санкт-Петербург, 26-28 ноября 2002), часть 1, с.87-88.
6. Тимофеев А.В. Интеллектуальное и мульти-агентное управление сложными системами. - В сб. “Экстремальная робототехника” (Материалы XI научно-технической конференции) - СПб: Изд-во СПбГТУ, 2001, с. 9-16.

Сведения об авторе

Тимофеев Адиль Васильевич - доктор технических наук, профессор, Заслуженный деятель науки Российской Федерации. Область научных интересов:

1) нейроинформатика и нейроуправление,
2) искусственный интеллект, базы знаний,
3) робототехника и мехатроника,
4) теория адаптивного, интеллектуального и мульти-агентного управления.

Тимофеев А.В. в настоящее время является заведующим лабораторией нейроинформатики и интеллектуального управления СПИИРАН и профессором СПбГУ и СПбГУАП, имеет тесные научные контакты с университетами и научными организациями Китая, США, Германии, Франции, Кореи, Польши, Болгарии и ряда других стран.

МУЛЬТИ-АГЕНТНАЯ И НЕЙРОСЕТЕВАЯ МАРШРУТИЗАЦИЯ ПОТОКОВ ДАННЫХ В ТЕЛЕКОММУНИКАЦИОННЫХ СЕТЯХ.

А.В.Тимофеев, А.В.Сырцев

Аннотация: *Рассматривается задача мульти-агентной маршрутизации в статических телекоммуникационных сетях с фиксированной конфигурацией. Задача решается в двух постановках: для централизованной схемы с агентом-координатором (глобальная маршрутизация) и для распределенной схемы с независимыми агентами (локальная маршрутизация). Для каждой постановки задачи и схемы строится модель нейронной сети решающей соответствующую задачу на основе нейронной сети Хопфилда (НСХ).*

Введение

При постановке и решении задачи оптимальной маршрутизации потоков данных в локальных и глобальных телекоммуникационных сетях (ТКС) обычно рассматривается только один узел-источник данных и один узел-приемник данных. В этом случае при выборе оптимального маршрута не учитывается возможность параллельной работы различных узлов-приемников данных (клиент ТКС) и узлов-источников данных (информационных ресурсов). Поэтому при коллективном (мульти-агентном) использовании ТКС для поиска информационных ресурсов могут возникать сетевые конфликты и перегрузки ТКС и, как следствие, снижение эффективности ТКС вплоть до потери работоспособности.

В данной статье рассматривается модель мульти-агентного распределения информационных потоков в ТКС, при котором учитываются интересы всех участников-агентов процесса поиска и передачи необходимых данных. При постановке задачи выделены два критерия оптимизации распределения потоков данных: глобальный, при котором оптимизируется общая нагрузка на ТКС, и локальный, при котором оптимизируется распределение информационного потока для каждой пары источник-приемник данных.

В качестве эффективных вычислительных моделей для решения подобных оптимизационных задач маршрутизации удобно применять нейронные сети Хопфилда (НСХ) [3]-[6]. В данной работе исследована возможность применения и построены модели НСХ для обоих критериев оптимизации.

Сначала рассматриваются задачи мульти-агентной маршрутизации потоков данных в двух вариантах:

- для централизованной схемы с координатором(«глобальная оптимальность»), когда принятие решений осуществляется специальным ведущим агентом-координатором;
- для распределенной схемы («локальная оптимальность»), когда каждый агент ТКС принимает решение самостоятельно;

В статье обсуждается и доказывается возможность решения поставленных задач к ТКС с ограниченной пропускной способностью каналов связи. Предлагаемые решения задач мульти-агентной маршрутизации потоков данных по централизованной и распределенной схемам основываются на использовании НСХ, адаптированных к условиям задачи.

В заключении резюмируются основные результаты работы и рассматриваются перспективные направления дальнейших исследований.

Постановка задачи

В качестве математической модели статической сети, т.н. ТКС, не изменяющейся с течением времени, будем рассматривать граф

$$G(V, E(V)) \qquad (1)$$

где V – множество узлов, E – упорядоченное множество направленных дуг. Для данного графа, моделирующего ТКС, рассмотрим множество пар его узлов D_0:

$$D_0 = \{(s,d) \mid s,d \in V, s \neq d\} \quad (2)$$

где первый элемент пары является узлом-источником необходимых данных, а второй – узлом-получателем запрошенных данных. Таким образом, любая задача мульти-агентной маршрутизации информационных потоков может быть описана конечным набором таких пар узлов ТКС из множества D_0. Обозначим этот набор как D ($D \subset D_0$) и будем считать, что он фиксирован, т.е. не зависит от времени.

Введем следующие обозначения:

Π_d – множество возможных маршрутов для данной пары узлов $d \in D$,

Π – упорядоченное множество всех маршрутов для всех пар узлов из D,

Φ_d – интенсивность информационного потока между узлами пары $d \in D$,

$\Phi = (\Phi_1, \Phi_2, \ldots)$ – вектор распределения интенсивности информационных потоков,

$\Phi(D) = \sum_{d \in D} \phi_d$ (или Φ_D) – общая интенсивность информационных потоков D,

x_p – общая интенсивность информационного потока на маршруте $p \in \Pi$,

$x = (x_1, x_2, \ldots)$ – вектор распределения информационных потоков,

δ_{lp} – доля интенсивности информационного потока x_p, которая приходится на дугу l,

ρ_l – нагрузка на дугу $l \in E(V)$, определяемая по формуле $\rho_l = \sum_{p \in \Pi} \delta_{lp} x_p$,

$\rho = (\rho_l, \rho_2, \ldots)$ – вектор распределения нагрузок на дуги,

$T_l(p_l)$ –стоимость нагрузки ρ_l, приходящейся на дугу l,

T_p – средняя стоимость маршрута $p \in \Pi$,

$T = (T_1, T_2, \ldots)$ – вектор распределения стоимостей по маршрутам.

На возможных маршрутах передачи данных Π_d зададим матрицу инциденций

$$\Gamma = \begin{bmatrix} \gamma_{11} & \gamma_{12} & \cdots \\ \gamma_{21} & \gamma_{22} & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix}, \quad \text{где } \gamma_{pd} = \begin{cases} 1, \mathrm{p} \in \Pi_d \\ 0, \mathrm{p} \notin \Pi_d \end{cases} \quad (3)$$

Для формализации и решения задачи оптимальной маршрутизации введем дополнительные условия:

I. Средняя стоимость маршрута определяется как сумма стоимостей нагрузок на его дуги $T_p = \sum_{l \in E(V)} \delta_{lp} T_l(p_l)$;

II. Для любой дуги $l \in E(V)$ справедливо $T_l : [0, \infty) \to [0, \infty]$ и $T_l(0) < \infty$;

III. Для каждой дуги $l \in E(V)$, $T_l(p_l)$ выпукла и либо строго монотонно возрастает на интервале, где $T_l(p_l) < \infty$, либо $T_l(p_l) = \text{const}$;

IV. Функция $T_l(p_l)$ непрерывна на всей области определения, причем на интервале, где $T_l(p_l) < \infty$, она непрерывно дифференцируема.

Для централизованной схемы с агентом-координатором решение задачи мульти-агентной маршрутизации представляет собой некоторое распределение информционных потоков, при котором общие затраты на их обслуживание минимальны. Пусть F – общая стоимость распределения информационных потоков. Тогда

$$F = \sum_{p \in \Pi} \frac{x_p}{\Phi(D)} T_p = \frac{1}{\Phi(D)} \sum_{l \in E(V)} \rho_l T_l(\rho_l). \quad (4)$$

Для любого маршрута $p \in \Pi$ и для любой пары узлов $d \in D$ справедливы соотношения:

$$x_p \geq 0, \qquad \sum_{p \in \Pi d} \gamma_{pd} x_p = \varphi_d \,. \tag{5}$$

Таким образом, постановка задачи для централизованной схемы мульти-агентной маршрутизации запишется следующим образом:

$$F \rightarrow \min \tag{6}$$

при следующих ограничениях:

$$\Gamma^T x = \phi, \qquad x \geq 0 \,. \tag{7}$$

Для распределенной схемы мульти-агентной маршрутизации задача ставится для каждой пары $d \in D$ отдельно. В данном случае оптимальным решением является распределение информационного потока, при котором его стоимость для каждой пары в отдельности – минимальна. Такое решение является локально оптимальным.

Введём понятие минимального потока между парой узлов d как функцию: $A_d(x) = \min_{p \in \Pi d} T_p(x)$, $d \in D$.

Тогда, как было показано в работе [1], решением задачи будет вектор распределения потоков x, удовлетворяющий следующим соотношениям:

$$\left(T(x) - \Gamma A(x)\right) \cdot x = 0, T(x) - \Gamma A(x) \geq 0, \qquad \Gamma^T x - \varphi = 0, x \geq 0 \,, \tag{8}$$

где $A(x) = (A_1(x), A_2(x), \ldots)$ – вектор минимальных потоков.

Модификация стоимости нагрузки на канал связи для ТКС с ограниченной пропускной способностью

Для ТКС, в которых пропускная способность каналов связи ограничена, функция $T_l(p_l)$ будет удовлетворять следующим ограничениям:

$$T_l(p_l) < \infty, 0 \leq p_l \leq p_{\max}, \quad T_l(p_l) = \infty, p_l > p_{\max} \,, \tag{9}$$

где p_{max} – максимальная пропускная способность канала связи *l*.

В этом случае условие IV будет нарушено. Этого можно избежать, если ввести вспомогательную функцию $T_l^{(\varepsilon)}(p_l)$:

$$T_l^{(\varepsilon)}(p_l) = \begin{cases} T_l(p_l), p_l \notin [p_{\max} - \varepsilon, p_{\max}) \\ T_l(p_l)\left(1 - \frac{\pi}{2\varepsilon}(p_l - (p_{\max} - \varepsilon)) + tg\left(\frac{\pi}{2\varepsilon}(p_l - (p_{\max} - \varepsilon))\right)\right), p_l \in [p_{\max} - \varepsilon, p_{\max}) \end{cases}, \tag{10}$$

где величина ε>0 и сколь угодно мала.

Лемма: Для $T_l^{(\varepsilon)}(p_l)$ вида (10) выполняются условия I-IV.

Доказательство: Очевидно, что условия I и II будут выполнены. Докажем, что $T_l^{(\varepsilon)}(p_l)$ непрерывно дифференцируема на $[0, p_{max})$.

Рассмотрим функцию $z(p_l) = \frac{\pi}{2\varepsilon}(p_l - (p_{\max} - \varepsilon))$. Она непрерывно дифференцируема на всей области определения и строго монотонно возрастает, причем

$$z(p_{\max} - \varepsilon) = 0, \qquad z'(p_{\max} - \varepsilon) = z'(p_l) = \frac{\pi}{2\varepsilon} \,. \tag{11}$$

Рассмотрим $T_l^{(\varepsilon)}(p_l)$ в точке $\{p_{max}-\varepsilon\}$. Учитывая (11), получаем:

$$\begin{aligned} T_l^{(\varepsilon)}(p_{\max} - \varepsilon) &= T_l(p_{\max} - \varepsilon)\left(1 - z(p_{\max} - \varepsilon) + tg(z(p_{\max} - \varepsilon))\right) = \\ &= T_l(p_{\max} - \varepsilon)\left(1 - 0 + tg(0)\right) = T_l(p_{\max} - \varepsilon) \end{aligned} \tag{12}$$

Из (10) следует, что в точке $\{p_{max}\}$ функция $T_l^{(\varepsilon)}(p_l)$ стремится к ∞, т.е. $T_l^{(\varepsilon)}(p_l)$ непрерывна на всей области определения. С учетом (10) и (11) получаем следующее выражение для производной вспомогательной функции:

$$\left(T_l^{(\varepsilon)}(p_l)\right)' = \begin{cases} T'_l(p_l),\ p_l \in (0,\ p_{\max} - \varepsilon_x), \\ T'_l(p_l)\left(1 - z(p_l) + tg(z(p_l))\right) + T_l(p_l)z'(p_l)\left(\dfrac{1}{\cos^2 z(p_l)} - 1\right),\ p_l \in (p_{\max} - \varepsilon,\ p_{\max}). \end{cases} \quad (13)$$

Отсюда следует, что функция $T_l^{(\varepsilon)}(p_l)$ непрерывно дифференцируема на $(0,p_{max})\backslash\{p_{max} - \varepsilon\}$. Рассмотрим пределы её производной слева и справа от точки $\{p_{max}-\varepsilon\}$.

$$\lim_{p_l \to p_{\max} - \varepsilon - 0} \left(T_l^{(\varepsilon)}(p_l)\right)' = (T_l(p_{\max} - \varepsilon))'$$

$$\lim_{p_l \to p_{\max} - \varepsilon + 0} \left(T_l^{(\varepsilon)}(p_l)\right)' = \lim_{p_l \to p_{\max} - \varepsilon + 0} \left(T_l(p_l)'\left(1 - z(p_l) + tg(z(p_l))\right) + T_l(p_l)z'(p_l)\left(\frac{1}{\cos^2 z(p_l)} - 1\right)\right) =$$

$$= (T_l(p_{\max} - \varepsilon))'(1 - 0 + tg(0)) + \frac{\pi}{2\varepsilon} T_l(p_{\max} - \varepsilon) tg^2(0) = (T_l(p_{\max} - \varepsilon))'.$$

Так как из этого следует, что $\lim_{p_l \to p_{\max} - \varepsilon - 0} \left(T_l^{(\varepsilon)}(p_l)\right)' = \lim_{p_l \to p_{\max} - \varepsilon + 0} \left(T_l^{(\varepsilon)}(p_l)\right)'$, то функция $T_l^{(\varepsilon)}(p_l)$ непрерывно дифференцируема на $(0,p_{max})$. Следовательно, условие IV выполняется для вспомогательной функции (10).

Для выполнения условия III достаточно доказать, что оно выполняется на промежутке $(p_{max}-\varepsilon,\ p_{max})$. На этом интервале функция $T_l^{(\varepsilon)}(p_l)$ как произведение двух выпуклых и положительных функций будет выпуклой и строго монотонно возрастающей.

Таким образом, лемма доказана.

Отсюда следует, что, рассматриваемые оптимизационные методы мульти-агентной маршрутизации применимы к ТКС с ограниченной пропускной способностью каналов связи. При этом оптимальные маршруты строятся с некоторой погрешностью.

Централизованная схема мульти-агентной маршрутизации

Рассмотрим оптимизационную задачу (6), (7). Как было показано в работе [1], при выполнении условий I-IV, эта задача имеет, по крайней мере, одно решение. При этом для всех решений значение вектора распределения нагрузок ρ будет одним и тем же.

В работе [2] была рассмотрена схема решения подобных систем при помощи нейронных сетей, а в работе [5] рассматривалась реализация этой схемы с помощью НСХ. Достаточным условием решения оптимизационной задачи (6), (7) с линейными ограничениями при помощи НСХ, является строгое монотонное возрастание минимизируемой функции ([2]). Рассмотрим систему (4). Так как для конкретного множества $D \subset D_0$ значением функции Φ_D является положительная константа, то её можно перенести в левую часть, не меняя условий задачи, т.е.:

$$F\Phi D = \sum_{l \in E(V)} \rho l T l(\rho l). \quad (14)$$

Введём матрицу Δ:

$$\Delta = \begin{pmatrix} \delta_{11} & \delta_{12} & \cdots \\ \delta_{21} & \delta_{22} & \cdots \\ \vdots & \vdots & \ddots \end{pmatrix}, \quad (15)$$

где δ_{ij} – доля интенсивности информационного потока x_j, которая приходится на дугу *i*.

Так как $\rho_l = \rho_l(x)$, то задачу (6)-(7) можно переформулировать следующим образом:

$$(F\Phi_D) \rightarrow \min \qquad (16)$$

при ограничениях

$$\Gamma^T x = \Phi, \qquad \Delta x = \rho, \qquad x \geq 0\,. \qquad (17)$$

В качестве возможных решений задачи (16), (17) будем искать векторы (ρ, x). Без потери общности можно считать, что $0 \leq \rho \leq 1$ и $0 \leq x \leq 1$ (подробный механизм приведения задачи к такому виду описан в [2]).

Построим энергетическую функцию E = E(ρ, x) для НСХ, решающей оптимизационную задачу (16), (17). Потребуем, чтобы функция E была квадратичной формой от (ρ, x).

Сначала рассмотрим функцию E_0:

$$E_0 = \frac{\alpha_{11}}{2}\Big(\sum_{l\in E(V)} \rho_l T_l(\rho_l)\Big)^2 + \sum_{d\in D}\frac{\alpha_{2d}}{2}\Big(\sum_{p\in\Pi}\gamma_{pd}x_p - \varphi_d\Big)^2 + \sum_{l\in E(V)}\frac{\alpha_{2l}}{2}\Big(\sum_{p\in\Pi}\delta_{lp}x_p - \rho_l\Big)^2, \qquad (18)$$

где α_{ij} - некоторые положительные константы, причем α_{11} – достаточно малое число ([5]). Однако E_0 не является квадратичной формой, так как в первой сумме присутствуют нелинейные элементы $T_l(\rho_l)$. Заменим первую сумму в (18) на квадрат линейной комбинации ρ. Тогда получим следующую энергетическую функцию для НСХ:

$$E_0 = \frac{\alpha_{11}}{2}\Big(\sum_{l\in E(V)} c_l\rho_l\Big)^2 + \sum_{d\in D}\frac{\alpha_{2d}}{2}\Big(\sum_{p\in\Pi}\gamma_{pd}x_p - \varphi_d\Big)^2 + \sum_{l\in E(V)}\frac{\alpha_{2l}}{2}\Big(\sum_{p\in\Pi}\delta_{lp}x_p - \rho_l\Big)^2 \qquad (19)$$

где $c_l \geq 0$ и достаточно малы.

Важно отметить, что основным требованием при составлении энергетической функции для решения подобных систем с линейными ограничениями является достаточно малое значение слагаемого соответствующего минимизируемой функции ([2]), чтобы приближённое решение не слишком сильно отличалось от точного. Поэтому коэффициенты линейной комбинации берутся достаточно малыми. В то же время, нельзя задавать их слишком малыми, поскольку это замедлит сходимость процесса поиска решения.

Замена (18) на (19) допустима, поскольку для двух строго возрастающих функций с одинаковыми областями определения (задаваемыми системой (17)) экстремумы достигаются в одинаковых точках.

Таким образом, синтезирована модель НСХ, состоящая из |E|+|П| нейронов ([5]) и адаптированная к условиям централизованной схемы мульти-агентной маршрутизации.

Распределенная схема мульти-агентной маршрутизации

Рассмотрим задачу локальной оптимизации (8). В работе [1] показано, что при выполнении условий I-IV решение этой задачи существует. Рассмотрим первое уравнение системы (8). В силу второго и четвертого неравенств системы получим, что для любого x справедливо неравенство:

$$\big(T(x) - \Gamma A(x)\big)\cdot x \geq 0 \qquad (20)$$

Отсюда следует, что неотрицательная функция (T(x)-Гу)x принимает нулевые значения (с учетом остальных ограничений) в точках возможных решений неравенств (20). Это означает, что точки минимумов данной функции совпадают с решениями оптимизационной задачи (8). С учетом (20) и того, что

$$T(x)x - \Gamma A(x)x = (F\Phi_D) - \Gamma A(x)x \qquad (21)$$

переформулируем задачу (8) следующим образом:

$$\begin{aligned} &(F\Phi_D) - \Gamma A(x)x \rightarrow \min, \\ &T(x) - \Gamma A(x) \geq 0, \qquad \Gamma^T x - \varphi = 0, \qquad x \geq 0 \end{aligned} \qquad (22)$$

Согласно работам [2] и [5] задачу (22) можно свести к следующей задаче:

$$(F\Phi_D) - \Gamma A(x)x \to \min,$$
$$T(x) - \Gamma A(x) - z = 0, \qquad \Gamma^T x - \varphi = 0, \qquad x \geq 0, \qquad z \geq 0 \tag{23}$$

Заметим, что z_p=0, если $T_p(x) = (\Gamma A(x))_p$. Поэтому с учетом неравенств $x_p \geq 0$ и $z_p > 0$ получим, что $T_p(x) > (\Gamma A(x))_p$ и x_p=0.

Построим НСХ, решающую оптимизационную задачу (23). В качестве возможных решений будем искать векторы (x, z). Так же, как и в предыдущем параграфе, будем полагать, что $0 \leq x \leq 1$ и $0 \leq z \leq 1$. Из первого равенства в системе ограничений следует, что

$$\left(F\Phi_D\right) - \Gamma A(x)x = zx \tag{24}$$

Таким образом, энергетическая функция E для НСХ будет иметь следующий вид:

$$E = \alpha_{11}\left(\sum_{p\in\Pi} z_p x_p\right) + \sum_{p\in\Pi}\frac{\alpha_{2p}}{2}\left(T_p(x) - \sum_{d\in D}\gamma_{pd}A_d - z_d\right)^2 + \sum_{d\in D}\frac{\alpha_{3d}}{2}\left(\sum_{p\in\Pi}\gamma_{pd}x_p - \varphi_d\right)^2 \tag{25}$$

Соответствующая (25) модель НСХ состоит из 2|П| нейронов [5] и адаптирована к условиям задачи локальной оптимизации для распределенной схемы мульти-агентной маршрутизации.

Заключение

Проведенные исследования показали применимость нейронных сетей для решения задачи мульти-агентной маршрутизации потоков данных в статических ТКС, конфигурация которых не меняется с течением времени. Были построены две модели НСХ для глобальной и локальной оптимизации распределения информационных потоков. По запросам агентов-пользователей ТКС. Кроме того, была исследована возможность адаптации полученных нейросетевых решений к ТКС с ограниченной пропускной способностью каналов связи.

Общим недостатком нейросетевой маршрутизации для распределенной и централизованной схем является большой объем предварительных вычислений (прокладка всевозможных маршрутов, расчёт весов, зависящий от большого числа параметров и т.д.). Поэтому в дальнейших исследованиях планируется преодолеть возникающие трудности и недостатки.

Работа выполнена при поддержке госконтракта №37.029.11.0027 Минпромнауки РФ.

ЛИТЕРАТУРА:

1 *E. Altman, H. Kameda,* Equilibria for Multiclass Routing in Multi-Agent Networks
2. *A. Cichocki, A. Bargiela,* Neural networks for solving linear inequality systems
3 *Ch. Pornavalai, G. Chakraborty, N. Shiratiri,* A neural network approach to multicast routing in real-time communication networks, IEEE, 1995
4 *W. Newton,* A neural network algorithm for internetwork routing, Technical report, 2002
5 *Тимофеев А.В., Сырцев А.В.* Нейросетевые методы оптимальной динамической маршрутизации потоков данных. - Материалы конференции “Региональная информатика-2002” (Санкт-Петербург, 26-28 ноября 2002), часть 1, с.87-88.
6 *Тимофеев А.В., Сырцев А.В.* Модели адаптивного управления потоками данных на базе нейросетевых маршрутизаторов. - Труды X Всероссийского семинара “Нейроинформатика и её приложения” (4-6 октября, 2002, Красноярск).
7 *Тимофеев А.В.* Проблемы и методы адаптивного управления потоками данных в телекоммуникационных системах. - Материалы конференции “Региональная информатика-2002” (Санкт-Петербург, 26-28 ноября 2002), часть 1, с.87.

Сведения об авторах

А.В.Тимофеев - Санкт-Петербургский институт информатики и автоматизации РАН, adil@iias.spb.su

А.В.Сырцев - Санкт-Петербургский государственный университет, airleks@mail.ru.com

NEURAL-LIKE GROWING NETWORKS IN INTELLIGENT SYSTEM OF RECOGNITION OF IMAGES

Vitaliy Yashchenko

Abstract: *The neural-like growing networks used in the intelligent system of recognition of images are under consideration in this paper. All operations made over the image on a pre-design stage and also classification and storage of the information about the images and their further identification are made extremely by mechanisms of neural-like networks without usage of complex algorithms requiring considerable volumes of calculus. At the conforming hardware support the neural network methods allow considerably to increase the effectiveness of the solution of the given class of problems, saving a high accuracy of result and high level of response, both in a mode of training, and in a mode of identification.*

Keywords: *Neural-like networks, images recognition.*

Introduction

The need in intellectual autonomous macro and micro robots, which can replace the man in environment, unacceptable and dangerous to his health and threat of the terrorist acts, steadily grows. The complex control systems are used for the decision of a complex of tasks on management of robots in extreme conditions. To correct orientation in the surrounded environment a robot must "understand" this environment and have its model. One can say that "consciousness" of robot is its ability to reflect a surrounding world in its memory, to model it in the process of its activity. Exactly this characteristic distinguishes intellectual robots from robots of previous generations, whose actions are done "unconsciously", subjected to strict programs.

The solution of such problems by traditional methods of mathematical programming frequently oriented on computer facilities with the consecutive architecture is integrated to temporary costs unacceptable for many applications.

Nowadays, the next rise of intensity of researches in the field of artificial neural networks is observed. The volume of financing of the projects using the technologies of neural networks in Japan, China, USA and Europe is about hundred million dollars.

In the given article we represent the new class of neural networks and its usage in the system of recognition of images.

Neural-like growing networks

The neural-like growing networks are the new class of neural networks, which are designed within the frameworks of bionic approach on the basis of technologies integration of a data processing in growing semantic and neural networks.

Neural-like growing networks (n-GN) were created specially for system engineering of artificial intelligence.

Multidimensional receptor - effector neural-like growing network (iren-GN) is actually a model of a brain of the man. In them, during functioning (life cycle) of system, the information on the external world accumulates. Simultaneously the own structure of a network is formed at the expense of "birth" (occurrence) of neurons and occurrence and disappearance of connections between them. Actually this structure is a model of the external world in the system. Therefore the class neural-like of growing networks has a wide spectrum of application. We have applied n-GN in the system of recognition of images, about which the speech will go further.

The neural-like growing network is a set of interdependent neural-like elements intended for a reception and transformation of information and in the process of reception of information the network is increasing in size.

The neural-like growing network (n-GN) can be submitted as a non-cyclic digraph with fluidized connections, in which neural-like elements are the tops. The tops which do not have entering arcs, are called the receptors, the rest of them are the neural-like elements.

Depending on the applied area, in which n-GN are used, the receptor can introduce the character, sign, parameter, meaning of physical or economical index, elementary fact from the description of a situation, etc. The neural-like elements correspond to the descriptions of visual images, words, phrases, subjects, objects, processes, plans, situations or phenomena.

The specific peculiarity of n-GN is the capability to adapt for these descriptions, changing the structure accordingly. The adaptation is accompanied by input in the network of new tops and arcs at transition of any group of receptors and neural-like elements in a condition of excitation. The class of neural-like growing networks consists of neural-like growing (receptor) networks themselves, multidimensional neural-like growing (receptor) networks, receptor and effective neural-like growing networks and multidimensional receptor and effective neural-like growing networks.

Formally they are set in the following way.

Neural-like growing (receptor) networks (n-GN) $\boldsymbol{S = (R, A, D, M, P, N)}$, where $\boldsymbol{R} = \{r_i\}$, $i = \overline{1, n}$ - final set of receptors; $\boldsymbol{A} = \{a_i\}$, $i = \overline{1, k}$ - final set of neural-like elements; $\boldsymbol{D} = \{d_i\}$, $i = \overline{1, e}$, - final set of arcs linking receptors with neural-like elements and neural-like elements between themselves; $\boldsymbol{P}=\{P_i\}$, $i = \overline{1, k}$ $\boldsymbol{N}$ = h, where P - threshold of excitation of the top a_i, P = $f(m_i)$ > P_0 (P_0 - the minimum permissible threshold of excitation) provided that to the set of arcs $\boldsymbol{D}$, coming on top a_i, corresponds to the set of weighting coefficients $\boldsymbol{M}$ = $\{m_i\}$, $i = \overline{1, w}$, and m_i can receive both positive and negative meanings.

Multidimensional neural-like growing (receptor) networks (mn-GN) $\boldsymbol{S = (R, A, D, P, N)}$, where $\boldsymbol{R} \supset \boldsymbol{R_l, R_r, R_v}$, $\boldsymbol{A} \supset \boldsymbol{A_l, A_r, A_v}$, $\boldsymbol{D} \supset \boldsymbol{D_l, D_r, D_v}$, $\boldsymbol{N} = \{n_l, n_r, n_v\}$, here $\boldsymbol{R_l, R_r, R_v}$ - final subset of receptors, $\boldsymbol{A_l, A_r, A_v}$ - final subset of neural-like growing elements, $\boldsymbol{D_l, D_r, D_v}$ - final subset of arcs, $\boldsymbol{P_l, P_r, P_v}$ - final set of thresholds of excitation of neural-like elements, that belong, for example, to linguistic, voice or visual information spaces. $\boldsymbol{N}$ - final set of floating factors of cohesion.

Receptor and effective neural-like growing networks (ren-GN) $\boldsymbol{S = (R, A_r, D_r, P_r, M_r, N_r, A_e, D_e, P_e, M_e, E, N_e)}$, where $\boldsymbol{R} = \{r_i\}$ - final set of receptors, $\boldsymbol{Ar} = \{a_{ri}\}$, - final set of neural-like elements of the receptor zone, $\boldsymbol{Dr} = \{d_{ri}\}$, - final set of arcs of a receptor zone, $\boldsymbol{E} = \{e_i\}$, - final set of effectors, $\boldsymbol{Ae} = \{a_{ei}\}$, - final set of neural-like elements of effecting zone, $\boldsymbol{D_e} = \{d_{ei}\}$, - final set of arcs of effecting zone, $\boldsymbol{N_r, N_e}$ - final set of floating factors of cohesion receptor and effecting zones accordingly, $\boldsymbol{P_r, P_e}$ - the threshold of excitation of tops a_{ri}, a_{ei}, $\boldsymbol{M} = \{m_i\}$ is a set of weighting coefficients, besides m_i can receive both positive and negative meanings.

Multidimensional receptor and effecting neural-like growing networks (mren-GN) $\boldsymbol{S = (R, A_r, D_r, P_r, M_r, N_r, A_e, D_e, P_e, M_e, E, N_e)}$, $\boldsymbol{R \supset R_v, R_s, R_t, A \supset A_v, A_s, A_t, D \supset D_v, D_s, D_t, P \supset P_v, P_s, P_t, M \supset M_v, M_s, M_t, N \supset N_v, N_s, N_t, E \supset E_r, E_{d1}, E_{dn}, A \supset A_r, A_{d1}, A_{dn}, D \supset D_r, D_{d1}, D_{dn}, P \supset P_r, P_{d1}, P_{dn}, M \supset M_r, M_{d1}, M_{dn}, N \supset N_r, N_{d1}, N_{dn}$ here $\boldsymbol{R_v, R_s, R_t}$ - final subset of receptors, $\boldsymbol{A_v, A_s, A_t}$ - final subset of neural-like elements of receptor zone, $\boldsymbol{D_v, D_s, D_t}$ - final subset of arcs of receptor zone, $\boldsymbol{P_v, P_s, P_t}$ - final set of thresholds of excitation of neural-like elements of receptor zone, that belongs, for example, to visual, sound, tactile information spaces, $\boldsymbol{N_r}$- final set of float factors of cohesion of receptor zone, $\boldsymbol{E_r, E_{d1}, E_{dn}}$ - final subset of effectors, $\boldsymbol{A_r, A_{d1}, A_{dn}}$ - final subset of neural-like elements of effecting zone, $\boldsymbol{D_r, D_{d1}, D_{dn}}$ - final subset of arcs of effecting zone, $\boldsymbol{P_r, P_{d1}, P_{dn}}$- final set of thresholds of excitation of neural-like elements of effecting zone, that belongs, for example, to voice information space and space of operating, $\boldsymbol{N_e}$ - final set of floating factors of cohesion of effecting zone.

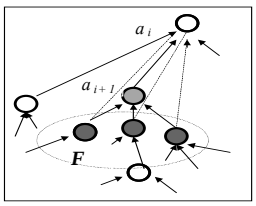


Figure1.Rule 1.

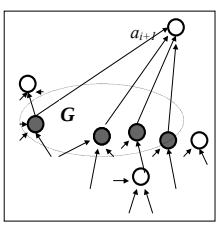


Figure 2. Rule 2.

The neural-like growing networks are the dynamic structure, which changes depending upon the meaning and time of reception of the information for receptors, and also for the previous condition of the network. In it the information on objects is represented by ensembles of exited tops and connections between them. The storage of the descriptions of objects and situations is accompanied by input in the network of new tops and arcs at transition of any group of receptors and neural-like elements in condition of excitation.

Thus, the combination of stable relationships of depicted object ensuring its integrity and identity to itself, i.e. the saving of the basic characteristics is formalized.

Information about objects and their classes is represented by ensembles of the associatively interconnected tops distributed on the structure of the net. The input of the new information in the network causes the process of its structure constructing (redistribution of the connections between already existing and again arising tops) with simultaneous excitation of the neural elements. In the result of this process the inclusion of the described object into the class, to which it belongs, is going on, or the new class of the objects is formed. So the classification and choosing the common attributes of the objects is carried out. Algorithm of the network construction establishes automatically the associative connections between descriptions of the objects accordingly their attributes. The description of the object or the class of the objects is located in some part of the network that lets to carry out various operations of associative search effectively. Profitability of the information representation in n-GN is carried out owing to compression of the information on each its level and representation of the identical combinations of attributes of several objects by one common subset of tops of a network. The training of the network is carried out simultaneously with their construction according to rules of construction and functioning of a network.

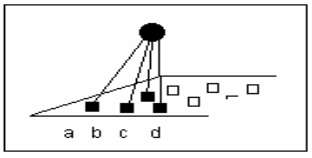


Figure 3.

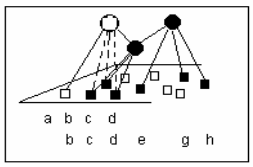


Figure 4. Image 2

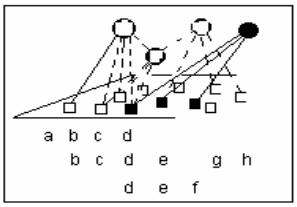


Figure 5. Image 3

Rule 1. If during the perception of information, a subset of tops ***F*** from the set of tops, having direct relationship Image 1with the top a_i, is excited, and $\overline{\overline{F}} \geq h$, the relationships of a top a_i with tops from the subset ***F*** are liquidated and a new top a_{i+1} joins the network, whose entries are connected with entries of all tops of the subset ***F***, and the exit of a top a_{i+1} is connected with one of the inputs of a top a_i, whereas the input relationships of the top a_{i+1} are assigned weighted factors m_i, corresponding to the weighted factors of liquidated relationships of the top a_i, and top a_{i+1} is assigned the threshold of excitation P_i, which equals $f(m_i)$, (function from weighted relationship factors, which fall into the top a_{i+1}). Outcoming relationship of this top is assigned a weighted factor m_i , which equal $f(P_i)$. Relationships, outcoming from receptors, are assigned a weighted factor,and $f(b_i)$, function from the code of sign b_i, correspond to a given receptor (fig.1).

Rule 2. If during the perception of information, a subset of tops ***G*** is excited, and $\overline{\overline{G}} \geq h$ a new associative top a_{i+1}, joins the network, which is connected by turning arcs with all tops of the subset ***G***. Each of turning arcs is assigned a weighted factor m_i, equal $f(P_i)$ of a corresponding top from the subset ***G***, and a new top a_{i+1} is assigned a minimum threshold of excitement P_i, equal to the function of weighted factors m_i of incoming arcs (fig.2).

Example of construction neural-like growing networks

The principle of constructing n-GN (for simplicity of perception) will be looked at the example of constructing the multiconnection growing network. Formally m-GN is described so: ***S=(R, A, D, N).***

Let be learning access, which consists of k-images : ***1.*** *a,b,c,d;* ***2.*** *b,c,d,e,g,h;* ***3****.d,e,f;* … ***k****. d,e,h.*

Let's set up variable coefficient of connectivity $N \geq 3$. In this case when entering the description of the first image (a,b,c,d) on the receptor field, the receptors 1,2,3,4 are changed over to the state of excitation. The vertex *a,b,c,d* is formed and the connections between vertex and excited receptors are set up (fig.3.). The vertex is changed over to the state of excitation. In a definite time the excitation is taken of from receptors and a vertex.

When entering the description of the second image (*b,c,d,e,g,h*) on the receptor's field ***R***, the receptors 2,3,4,5,7,8 are changed over to the state of excitation. The number of signs, coincided with the description of the first image (*b,c,d*)=3, then N=3, the vertex (*b,c,d*) and (*b,c,d,e,g,h*) are formed. The connection of the vertex *a,b,c,d* with receptors 2,3,4 are liquidated. Inputs of the vertex *b,c,d* are connected with receptors 2,3,4 and outputs of this vertex are connected with inputs of the vertex (*a,b,c,d*) and *b,c,d,e,g,h*, and these vertices are changed over to the state excitation (fig.4).In a definite time the excitation is taken off from the vertex (b,c,d) (b,c,d,e,g,h) and receptors.

When entering the description of the third-image on the receptor's field, the new vertex (d,e,f) is formed (fig.5).

When entering the description of k-image on the receptor's field, the new vertex (d,e,k) is formed (fig.6).

In this case the separation of the common signs, described notions are performed.

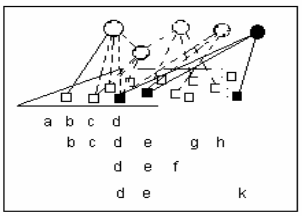


Figure 6. Image k

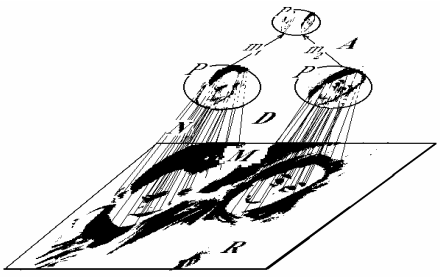


Figure 7. Storing the images

Thus the description of the notion (the vertex of the network) and signs are stored. Besides, the information, which enter the receptor's fields of the networks, is classified and structured automatically .

When forming new vertex in n-GN, weight coefficients of connection m_i and thresholds of the excitation of the vertex P_i are considered, that is, constructing n-GN is performed analogically with building m-GN, but in accordance with rules, which are described in the materials presented before.

Information in neural-like growing networks is stored as a result of its reflecting in the structure of a network. New information input into the network brings about a process of building its structure.

Neural-like growing networks are a dynamic structure, which changes depending on values and time of arrivals of image to receptors, as well as former condition of the network. Storing the images descriptions is accompanied by input in to the network of new tops and arcs when turning a group of receptors and neural-like elements became excited (fig.7). The process of excitation spreads on the network, as a wave [1,2].

System of identification of the person

So, first of all there is a problem of detection and allocation of the face on the image.

The program of detection of a face searches for it, sequentially scanning the image in all possible scales, since some minimum size of area (in pixels).

For recognition of faces the different methods are used.

The method of matching with a template is based on matching of the images with the usage of a mask, filter and so on. This method is one of the most simple methods, but it is not steady and responds to the noise.

The method of main components. In the given approach the two-measured image of the face is considered as a vector. If the image width *w* and altitude *h* pixels then, the number of components of this vector will be w*h. In the method the main components the integrated change in a set of faces and this change of several coordinates in new basis is described. Thus the dimension decreases. In the method of the main component statistical image processing will be used [3].

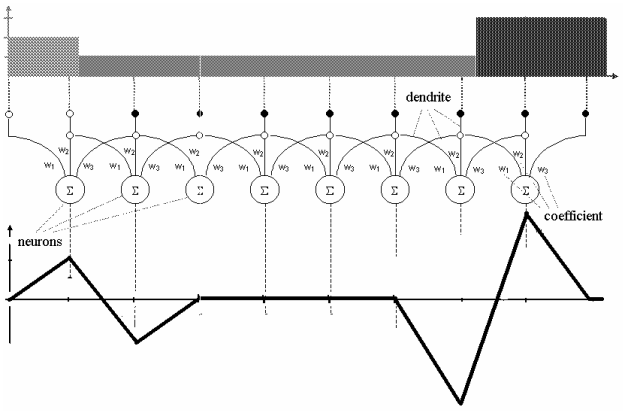
The method of "Distinctive features of the face" (Eigenface Technology). This method uses a set of distinctive features of faces, which represent a combination of light and dark areas. Such a set is formed by means of combination of all images with allocation of individuals having definite similar tags in separate groups. The distinctive features of each face act the role of the pieces of a face. For identification of a face the program compares its distinctive features, which are joint in so-called template of the face with templates of faces from the database, selecting those that coincide most of all [4-6].

Method of the analysis of local elements of the face (Local Feature Analysis). The given method goes from the previous one, but allows to get rid of such problems as sensitivity to deformation of the face, to its relative position, and also to the degree of illuminating intensity. The method of analysis of local elements of the face attracts the attention to concrete details of features of the face, instead of its common view. The method selects the elements, which fully determine each face. These elements play the role of building units, from which all images of the faces can be constructed [5-8].

There are also methods that perform the operations described above. So, the Miros company in its TrueFace ID technology combines the methods *of distinctive features and analysis of local elements of the face* realized in paradigm of artificial neural networks.

The suggested method is based on bionic approach, in which the data processing is realized by means of the neural network.

In bionic approach it is supposed that each conditional point of the image, accepted by man, corresponds to one neuron. Each such a neuron has some dendrites connected with adjacent neurons. The signal from the point of the image, corresponding to the neuron, goes in it through the receptor, strengthening by a positive weighting coefficient of central dendrite, and the signals corresponding to adjacent pixels of the images, which go through receptors, are braked by negative weighting coefficients of lateral dendrites.



Going into neuron the signals multiplied on the corresponding weighting coefficients are summarized and is moving on the output.

On the output of neural network the sequence of numbers takes the place, which according to definite law corresponds to the meanings of codes of colours of input points of the image.

If to figure the data received on an output in the form of schedule, in some places it is possible to note sharp differences of the obtained function, which responds to sharp gangs of colour. Having fixed these differences it is possible to compare their size with the given threshold. The excess of this size of the threshold testifies of availability of a contour of the image in the given point (fig. 8).

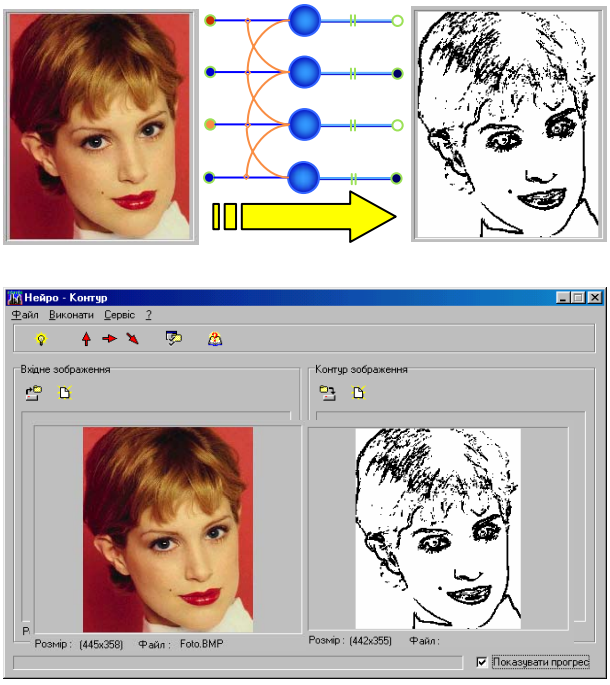


Figure 8. Contour of the image

Thus, the abstracting from colour of the image, illuminating intensity, and superfluous background "noise" is carried out. It allows to present object for classification more precisely, having deprived the image of elements, which do not bear any information load. And it also allows to reduce a volume of the information about object which is necessary for classification.

The problem of allocation of contours on the image has been solving by classic algorithms for a long time. They are: spatial differentiation, functional approximation, a high frequency filtration.

There is a tendency to esteem the border as the area of a sharp difference of a function of picture level f (x, y) that is general one for all these methods. But they are alike in their mathematical model of the notion edge and search algorithm of edge points [5-8].

There is one main disadvantage of all these algorithms; it a rather low speed of activity, which will not be enough for image processing of the large size at a high frequency of their reception. In the bionic approach, due to full parallelism of processing of each point of the image, at a hardware representation, the fast enough response time of processing of the input images is reached.

Internal representation of the image

For creation of the internal representation the image, which has passed prior processing, is braked into definite pieces, which are the characteristic peculiarities of the given face. The quantity of such pieces can be changed, selecting optimal.

Then the abstracting from the sizes of the entering image and seal of the obtained data of each piece is made, that allows to get rid of the unnecessary or unessential information. Then the obtained packed data are integrated in a so-called internal form of the image of the face.

The size of one such representation depends on quantity of fragments. The conducted researches have shown that internally the representation with optimal for further recognition by quantity of units takes no more than 25-100 bytes. So, for 100 000 images it is enough less than 2,4 - 100 Mbytes of a disk space.

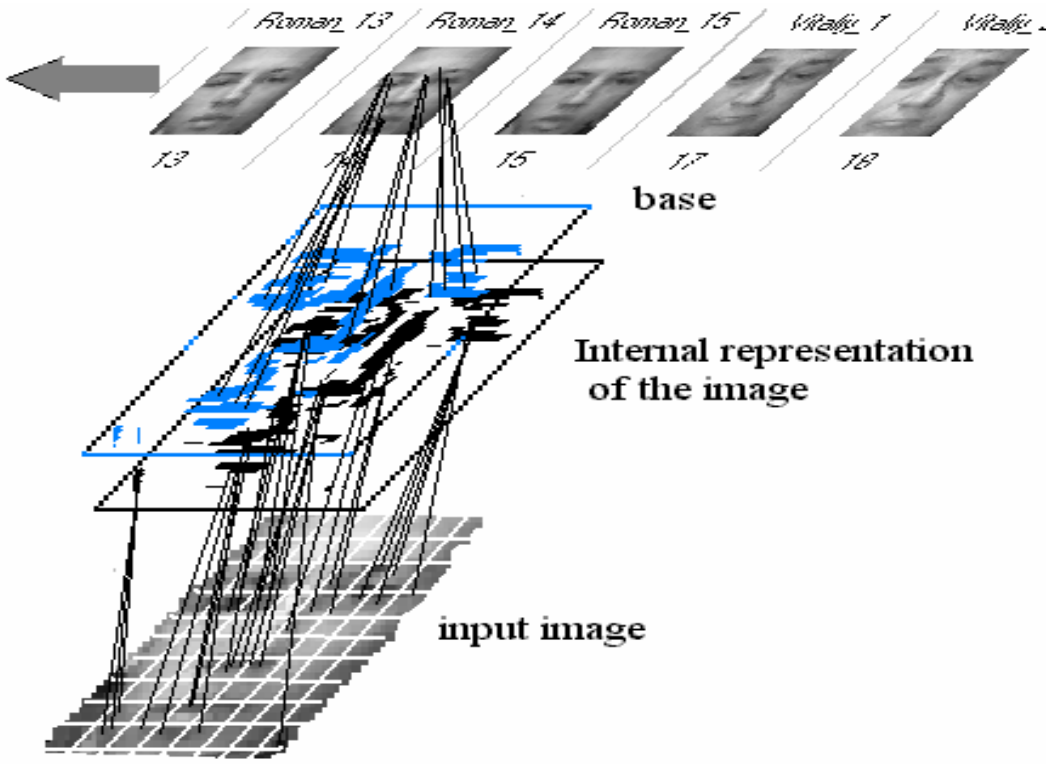


Figure 9. Factor of similarity.

Training and identification of the images in neural-like growing network

While training, internally the representation of the image containing the special features of the face is moved on receptors of neural-like growing network. Then, the excitation is transmitted on neural-like network elements, which correspond to the most similar faces. Further on, according to the rules of formation of neural-like growing network, the exited elements are integrated in one that corresponds to this representation.

Thus, there a case of learning, classification and accumulation of the obtained information process is observed. And the structure of neural-like growing network is built in such a way that the accumulation of any duality of the existing information is eliminated. At the stage of identification, on the input of neural-like growing network the internal representation of the identified image is moving. As a result, definite neural-like elements corresponding to the most eligible images that are accumulated in the network are excited.

According to the degree of excitation of elements the factor of similarity is determined, which in case of excess of the predetermined threshold value indicates a positive result of identification (fig.9).

At hardware representation of neural-like growing network, where the information went on each receptor is processed in parallel way, the classification descends immediately.

Figure 10. The program system

Program implementation

The program system (fig.10) is designed in Borland C ++ Builder 4.0 under platforms Windows 95/98/NT/2000.

System decides a problem of identification of the person according to the image of his face, having the database of the stored images. The images of faces on a program input are moved by means of the digital video camera, TV and also from the file of the beforehand recorded image or from a clipboard of Windows system. On the stage of prior processing, as it was described above, the problem of allocation of contours of the image is decided, using the bionic approach. The quality of further recognition fully depends on quality of the image preparation on this stage. So a flexible system of parameters selecting of allocation of contours is realized there.

Conclusions

One of the most complex problems, that faces specialists of the artificial intelligence, is a problem of perception of information and organization of intellectual system or robot behavior.

Considered here types of neural-like growing networks allow to the system, containing n-GN structure, in the process of its activity to form in it a model of external environment and work out adequate actions.

The neural-like growing networks give an opportunity to form the notions as the objects and relations among them when constructing the network itself. For this each sense (notion) gets a separate component of the network as an vertex connected with the other vertexes. In general, this fully corresponds to the structure reflect able in the brain, where each explicit notion has represented as a definite structure and has its denoting symbol. Thus, the neural-like growing network is represented as a convenient apparatus for simulating the mechanisms of purposeful thinking as performing the specific psychophysiological functions.

In comparison with known intellectual systems and robots, behaving according to the preprogrammed functions, the intellectual systems based on the new technology, provide possibility to generate own functions of behavior through the analysis of external information.

The usage of the bionic approach on the stage of prior processing of the image and the technology of data processing in neural-like growing networks allows to reduce considerably the volumes of computing operations. All operations made above the image on the stage of prior processing and also the classification and information storage about the images and their further identification are made only by the mechanisms of neural-like networks without the usage of complex algorithms, which demand a large volume of calculations. By the corresponding hardware support the neural network methods allow to increase considerably an effectiveness of the solution of the given class of problems, saving a high accuracy of result and high level of response, both in the mode of training, and in the mode of identification. The described method can be applied in security systems and different technical systems for recognition of current situations with the purpose of decision making and fulfillment of actions based on the processing of the visual information.

The technologies of recognition of the man are already submitted in the software market.

The company Miros has presented TrueFace ID system of recognition of the person of the man [3]. It is a new product Miros, constructed on the basis of technology TrueFace. The system TrueFace ID is constructed on the basis of the appropriate software, which on the reclame of the developer, allows to identify the person of the man in a database from thousand images in some seconds. TrueFace ID makes comparison of the images of the persons received directly with the help of a videocamera or written down on a videofilm, with stored(kept) in a database, and precisely identifies the man.

The corporation Visionics offers a means of recognition of under the name FaceIt PC [4], which is intended for amplification (strengthening) protection of independent PCs equipped OC Windows 95. The company DCS (Dialog Communication Systems) Inc. represents technology BioID, on the basis of which the identification of the person is made on three biometric parameters: the person, vote, movement of lips [5]. The testing of our system has shown results not making a concession of systems TrueFace ID and BioID. However, taking into account an orientation of new technology – n-GN on mass parallelism, the hardware realization of system will give significant advantages above systems TrueFace ID and BioID

References:

[1] V.A. Yashchenko Neural-like growing networks - new class of the neural networks // Proceedings of the International Conference on Neural Networks and Brain Proceeding, pages 455 -458, Beijing, China, Oct. 27-30' 98.

[2] V.A. Yashchenko Receptor-effector neural-like growing network - an efficient tool for building intelligence systems // Proceedings of the second internatiional conference on information fusion, July 6-8, 1999, Sunnyvale Hilton Inn, Sunnyvale, California, USA, Vol.11, pp. 1113-1118

[3] Technology TmeFace ID of the company Miros. - http: // www.miros.com, 2000

[4] Technology FaceIt of the company VISIONICS. - http: // www.faceit.com, 2000

[5] Technology BioID of the company DCS Inc.- http: // www.bioid.com, 2000

[6] Method main компонент.-http:// ww.mechmath. psu.ru, 2000

[7] P. Duda, item Харт Pattern recognition and analysis. With engl., under ed. V.L. Stefanuk m. The world, 1976E. Hall Computer Image Processing and Recognition - N.Y.: Academic Press, 1979

[8] K.S. Fu, J. Mu Pattern Recognition, 1981,v.l3.№l

Author information

Vitaliy Yashchenko - Institute of Mathematics of Machine and Systems; Kiev, Ukraine;

e-mail: mis@immsp.kiev.ua

Section 4: Intelligent Technologies

ВЫСОКООРГАНИЗОВАННАЯ СРЕДА КОРПОРАТИВНО - ВЗАИМОДЕЙСТВУЮЩИХ РАСПРЕДЕЛЕННЫХ ИНФОРМАЦИОННЫХ СИСТЕМ

Н. Алишов

Abstract: *Consider architecture of highly organized (high-order, higher-order) enterprise systems on base virtual network computers. Criterion for evaluation degrees of organization higher-order information system has been offered. On the basis of stratified enterprise network designed integrated model of highly organized systems.*

Ключевые слова: *высокоорганизованная система, корпоративная сеть, виртуальный сетевой компьютер, стратификация,*

Введение.

Эффективность обработки распределенных информационных ресурсов во многом определяется степенью организации средств управления взаимодействием системных и прикладных процессов. Если информатизацию рассматривать как процесс разработки и внедрения средств и технологий, обеспечивающих возможность пользователям получить доступ к необходимым и доступным информационным ресурсам во времени и в пространстве [1], то факторы время и пространство становятся базовыми показателями при создании соответствующих систем. Причем, эти показатели служат неявными критериями для оценки функциональных возможностей информационных систем. Например, если размещение распределенных информационных ресурсов и аппаратно – программных средств не будет обосновано научно – прикладными методами то, эффективность соответствующих систем обработки может свестись к некоторому минимуму, преодоление которого не может быть осуществлено никакими доработками в системе. Поэтому линейное расширение функциональных возможностей средств обработки информации, за счет разработки дополнительных прикладных систем, не в состоянии обеспечить современные потребности процесса информатизации в целом.

Анализ развитых информационных технологий показывает, что в отдельных случаях разрабатываются системы, обладающие универсальной полнотой, благодаря которой прикладные приложения отличаются высокой эффективностью за счет внутренней организации среды. В настоящее время повсеместное внедрение подобной концепции еще осложняется чисто прагматическими интересами ряда ведущих фирм, которые добились де-факто международного признания их проблемно–ориентированной продукции. Однако, мировая тенденция постепенного перехода к интеллектуальным сетям, а также необходимость максимальной инвариантности вновь создаваемых информационных средств и технологий относительно области применения, обусловливают разработки соответствующих концептуально – теоретических основ и архитектурных решений для создания систем с высокой степенью организации операционной среды.

В данной работе сделана попытка синтеза высокоорганизованной архитектуры операционной среды управления ресурсами в рамках интеллектуальных корпоративно – взаимодействующих сетей компьютеров.

Степень организации информационных систем.

Степень организованности информационных систем - это функционал Ψ, позволяющий апостериори оценить количественные и качественные характеристики информационных систем при решении класса задач, задаваемый лицом принимающего решения (ЛПР):

$$\Psi = F(V_\eta, Q),$$

где V_η и Q соответственно уровень, и объем обработки данных для решения множества задач S.

Пусть $V_\eta = \{x, \eta_v(x)\}$ расплывчатое множество качественных оценок решаемых задач; $\eta_v(x)$ - функция принадлежности множества V_η. Функцию $\eta_v(x)$ будем использовать в качестве критерия оценки уровня решаемых задач.

В простейшем случае

$$\Psi^i = \eta_v^i(x) * f^i(Q^i, I^i),$$

где функционал $f^i(Q^i, I^i)$ характеризует объем решаемых задач, I^i - информационная характеристика $i-$ ой задачи.

Если считать, что каждая задача s_j^i, в классе задач S^i имеет свой весовой коэффициент q_j^i, определяемый ЛПР ($1 \geq q_j^i \in Q^i$), с учетом

$\sum_{j=1}^{k_i} q_j^i = 1$, то величина $q_{jps}^i = -q_j^i * h_{jps}^i$, будет характеризовать значимость полученной информации в результате решения задачи $s_j^i \in S^i$.

Таким образом, обобщая вышесказанное (безотносительно класса решаемых задач) приходим к выводу, что степень организованности информационных систем определяется формулой

$$\Psi = \eta_v(x) \sum_{j=1}^{n_I} \eta_v^j \; q^j \frac{I_{jps}}{I_{jpr}},$$

где I_{ipr} - возможное максимальное количество информации, которое может быть получено в результате решения i –ой задачи, I_{ips} - реальное количество информации, полученное в результате решения i –ой задачи.

Единица измерения величины $0 \leq \Psi \leq 1$ - int.

$$1\,\text{int} \equiv [\eta_v(x) = 1, \ \forall x \in X] \wedge [\eta_v^i(x) = 1, \ \forall x \in X] \wedge [\frac{I_{ips}}{I_{ipr}} = 1, \quad \forall i = \overline{1,n}] \wedge [\sum_{i=1}^{n} q_i = 1].$$

Это означает, что информационная система решает все задачи множества S^i, причем в результате полностью снимется априорная энтропия H_{pr}, каждая задача решается наилучшим образом с точки зрения ЛПР и процесс получения результатов при решении всех задач в комплексе также выполняется наилучшим образом. Кроме того, обеспечивается целостность множества задач т.е. $\sum_{i=1}^{n} q_i = 1$.

Формализованная модель корпоративно–взаимодействующих информационных систем.

Производственные информационные ресурсы, обеспечивающие деятельность крупных организаций и предприятий, имеющих развитые инфраструктуры организационного управления, являются, как правило, корпоративными (взаимоопределяющими и взаимодействующими). Компьютерные сети, поостренные для информатизации деятельности подобных организаций, называются корпоративными сетями (Enterprise Networks) и характеризируется следующими особенностями.

Во-первых, это большое количество объединяемых в общую сеть компьютеров, в том числе - файловых серверов, серверов баз данных и приложений.

Во-вторых, гетерогенный характер сети: различные протоколы, разнородные среды передачи данных, компьютерные платформы и средства коммутации, произведенные разными компаниями, различные операционные системы и т.п.

В-третьих, корпоративность сети полагает, что функциональные задачи отдельных подсистем могут быть существенно различными, хотя в целом сеть ориентируется на решение единой задачи крупной системы организационного управления.

Наконец, корпоративные сети характеризуются, как правило, наличием многих производственных площадок, распределенных в определенном региональном масштабе.

Наряду с этим корпоративные сети должны обладать расширяемостью и масштабируемостью, что однозначно определяется функциональными возможностями выбранных сетевых аппаратно - программных средств.

Модель корпоративной сети может представляться различными стратами: *структурной, протокольной, операционной, прикладной и физической.* Модель ***структурной стратификации*** представляется в виде симметричного ориентированного графа $G = (X, A)$. В общем случае вершинами $X = \{x_1 ... x_m\}$ графа G могут быть двух типов:

- Обрабатывающие – информационные (ОИ);
- обрабатывающие – коммуникационные (ОК).

ОК вершины могут быть трех типов:

- одинарные (один вход один выход);
- мультиплексные (один вход много выходов);
- коммутирующие (много входов много выходов).

ОК вершины характеризируются коэффициентом производительности

$P = \frac{K_П}{T}$, где $K_П$ - количество пакетов транлируемых ОК вершиной.

Ребры $A = \{a_1 ... a_n\}$ графа $G = (X, A)$ характеризуется коэффициентом проводимостью $П_P = \begin{Bmatrix} \le R \ при \ S = 1; \\ 0 \quad при \ S = 0 \end{Bmatrix}$, где R - пропускная способность ребра, S - булевая переменная, показывающая состояния ребра.

ОПРЕДЕЛЕНИЕ 1. Разделяемым графом называется логически полный симметрично ориентированный граф, в котором в промежутке времени ΔT, называемым “окном” связи могут быть смежными только две ОК вершины. Иными словами в промежутке времени ΔT для $(\forall a_i)_{i=\overline{1,j-1}} \wedge (\forall a_i)_{i=\overline{(j+1),n}} \wedge a_j$ выполняются условия $\left(\forall\left(i = \overline{1, j-1}\right)\right) \wedge \left(\forall\left(i = \overline{(j+1), n}\right)\ S_i = 0\right) \wedge \left(\forall\left(j = \overline{1, n}\right)\ S_j = 1\right)$, где при $j = 1, i = \overline{2, n};\ a \ при \ j = n,\ i = \overline{1, (n-1)}$.

ОПРЕДЕЛЕНИЕ 2. Кусочно – разделяемым графом называется логический полный симметрично ориентированый граф, в котором в промежутке времени ΔT, называемым “окном” связи могут быть смежными v пар ОК вершин. Иными словами, в промежутке времени ΔT для v дуг остовного подграфа $G_v = (X, A_v)$ и для p дуг остовного подграфа $G_p = (X, A_p)$ выполняются условия $(\forall(S_i)_{i=\overline{1,v}},\ S_i = 1) \wedge \left(\forall(S_i)_{i=\overline{1,p}},\ S_i = 0\right)$, где $p = n - v$ и $v \le n/2$.

ЛЕММА 1. Степень организованности $\Psi^{кр}$ кусочно – разделяемого графа $G_v = (X, A_v)$ при решении задачи $S \equiv$ ” обеспечить производительность ребра $P_p^* = P_p$ ” определяется следующимм соотношением

$\Psi^{кр} = \begin{Bmatrix} 1,\ при\ i = \overline{1,v} \\ n/2 * \eta_v,\ при\ i > v \end{Bmatrix}$, где η_v функция принадлежности

разделяемого графа при решениии той же задачи.

Доказательство. Логический полный граф $G = (X, A)$ содержит $m(m-1)/2$ ребер. При этом $\max v = m/2$. В интервале ΔT максимально достижимая производительность в разедляемом графе $P_p = P$. Функция принадлежности η_v =1 только для одной пары $(ОК_i, ОК_j)_{\Delta T}$. Функция принадлежности $\eta_м^{кр} = 1,\ при\ (i = \overline{1,v})_{\Delta T}$. При v>v, $\eta_м^{кр} = m/2\,\eta_v$. Так как при решении данной задачи весовой коэффициент q=1, то $\Psi^{кр} = \eta_м^{кр} q^{кр}\, I_p^{кр} / I_{pr}^{rh} = (m/2)\eta_v\, I_p^{кр} / I_{pr}^{кр}$.

С целью ***аппаратной стратификации*** система представляется в виде совокупности взаимодействующих пассивных и активных компонент. Пассивными компонентами является все те составляющие, которые не поддаются программному управлению (линии связи, каналообразующие оборудования, повторители и т.п.). Однако, такие компоненты системы как физические интерфейсы, разъемы, порты и др. должны быть отнесены условно пассивным компонентам, так как их характеристики являются предопределяющими с точки зрения расширяемости и масштабируемости системы. Активным компонентам относятся все программируемые модули, в том числе те, которые могут быть использованы для удаленного доступа. В свою очередь активные компоненты делятся на три категории:

1.сетенезависымые модули (ОИ)

2.сетезависымые модули (ОК)

3.общесетевые ресурсы.

На уровне аппаратной стратификации степень организованности распределенной системы должна быть оценена, кроме того, с учетом расширяемости и масштабирумости системы. При этом базовый критерий применяется с учетом таких показателей системы как «эффективность по стоимости» и «возможность защиты вложенных инвестиций». В докладе доказывается теорема о существовании проектных решений, обеспечивающих максимального значения критерия оценки степени организованности аппаратных средств при средних значениях этих показателей. Причем учитывается параметры структурной стратификации системы.

Протокольная стратификация предусматривает формальное описание двухмерных и трехмерных правил взаимодействия одноименных слоев распределенной системы. Наилучшим способом формализации является способ, предложенный международной организацией по стандартизации в части эталонной модели взаимодействия открытых систем [2].

Для простоты некоторый произвольно выбранной уровень обозначим как (N) – уровень, а смежные с ним верхние и нижние уровни - соответственно (N+1) - уровень и (N-1) уровень. Суть уровневой организации сводится к тому, что каждый уровень повышает потребительную стоимость услуг (путем расширения их количества или повышения качества), обеспечиваемых всеми нижерасположенными уровнями, таким образом, чтобы наивысший уровень быль обеспечен полным набором услуг, необходимых для функционирования распределенных прикладных процессов. Тем самым уровневая организация позволяет разложить сложные задачи на небольшие части. Другой основной принцип уровневой организации состоит в обеспечении взаимонезависимости всех уровней путем определения услуг, которые могут предоставляться тем или иным уровнем смежному с ним верхнему уровню, безотносительно к конкретному способу их реализации. Такой подход позволяет вносить изменения в способ функционирования одного или нескольких уровней при сохранении неизменным набора услуг, предоставляемых каждым таким уровнем смежному с ним верхнему уровню.

Каждый уровень, за исключением самого верхнего, обеспечивает услуги для смежного с ним верхнего уровня. Услуга – это функциональные возможности (N) уровня, которые предоставляется в распоряжении (N+1) компонентам. Следует, однако, иметь в виду. Что к понятию услуг относятся не все функции, выполняемые внутри (N) уровня, а только те из них, которые могут использоваться смежным верхним уровнем. (N) компоненты, распределенные по взаимосвязанным открытым системам, действуя совместно, предоставляет (N) услуги (N+1) компонентам. Другими словами, (N) компоненты повышают

потребительную стоимость (N-1) услуги, получаемой из (N-1) уровня, предоставляя затем эту дополнительную услугу, т. е. (N) услугу, (N+1) компоненте.

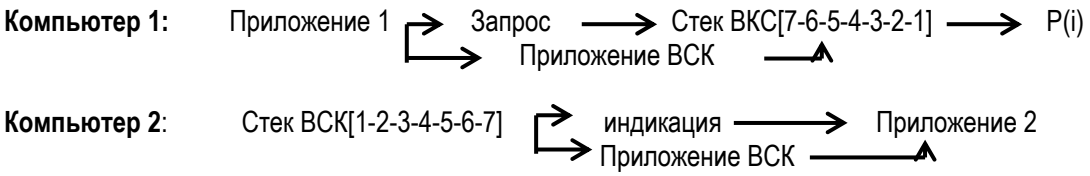
(N) услуги предоставляются (N+1) компонентам в (N) точках доступа к услугам ((N) - ТДУ), которые играют роль логических интерфейсов между (N) компонентами и (N+1) компонентами. (N+1) компонента обменивается с (N) компонентой той же системы через (N) – ТДУ. Услуги в (N) – ТДУ могут поступать только из одной (N) – компоненты и выдаваться из нее только для одной (N+1) компоненты. В то же время одна (N) компонента может выдавать свои услуги через несколько (N) – ТДУ.

Подобное представление протокольной стратификации распределенных систем с целью оценки степени их организованности позволяет оптимизировать количественные и качественные характеристики услуг предоставляемых отдельными уровнями. Причем инвариантность протокольных услуг относительно аппаратной стратификации систем обусловливает оптимизирующие требования ко всем остальным стратам распределенных систем.

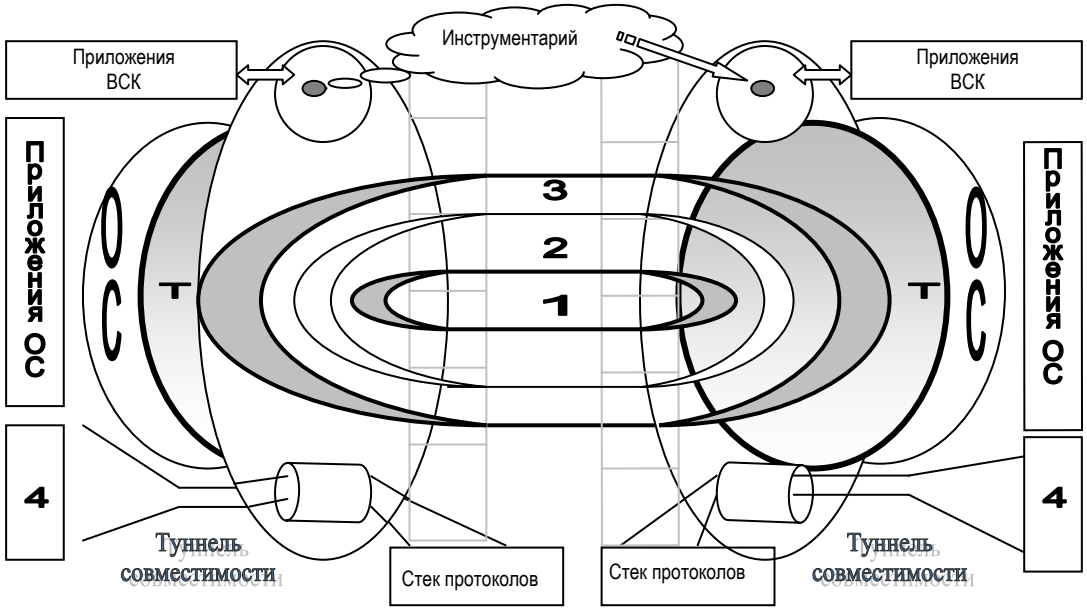
Особое значение для высокоорганизованных систем предоставляет ***стратификация системы на уровне операционной среды*** функционирования всех остальных страт. Исследование проблем, возникающих при системной интеграции множества сетевых и компьютерных технологий, показывает, что эти проблемы полностью можно было бы устранить, если бы сетевая архитектура была реализована на основе концепции виртуального сетевого компьютера (ВСК). ВСК представляет собой модель некоего компьютера, которая имеет унифицированную систему команд, сетевую операционную систему, командный язык ОС, язык программирования, языки скриптов и директив, протоколы передачи данных и т.п. (см. рисунок). Кроме того, ВСК должен включать в себя возможность расширения командных и языковых средств.

Прообразом подобной системы организации взаимодействия процессов является JAVA машина, которая реализуется во всех современных сетевых технологиях. В общем виде ВКС – это JAVA машина с расширенными сетевыми возможностями, установленная не как приложение в рамках операционной среды, а как фронтальный сетевой препроцессор, осуществляющий все без исключения сетевые взаимодействия.

В общем случае, Internetworking и Interoperability через ВСК реализуется по схеме приведенной ниже.



ВСК в качестве сетевого фронтального процессора предоставляет возможность реализовать как сетевую, так и распределенную операционную систему. В общем случае между этими операционными системами имеются принципиальные различия. Задачей сетевой операционной системы является, как правило, обеспечение взаимодействия между приложениями посредством протокольных преобразований. Распределенная операционная система же, кроме того, позволяет управлять сетевыми ресурсами в рамках функционально однородной операционной среды. Этим достигается возможность создания интеллектуальных сетевых приложений, в том числе сетевых вычислений за счет внутренней организации системы, а не за счет обеспечения «взаимопонимания» между множеством прикладных систем. Иными словами распределенная операционная система, обладая универсальной полнотой, позволяет реализовать бесчисленное множество приложений посредством унифицированного программного интерфейса взаимодействия приложений. Являясь архитектурой для будущих высокоорганизованных систем, отдельные подсистемы распределенной среды уже сегодня находят практическое воплощение в таких областях, как кластерные системы, распознавание образов, интеграция мультимедийных систем, распараллеливание обработки информации и т.п. Технология ВСК может стать идеальной базой для реализации распределенной операционной сетевой среды в интеллектуальных сетях.



1 – система команд ВСК; 2 - виртуальное драйверное поле ВСК;
3 – распределенная операционная среда ВСК; 4 – приложения ранних разработок.

Взаимодействие приложений с ВСК может быть осуществлено тремя способами.

1. Приложение, реализованное в резидентной (базовой) операционной среде транслируется в операционную среду ВСК, который через унифицированный стек протоколов устанавливает все сетевые соединения с удаленными ВСК. Все запросы приложения перетранслируются в соответствующие запросы резидентной операционной среды удаленного компьютера. Результаты передаются запросившему их приложению в обратном порядке.
2. Все сетевые приложения, разработанные без учета наличия ВСК взаимодействует между собой в сети по классической схеме через предусмотренный для этого туннель совместимости, который содержит все необходимые стеки протоколов и сетевые утилиты.
3. Так как ВСК является полноценным логическим компьютером со своей распределенной операционной системой, то все сетевые приложения могут быть разработаны непосредственно на инструментариях ВСК. В этом случае сетевое взаимодействие приложений осуществляется без каких либо преобразований.

Последний способ с одной стороны, повышает эффективную производительность системы, с другой - позволяет разработать полноценные сетевые приложения с учетом всех особенностей корпоративной сети. В принципе локальные приложения также могут быть реализованы в операционной среде ВСК. Однако эффективность подобных приложений не может быть гарантирована. Следует обратить внимание на то, что возможности распределенной операционной системы ВСК позволяют при необходимости рассматривать все ресурсы корпоративной сети как единый виртуальный суперкомпьютер и разрабатывать соответствующие приложения с использованием всех современных технологий. Подобная необходимость может появиться, например, при распределенном имитационном моделировании (мыслительных процессов, сверхсложных объектов, макроэкономических процессов и т.п.).

Высокоорганизованная система функционирует по схеме ЗАПРОС – ОТВЕТ.

Причем, источником ЗАПРОСа может быть как субъект, так и объект; ОТВЕТ может быть предоставлен как объекту, так и субъекту. ЗАПРОС обрабатывается иерархической схемой:

ПРОБЛЕМЫ – ЗАДАЧИ – АЛГОРИТМЫ – ОБЪЕКТЫ – ПРОЦЕДУРЫ.

На каждом иерархическом уровне выбор вариантов может быть как однозначным (безальтернативным), так и многозначным (альтернативным), в зависимости от содержания ЗАПРОСа и состояние системы (СС). Это означает, что, например, АЛГОРИТМ решения задачи в системе может иметь множество вариантов. Причем эти варианты являются производными от основной версии АЛГОРИТМа. и, являются прерогативой самой системы (в докладе рассматривается технология подобной организации). Объединяя моделей выбора альтернатив в соответствии с иерархической структурой обработки ЗАПРОСА, получаем обобщенную модель высокоорганизованной системы.

Литература

[Алишов Н.] Н.И. Алишов. Базовые технологии системной интеграции в интеллектуальных корпоративных сетях. Управляющие машины и системы. – 2000.- № 5/6. – с. 25-35.

[John, 1983] John D. Day and Hubert Zimmerman. The OSI Reference Model. Proceedings IEEE .- V.71. – number 12. – December, 1983. –pp. 8-16.

Сведение об авторе

Алишов Надир – Институт кибернетики им. В.М. Глушкова НАН Украины, г. Киев –03187, пр. Глушкова 40. тел.: (044) 266-34-27, mailto: anio@i.com.ua.

PROCESSING OF KNOWLEDGE ABOUT OPTIMIZATION OF CLASSICAL OPTIMIZING TRANSFORMATIONS

Irene L. Artemjeva, Margarita A. Knyazeva , Oleg A. Kupnevich

Abstract: *The article describes the structure of an ontology model for Optimization of a sequential program. The components of an intellectual modeling system for program optimization are described. The functions of the intellectual modeling system are defined.*

Keywords: *Knowledge based system; Program optimization; Domain ontology*

Developing knowledge-based systems for any domain needs constructing its ontology [Kleshchev, 2002]. Ontology is an explicit description of domain notions and contains terms for describing reality and knowledge and agreements restricting the interpretations of these terms. The ontology of a domain defines the structure of knowledge and the structure of domain reality.

The problems in the "Program optimization" domain are mainly grouping around the unification of the notion system for describing program schemes in terms of which one could describe optimizing transformations and standardization of notion system for describing optimizing transformations. The other set of problems has to do with how to effectively use the accumulated knowledge in the problematic area, i.e. is connected with the special training in program optimization, the development of skills of putting theoretical knowledge about program optimization into practice.

The ontology model of the knowledge domain "Sequential program optimization" and its using when developing computer knowledge banks on program optimization is presented in this work.

The ontology model of the "sequential program optimization" domain

The formal description of the terminology of a knowledge domain together with definition of meanings of terms is called "ontology model" [Kleshchev, 2001]. The terms of the knowledge domain "Program optimization (classical optimizing transformations)" can be divided into two groups: (i) the terms for describing programs (the terms of this group will be called the terms for describing the optimization objects), and (ii) the terms for describing the optimization process. Therefore the ontology model of this knowledge also consists of two parts.

The optimization object is a program. The characteristics of a program are always formulated in terms of a mathematical model of this program. The characteristic of a program is the language (a set of programs) this program belongs to. The characteristics of the language are also formulated in terms of a mathematical model of

this language. Thus, a number of terms for describing the object of optimization can be divided into two groups: (i) the terms for describing the language model, (ii) the terms for describing the program model. Before optimizing a program, the language this program belongs to must be determined. Consequently, the terms for describing the language model are parameters of the ontology model, and the terms for describing the program model are the unknowns of the ontology model. Then the language model is represented by the values of the parameters, and the program model – by the values of the unknowns.

It is evident that the program model describes not one program but a set of programs that have the same characteristics; the language model describes a set of languages that have the same characteristics. Therefore the following requirements are set on the language model: it must allow to present basic constructs of imperative programming languages essential for describing sequential OTs; it must be flexible and extensible in order to expand the class of modeled programs, if necessary; the form of presenting program models must be convenient for analyzing both information flows and control flows in the program.

The ontology model of the knowledge domain "Sequential program optimization" consists of two modules. The first module contains the terms for describing the optimization object, the second one – the terms for describing the optimization process.

The first module is an unenriched system of logical relationship with parameters written in sentences of a many-sorted language of the applied logic. Any program consists of fragments that in their turn consist of other fragments [Kasyanov, 1988] [Knyazeva, 1999], i.e. any program has its syntactic structure that is reflected by a mathematical model of this program.

Each fragment – as an element of the program – has a number of characteristics. First of all, it has *the address of the fragment* – the unique characteristic unambiguously defining each fragment in the program.

All the fragments can be divided into three groups: declarative statements, imperative statements, and entries of statements. Its class characterizes each fragment, e.g. the fragment can have the class of assignment statement, iteration statement, declarative statements of functions, etc. The function *FragClass* returns the class of this fragment for the indicated address of the fragment. A set of names of fragment classes (of each group) in the program assigns the values of the parameters D*eclarative statements, Imperative statements, Entries of statements.*

Control arcs assign the syntactic structure of the program. Each control arc connects two fragments of the program. Each control arc has its label identifying a type of connection between the fragments. The function the name of which coincides with the arc label is used to assign the control arc. The value of the parameter *Names of control arcs* assigns what labels can be owned by control arcs in the program. The control arc area is defined for each control arc. This area is assigned by the value of the functional parameter *Control arc area* – a function that matches each arc label with a pair consisting of two sets of names of fragment classes: the first set determines what fragments classes can be arguments, the second one determines what fragments classes can be results of the control arc with this label.

There are always a number of various *Identifiers* in the program. Identifiers can be of different types, e.g. identifiers of variable, functions, constants, and data types. In the program identifiers of each type make a set the name of which coincides with the name of the type. The value of the parameter *Types of identifiers* assigns what types of identifiers can be present in the program.

Functions and relationships identifying the structure of the program and some of its characteristics are defined on a set of fragments and identifiers of the program.

Functions with one argument (a fragment or an identifier) are called attributes. Each attribute has its name. The value of the parameter *Names of attributes* assign what attributes can be used for describing the characteristics of identifiers or fragments in the program. The definitional domain and the range of attribute values are established for each attribute. They are assigned by the values of the functional parameters *Definitional domain* and *Range of attribute value.*

The value of the parameter *Names of functions* assigns what functions with two or more arguments can be used for describing the characteristics of fragments or identifiers of the program. The definitional domain and the range of values are established for each function They are assigned by the values of the parameters *Definitional function domain* and *Range of function values.*

The value of the parameter *Names of relationships* assigns what relationships can be defined on a set of fragments and identifiers of the program. The values of the parameters *Determination of Relationship* for each relationship name assign a formula of truth determination of a relationship between fragments and identifiers of the program.

Correctness is another characteristic of the fragment. The value of correctness is assigned by the predicate *Correctness* that matches the address of the fragment with the truth if it has all the control arcs and attributes necessary for this fragment. The value of the parameter *Determination of correctness* assigns control arcs and attributes for each class of fragments.

The value of the parameter *Elementary types* assigns for the language a set of identifiers of data types that are basic for all the rest data types of this language.

The value of the parameter *Modes of generation* assigns a set of names of constructors of data types in the language. The values of such characteristics as a base type and the method of constructing a type from the base one are set for each identifier of data type in the program. The value of the first characteristic is assigned by the function *BaseType* that matches each identifier of the type with a chain of identifiers of types that are used when constructing this data type. The value of the second characteristic is assigned by the function *ConstructMethod* that matches each identifier of the type with the name of the constructor. The value of the parameter *Compatibility of types* is a predicate defining the possibility of implicit transformation of one type to another.

The value of the parameter *Reserved names* assigns a set of names of roles that can be played by fragments or identifiers of the program. The value of the parameter *Area of reserved name value* matches each name of a role with a predicate defining proprieties for a fragment or an identifier playing this role in the program.

The value of the parameter *Operators* assigns a set of symbols used in the program for identifying operations in expressions (arithmetic, logical, of a transformation, etc.) of the program. The functional parameters *Arity* and *Priority* match each symbol of the operation with a number of arguments and priority.

Optimization is understood as a chain of steps at each of them one transformation is applied to an optimized program. Each step will be called a step of the optimization history. The program model written in terms of the language model is the optimization object. The program model changes at each step of optimization: the rule of transformation that is established by the optimizing transformation that is used at the current step of optimization is applied to it. Thus, at each step of optimization there is its own version of the program model with its set of fragment addresses, its set of identifiers, etc. Therefore all the terms defined in the work [Artemjeva, 2002] are functions the first argument of which unambiguously identifies the current program model (i.e. the current set of fragment addresses, the current set of identifiers, etc.). A number of a step of the optimization history plays a role of this argument.

Let us define the terms for describing optimization process. Each optimizing transformation has the following characteristics: the saving block, the context condition, the indicative function, and the application strategy.

Normally an optimizing transformation (OT) is not applied to the whole program but to one of its blocks not necessarily continuous. The set of program fragments necessary for making a decision about optimization will be called candidate for saving blocks and the set of fragments where the context condition is true will be called saving block. Thus the saving block is a candidate for which the context condition of an optimizing transformation is true.

An ordinary saving block is a fixed set of program fragments for which the number of fragments, their types and their mutual location in the program are known. However the saving block consists of two parts. The first part is a tuple of fragments where each element's type and location in the saving block are known.

The second part is a set of tuples of fragments. The number of tuples belonging to the second part of the saving block is variable for different saving blocks of one program but the number of elements of each tuple, types and mutual location of fragments included in the chain are fixed, i.e. these tuples of fragments own certain identical characteristics. When applying an optimizing transformation, all the tuples of fragments of the second part of the saving block change the same way. For example, the saving block contains both declarative statement of a procedure and all its invocations for certain optimizing transformations of procedures (functions). The declarative statement of a procedure is always single but invocation operators can be numerous. All the elements of the set of invocations are assigned by declarative statement of a procedure. Thus, after applying an optimizing transformation, all operators of invocation of a procedure (function) change the same way.

The saving block in the model is represented as a pair the first element of which will be called the simple part of the saving block, the second one will be called the multiple part of the saving block. The first element of the pair is a tuple of addresses of fragments. The second element of the pair is a set of tuples of addresses of fragments. For the example given above, fragments of declarative statement of a function make the simple part of the saving block and a set of tuples of fragments of invocation operators makes the multiple part of the saving block.

Each optimizing transformation has simple and multiple parameters. Each fragment of the simple part of the saving block will be called the value of the simple parameter of the saving block corresponding to it.

The set of fragments included in the set of tuples of fragments of the multiple part of the saving block will be called the value of multiple parameter of the saving block. The function *Number of simple parameters of OT* assigns the number of elements of the tuple of the simple part of the saving block. The function *Number of multiple parameters of OT* assigns the number of elements of each chain of the multiple part of the saving block. The functions *Classes of simple parameters of OT* and *Classes of multiple parameters of OT* assign chains of classes of fragments forming the simple part of the saving block and chains of classes of fragments forming each tuple of the multiple part of he saving block.

The context condition of the optimizing transformation describes the characteristics of the saving block this optimizing transformation is applied to. In this ontology model the context condition is presented as a predicate the arguments of which are the number of a step of optimization history and a candidate to saving blocks.

There can be several saving blocks in each step of the optimization process. Therefore it is necessary to have a criterion to choose one block form many. *Indicative function* (IF) is a formula the arguments of which are fragments from the saving block. It matches each saving block with its estimate – the rational number.

Optimization strategy is a formula helping to choose one number from the set of rational numbers – results of indicative function for various saving blocks. Correspondingly the saving block with the chosen characteristic undergoes optimization on this step of history.

Parameters with the identical context are a set of pairs of numbers of parameters of the saving block before and after optimization contexts of which must coincide.

The chain of application of OT assigns the order of OT application. Each OT is applied until there are valid saving blocks left for it.

All the terms defined above are considered as terms for describing knowledge of program optimization, whereas optimization strategy, indicative function and chain of application of OT allow to assign parameters of optimization; context conditions and transformation formulas assign statistic knowledge about optimizing transformations.

Let us further define terms for describing situations of the knowledge domain. These terms are mainly meant for complete recording of the modeling process of application of optimizing transformations to the program.

The situation consists of a chain of optimization steps (history steps). Sets of fragment addresses, identifiers, values of all attributes, functions and relations in the program are defined for each step. Some fragments, attributes, arcs and identifiers are defined while constructing the program model and known on the first step of optimization but some are recomputed before the beginning of each next step with the help of a special function *Enrichment of SMP*. The work of this function leads to that all the DSCH class fragments are added to the program model, Begin, End, Parent, SCH arcs are defined for them, all relations and functions are defined, values of all the attributes from a set of *Computable* attributes are also computed.

At each history step there is *a chain of current OT*. At the beginning of optimization a chain of *Current OTs* coincides with a chain of OT application.

The first OT from a chain of current ones for which a set of candidates to the SB is not empty is called *applied OT* and its number in the application chain is written in *Number of Applied OT*. A set of estimates is formed for all candidates to SB for applied OT with the help of indicative function and one estimate is selected from this set and with the help of optimization strategy and becomes the *characteristic of chosen SB*. The saving block corresponding to this characteristic becomes *chosen SB*. Apart from this, *optimized saving block* is defined on each step – it is a combination of fragments from next history step that becomes true when permutated to the transformation formula together with chosen SB. Optimized block appears as a result of OT application to the chosen saving block.

As a result of an applied optimizing transformation a number of fragments of the source saving block can be deleted, a number of fragments can be changed, new fragments (with new addresses)can be added, existing identifiers can be deleted or new ones can be added. For each optimizing transformation it is known how many elements will be included in the simple part of optimized saving block and how many elements will be included in each element of the multiple part of the saving block. This information is assigned by functions *Number of elements of the simple part of optimized SB* and *Number of elements of multiple part of optimized SB.*

Transformation formula is a predicate the arguments of which are two saving blocks from two consequent history steps; this predicate is true if the second SB is the result of OT application to the first SB.

Tasks given in terms of ontology model

As it follows from the previous section, in the ontology model there are groups of parameters defining Programming Language (PL), Optimizing Transformation Description Language (OTDL), Optimizing Transformations (OT); Optimization Strategy and groups of unknowns defining program characteristics before optimization, complete protocol of optimization process and characteristics of optimized program. Besides it is necessary to enter parameter Estimating function. This function will allow to compare various programs and, thus, to estimate optimization results.

Let us mention main classes of tasks that can be specified in terms of ontology model:

1. given PL, OTDL, an optimizing transformation and a program, required to get an optimized program and check the correctness of OT application;
2. given a program, PL and OTDL, strategy, estimating function, required to get an optimized program, estimate its optimality, check the correctness of OT, Strategy, analyze protocol;
3. given a set of programs, PL and OTDL, strategy and estimating function, required to get a set of optimized programs and estimates of their optimality, study the dependence of program optimality estimate on its characteristics;
4. given a set of programs, PL and OTDL, a set of strategies, estimating function, required to get a set of optimized programs and their optimality estimates, study the dependence of program optimality estimate on its characteristics and applied strategy;
5. given a set of programs, PL and OTDL, a set of strategies and estimating functions, required to get a set of optimized programs and their optimality estimates, study the dependence of program optimality estimate on its characteristics and applied strategy for various estimating functions;
6. given a set of programs, PL and OTDL, a set of strategies, an optimization criterion, required to get a set of optimized programs and their optimality estimates, study the dependence of program optimality estimate on its characteristics, applied strategy and programming language.

From the list given above one can see that all tasks are special cases of one task: given a set of PL, OTDL, input programs on these PL, and also a set of optimizing transformations, strategies of their application and functions – optimality estimates, required to build a set of optimized programs, protocols of their optimization and a set of optimality estimates for all programs on each step of optimization.

Method of solving the task of modelling optimization process

It is obvious that solving any task from the given above comes to solving the following task: with PL, OTDL fixed, the only input program, a set of OT, strategy of their application and function – optimality estimate are defined, required to build an optimized program, get the protocol of the optimization process and optimality estimate.

The algorithm to solve the task is given below.

```
BEGIN
Step of history=1;
Current OT (Step of history)= Chain of OT application;
Number of applied OT(Step of history)=1
Last step=False
REPEAT
    First OT application=True
```

To analyze SMP(Step of history)
REPEAT
IF Not First OT application
THEN Number of applied OT(Step of history)= Number of applied OT(Step of history)+1
Applied OT= Chain of OT application[Number of applied OT (Step of history)]
Candidates to SB(Step of history)=**To find saving blocks(Step of history, Applied OT)**
First application **=False;**
UNTIL (Candidates to SB(Step of history)≠∅) or
(Number of applied OT (Step history)=Length(Chain of OT application))
IF Candidates to SB(Step of history)≠∅
THEN
FOR SB in Candidates to SB(Step of history)
DO SB characteristic(Step of history, SB) =
To build SB characteristics (Indicative function(Step of history, SB))
Characteristic of chosen SB(Step of history)=**To realize Strategy** (Strategy of OT(characteristic of SB(Step of history)))
Chosen SB=**To choose SB** (Step of history, Characteristic of chosen SB(Step of history))
Optimized SB(Step of history)=**To build optimized SB**(Step of history, Chosen SB, Applied OT)
To build new SMP(Step of history+1), where exists the only Optimized SB (Step of history) where Transformation Formula (Step of history, chosen SB(Step of history), Step of history+1, Optimized SB(Step of history)), and the rest context coincides.
Step of history= Step of history+1
ELSE Last step=True
UNTIL Last step=True
Number of Optimization Steps= Step of history
END.

This algorithm is simple and obvious enough to serve as a kernel of instrumental system for program optimization. However, the functions applied in it: **To analyze SMP, To find saving blocks, to build characteristics of SB, To realize strategy, To build optimized SB and To build new SMP** are not that obvious and can be considered as separate subtasks.

The structure of intelligent system on program optimization

The developed ontology, the tasks given in it and proposed methods of solving them provide the basis for the instrumental system for program optimization. Instrumental modeling expert system of program optimization (I_MESPO) is intended to support teaching of classical optimizing transformations.

This system allows the user to describe optimizing transformations, to set their application conditions and transformation rules, to form various sets of optimizing transformations, to assign chains, to trace the program optimization history.

The input data of the system are optimizing transformations knowledge, testing program on an algorithmic high-level language.

The result of the work of the system is the protocol of the optimization history protocol where for each optimization step it is shown what transformation has been applied, what saving blocks have been found on this step, which block has been chosen and what it has been replaced by.

Since this task is connected with the complicated processing of the knowledge given and generation of new one, the given subsystem was done as an expert system that models program optimization process.

The expert system includes a subsystem of visual input of knowledge about program optimization, knowledge base description language translator (Synthesizer), high-level language translator in the model of structured programs, estimating and result visualizing subsystem, integrated shell providing interface among these subsystems.

The work with the system I_MESPO begins with that the researcher defines the system of optimizing transformations, i.e.: assigns a list of OTs, a chain of their applications, and defines context conditions for each OT, transformation formulas, indicative functions and optimization strategies. After assigning the values of these

parameters, Synthesizer executes translating knowledge base about program optimization into implemented module, thus creating application expert system (AES).

The functioning of the created application expert system begins with the work of a translator included in the system that transforms the structured program written in a high-level language into the structured program model (SPM). SPM is an internal form of relational presentation of programs that is convenient for analyzing and optimizing. According to this model, the application expert system makes an inference and builds up the optimization history protocol of this program. After receiving this protocol, the estimate and visualizing subsystem allows to estimate the program optimality before and after optimizing with the help of assigned estimating function and to compare the texts of the programs on each optimization step on high-level language and to analyze the implemented changes.

Conclusion and Acknowledgements:

Knowledge processing in the field of program optimization makes it possible to use this knowledge in industry, science and education.

Describing optimizing transformations within the terms of one model must facilitate the unification of different transformations within the framework of one system of OT. Thus, specialists can spend much less effort to study and use optimizing transformations to solve program optimization problems.

The use of the knowledge gives an opportunity to train highly qualified specialists to solve tasks on program optimization.

The access to knowledge through the Internet will attract all specialists interested in knowledge exchange on this problem.

References:

[Artemjeva, 2002] Artemjeva I.L., Knyazeva M.A., Kupnevich O.A. Domain ontology model for the domain "Sequential program optimization". Part 1. The terms for optimization object description. In The Scientific and Technical Information, 2002. (In Russian).

[Kasyanov, 1988] Kasyanov V. N. Optimizing transformations of the programs. Moskow: Nauka, 1988.

[Kleschev, 2001] Kleschev A.S., Orlov V.A. Requirements on a computer bank of knowledge. In Proceedings of the Pacific Asian Conference on Intelligent Systems 2001, Seoul, Korea, 2001.

[Kleshchev, 2002] Kleshchev A.S., Chernyakhovskaya M. Yu. The present state of computer knowledge processing. http://www.dvo.ru/iacp/es/publ/kpe.htm

[Knyazeva, 1999] Knyazeva M.A., Kupnevich O.A.. Expert system for simulation of program optimization. Joint NCC&IIS Bull., Comp. Science,12 (1999), 24-28.

Author information:

Irene L. Artemjeva: artemeva@iacp.dvo.ru

Margarita A. Knyazeva: mak@nt.pin.dvgu.ru

Oleg A. Kupnevich: garfield1@yandex.ru

Institute for Automation & Control Processes, Far Eastern Branch of the Russian Academy of Sciences

5 Radio Street, Vladivostok, Russia

PLA TOPOLOGICAL OPTIMIZATION BY BIPARTITE FOLDING

Liudmila Cheremisinova

Abstract. *This paper presents some results of PLA area optimizing by means of its column and row folding. A more restricted type of PLA simple folding is considered. It is introduced by Egan and Liu and called as bipartite folding. An efficient approach is presented which allows to find an optimal bipartite folding reducing exhaustive computational efforts.*

Key Words: *Programmable Logic Array, area optimization, PLA folding, bipartite folding*

Introduction

Programmable Logic Array (PLA) is widespread used hardware form for the structured design of digital VLSI systems due to the regularity of its structure. PLA layout design is easily automated because of its direct correspondence with PLA personality matrix. The price paid for the structural regularity is that much PLA area is unused because a large percentage of the row-column intersections are not personalized. Several techniques have been proposed for reducing the area required.

Two approaches are usually used to reduce the area occupied by the PLA: logic minimization that provides logic expressions with minimal number of products and topological minimization reclaiming unused space. The proposed paper deals with the problem of topological optimizing PLA area by means of its folding [1 – 5]. PLA folding allows reducing needed area without loss of regular structure of the PLA. There exist different types of PLA folding. They are based on merging several columns (and/or rows) of a PLA into a single column (row). The paper focuses on simple column (and/or row) folding that involves merging pairs of columns (and/or rows) into single columns (rows).

Folding a column of a PLA supposes to split that column into two segments so that two inputs or outputs may share the same column of the folded PLA. This means vertical lines are broken into an upper and a lower parts and two variables (pair of inputs or pair of outputs) don't need two lines but only two segments of the only line of the PLA. The task is to find such a permutation of PLA rows, which allows a maximum set of column pairs to be implemented on segments of single lines of the folded PLA.

In [3, 4, 5] a special type of simple column and row PLA folding is considered, in which all of the breaks of the columns (or rows) occur at the same level (Fig. 1, 2). Such a case is referred to as bipartite folding. The single break level of bipartite folding allows to speak of an upper and a lower folding regions, which contain the segments of those folded columns that are correspondingly above and below the breaks.

While bipartite folding may theoretically be only 25 percent as effective as regular simple folding for PLA's with column type constraints, this class of folding approaches the effectiveness of column simple folding for sparse PLA's. Some justifications for this approach are offered [3, 4], the most important of them are 1) the folded columns (rows) entering from the top (left) of PLA can be ordered independently of the folded columns entering from the bottom (right) of the PLA, that simplifies the routing signals; 2) the same algorithm can be applied for the row folding a previously column folded PLA, that simplifies subsequent PLA row folding; 3) a bipartite folded PLA allows to use much less additional area required for inclusion of testability features. The bipartite folding can be used to partition a large PLA into smaller PLA's, i.e. bipartite folding can be considered as a special type of PLA decomposition too.

In this paper a new bipartite PLA folding technique is presented. It is based on transformations of a Boolean column disjoint matrix that specifies the relation to be disjoint on the column set. That allows the PLA bipartite folding problem to be treated as a maximum unit minor problem. Before searching for a desired maximal unit minor some procedures of the column disjoint matrix reduction are made that allow pruning some rows and columns. Some of suggested results could reduce the search space for algorithm from [4], the other allow yielding optimal solutions.

Definitions

The overall combinational PLA is a standard two level NOR-NOR structure. Vertical lines of a standard combinational PLA are assigned with the input variables (and their complements) and output variables. Inputs run vertically through the first part of a PLA matrix, called as the PLA AND plane. It generates signals on its rows, which are used as inputs to the second part of a PLA matrix called as the PLA OR plane.

An example of the PLA AND plane (that is the example PLA from [4]) is shown in Fig.1. In this figure, columns are associated with complemented and uncomplemented inputs. Each horizontal line of the PLA carries a product term. A dot means placing a transistor on crosspoint of vertical and horizontal lines. This PLA AND plane will be used further throughout the paper. Here the OR plane is not shown but either AND plane or both AND and OR planes together can be described in symbolic form by a Boolean matrix. And the only difference is that inputs can share the columns with inputs only and outputs share the columns with outputs.

The area of a PLA is proportional to the total number of its columns times the number of rows. The area occupied by the PLA AND plane in Fig.1 is 273.

Before we formulate a mathematically tractable definition of PLA folding problem and its solution we have to give some definitions.

Each PLA column c_i implies the set $R(c_i)$ of rows, which are populated on it. Any two columns c_i and c_j are *disjoint* if $R(c_i) \cap R(c_j) = \varnothing$. A pair of disjoint columns is defined to be a column *folding pair*, and only those two columns of a PLA can be folded together.

For example PLA, columns c_1 and c_6 are disjoint, since $R(c_1) \cap R(c_6) = \{r_{12}, r_{21}\} \cap \{r_{14}, r_{16}, r_{17}\} = \varnothing$. But $R(c_5) \cap R(c_6) = \{r_{14}, r_{17}\}$, hence columns c_5 and c_6 do not generate folding pair.

The square Boolean matrix that depicts when the PLA columns are disjoint will be called a *column disjoint matrix* ***D***. This matrix has as many rows and columns as the number of the PLA columns. The element $d^j_i \in \boldsymbol{D}$ is 1 if columns c_i and c_j are disjoint, otherwise $d^j_i = 0$.

The following column disjoint matrix corresponds to the example PLA AND plane of Fig. 1:

	1	2	3	4	5	6	7	8	9	10	11	12	13
1	0	0	0	0	1	1	**1**	**1**	**1**	**1**	0	**1**	0
2	0	0	0	0	0	0	**1**	**1**	**1**	**1**	0	**1**	0
3	0	0	0	0	0	0	1	0	0	0	0	0	0
4	0	0	0	0	1	1	**1**	**1**	**1**	**1**	1	**1**	0
5	1	0	0	1	0	0	**1**	**1**	**1**	**1**	1	**1**	0
6	1	0	0	1	0	0	**1**	**1**	**1**	**1**	1	**1**	0
7	1	1	1	1	1	1	0	0	1	1	1	1	0
8	1	1	0	1	1	1	0	0	1	1	1	1	0
9	1	1	0	1	1	1	1	1	0	0	0	0	0
10	1	1	0	1	1	1	1	1	0	0	0	0	0
11	0	0	0	1	1	1	1	1	0	0	0	0	0
12	1	1	0	1	1	1	1	1	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0
weight	7	5	1	8	8	8	10	9	7	7	5	7	0

The relation to be disjoint on the column set is symmetric and irreflexive, so the column disjoint matrix ***D*** is symmetric too and has all 0's on the leading diagonal.

Bipartite folding is a column or row folding in which all of the breaks of the columns (or rows) occur at the same level (Fig. 2). The single break level of bipartite folding allows speaking of an upper and a lower folding regions, which contain the segments of those folded columns that are correspondingly above and below the breaks.

In bipartite folding all column breaks used to facilitate folding are made between the same two rows. Thus all columns, which are folded and placed at the top of the PLA must be disjoint from all columns folded and placed to the bottom of the PLA. Let the PLA columns of the set C^u belong to the upper folding region and the columns of the set C^l belong to the lower one.

Thus the *necessary and sufficient condition*, the pair C^u, C^l of the column sets involves bipartite folding, is the columns of C^u to be disjoint from each column of C^l (and symmetrically vice versa). So, the rows of the column disjoint matrix ***D*** corresponding to the PLA columns of the set C^l have 1's in all columns of the set C^u.

Such a pair of equinumerous sets is called a *bipartite folding pair of sets*. The cardinality of these sets defines the size of the bipartite folding pair of sets and the size of induced PLA bipartite folding.

It is clear that a pair C^u, C^l of column sets contains all information needed to fold a PLA i.e. it specifies the pairs of columns to be folded and their relative position (top or bottom).

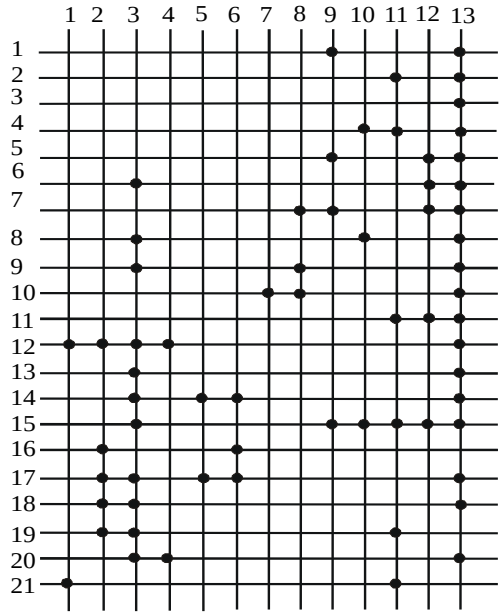


Fig.1. An example PLA

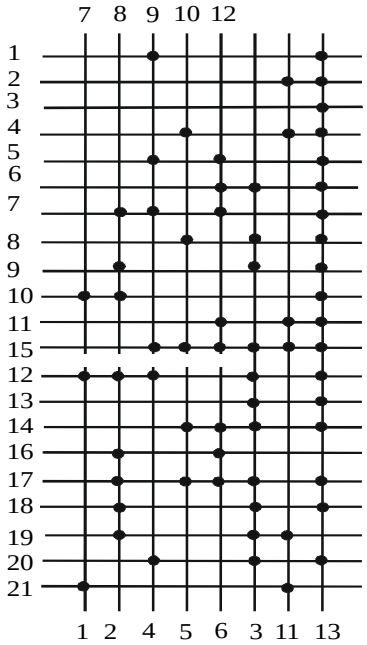


Fig.2. A bipartite folded example PLA

Transforming bipartite PLA folding into unit minor identification

The column disjoint submatrix, their columns and rows correspond to the PLA columns of the set C^u and the rows correspond to the PLA columns of the set C^l, is called a folding matrix ***F*** if the pair C^u, C^l involves bipartite folding. Such a matrix satisfies the following properties:

– it is square, $m \times m$ matrix, where m is the number of folding pairs of PLA columns;

– columns of the matrix correspond to the elements of the upper folding region, and rows the matrix correspond to the elements of the lower folding region;

– $C^u \cap C^l = \varnothing$;

– all matrix elements are 1's.

Thus a folding matrix is a unit square minor of the column disjoint matrix of a PLA. It implies from this that bipartite folding of size m exists if and only if there is a folding matrix of size m.

A folding matrix ***F*** of the bipartite folded PLA in Fig.2 is specified by the pair $C^u = \{c_7, c_8, c_9, c_{10}, c_{12}\}$ and $C^l = \{c_1, c_2, c_4, c_5, c_6\}$ and is highlighted above in the matrix ***D*** in thick print.

Identifying the optimal column bipartite folding set is to find a bipartite folding pair of sets of the greatest size, in other words the task is to find unit square minor of the PLA column disjoint matrix ***D*** with the greatest number of rows (or columns). That minor will be folding matrix required.

It is evident that the greatest size m of the folding matrix does not exceed $n/2$, where n is the number of PLA columns. The example PLA (Fig. 1) has 13 columns, so it allows for at most 6 pairs of folded columns.

Let the *weight* of the column c_i (row r_i) of the column disjoint matrix ***D*** be the number of 1's in it. The last row of the matrix ***D*** given above shows weights of its columns.

So, if a PLA permits bipartite folding pair of sets of size m then there is at least $2m$ columns having weights greater than or equal to m in the column disjoint matrix [4]. This statement follows from folding matrix definition. In other words, if looking for a bipartite folding of size m all candidate columns and rows of the column disjoint matrix should have weights greater than or equal to m.

The example PLA (Fig. 1) has only 9 columns with weights at least 6, so there is no folding matrix of size 6. But the example PLA has 11 columns with weights greater than or equal to 5. Thus a folding matrix is upper-bounded by 5.

Reducing the search space for bipartite folding

The optimal bipartite folding problem was shown to be NP-complete [3]. The classification of a problem as NP-complete is not sufficient reason to develop only heuristic optimizing methods. In many cases it is interesting to get just an optimal solution. Some results of reduction of exhaustive computational efforts on the search for the optimal PLA bipartite folding have been described.

The proposed PLA bipartite folding algorithm is organized such a manner to give an optimal solution during an exhaustive search but to allow finding "good" solution at the first its iteration. This algorithm starts from the pruned column disjoint matrix ***D***. And then after each algorithm step pruning techniques are made.

The evident way to find a bipartite folding pair of sets of the greatest size is to find a unit minor of the column disjoint matrix ***D*** of the greatest size. PLA columns that correspond to the unit minor columns can be referred to the upper folding region, and the PLA columns that correspond to the unit minor rows can be referred to the lower folding region. These two PLA column sets C^u and C^l are independent. Before the algorithm functioning the set C^u contains all the PLA columns and the set C^l is empty. It should be noted that the same PLA column is never placed into the unit minor row and column sets simultaneously (both in C^u and in C^l) owing to the column disjoint matrix ***D*** has 0's on the leading diagonal. In the process of the algorithm functioning, some PLA columns will be referred to C^l and eliminated from C^u. Thus corresponding rows (or columns) of the PLA column disjoint matrix ***D*** should be eliminated too.

Before seeking for the minors we can reduce search space, i.e. reduce the matrix ***D***. First, for a dense PLA some columns have small weights.

1. It is trivial that columns (and rows) with weights equaled to 0 should be eliminated [4]. They are useless for folding. As well the columns (and rows) with weights equaled to 1 are superfluous: the bipartite folding pair of sets of size 1 can be found trivially.

For example, there exist two superfluous columns (and rows) in the example column disjoint matrix ***D***: 3 and 13, having weights 1 and 0.

When the column disjoint matrix ***D*** has at least $2m$ columns having weights greater than or equal to m there can exist a folding matrix of size m. It can be found after reducing search space by means of elimination of all columns with weights less than m. But if the search fails we will have to decrease the value of m and repeat the search as it is proposed in [4]. So, we should choose such a value of m that there would be substantially greater than $2m$ columns having weights greater than or equal to m to refrain from repetition of the search for folding matrix.

2. The identical rows of matrix ***D*** should be united. These rows are not disjoint. Otherwise they would not be equal. And they are in the same relation with the rest PLA columns. So they will be in the same folding region in any permissible solution of the folding problem.

After removing superfluous columns (and rows) from the example column disjoint matrix ***D*** there are three groups of identical rows (and columns): 5 and 6; 7 and 8; 9, 10 and 12. Thus they can be united as it is shown below.

	1	2	4	5	6	7	8	9	10	11	12
1	0	0	0	1	1	1	1	1	1	0	1
2	0	0	0	0	0	1	1	1	1	0	1
4	0	0	0	1	1	1	1	1	1	1	1
5,6	1	0	1	0	0	1	1	1	1	1	1
7,8	1	1	1	1	1	0	0	1	1	1	1
9,10,12	1	1	1	1	1	1	1	0	0	0	0
11	0	0	1	1	1	1	1	0	0	0	0
weight	7	5	8	8	8	9	9	7	7	5	7

Here weights take into account repetition factor of the corresponding rows.

3. Further, the PLA column (and row) that is disjoint from every other can be added both to the set C^u, identifying the upper folding region, and to the set C^l, identifying the lower folding region. The column (and row) of the matrix ***D***, corresponding to such a PLA column, has all 1's but the only 0 (on the leading diagonal). The column weight is equal to n–1. So the column and the row can participate in no process of minor seeking and can be eliminated from matrix ***D***. After desired minor has been found the column can be included either in C^u or in C^l, depending on what of them has the less cardinality. Thus we should seek not necessarily a square unit minor but a unit minor of the greatest area.

For column disjoint matrix ***D***, depicted above, the group of rows (columns) 7 and 8 is disjoint from all others. So that group can be eliminated from ***D***. Having 1, 2, 3 in mind we get the following reduced column disjoint matrix ***D***:

	1	2	4	5	6	9	10	11	12
1	0	0	0	1	1	**1**	**1**	0	**1**
2	0	0	0	0	0	**1**	**1**	0	**1**
4	0	0	0	1	1	**1**	**1**	1	**1**
5,6	1	0	1	0	0	**1**	**1**	1	**1**
9,10,12	1	1	1	1	1	0	0	0	0
11	0	0	1	1	1	0	0	0	0
weight	5	3	6	6	6	5	5	3	5

For dense PLAs, most columns have small weights, resulting in a small size of the column disjoint matrix. For sparse PLAs, most columns have large weights, resulting in a large number of identical rows (and columns).

The search algorithm

The maximal unit minor is sought in the process of traversing the search tree in which each node represents a pair of two sets C^u and C^l of unit minor columns and rows. The rows for including to C^l are chosen from the set R of row-candidates corresponding to the node. C^u includes all columns that correspond to intersection of rows from C^l. At the first step the set C^u contains all the columns of the reduced column disjoint matrix ***D***, the set C^l of minor rows is empty and the set of row-candidates for C^l contains all rows of the matrix ***D***. The size of a folding matrix, associated with the unit minor described by the sets C^u and C^l, is the lesser of the cardinalities of these two sets.

The transition to a son of any node of the search tree consists in adding one more new row-candidate from R to the set C^l, thus decreasing, in general case the number of minor columns. The optimal solution is found during an exhaustive search of the tree reduced by means of some pruning techniques.

We are going to build successively one by one the unit minors of the Boolean matrix ***D***. After getting a new better solution we store it and its size k. At each next step we need to consider only those minors of greater size than that early-found ($i > k$). If we get a new better solution we can reduce the Boolean matrix ***D*** at the sacrifice of elimination of all its columns and rows with weights less than or equal k.

The search tree. The set of all possible unit minors is organized in the form of a search tree in which each node represents two sets: C^u and C^l, presenting columns and rows of the corresponding unit minor and the set R of row-candidates for including in C^l. The sons of a node represent those minors that contain one additional row in C^l. By traversing this search tree all possible unit minors of increasing size can be systematically examined. The process continues until a solution is found or all possible choices have exhausted.

Pruning the search tree. Traversal of the search tree is done in a depth-first manner, backtracking from a node whenever the sub-tree rooted by a node has been completely explored. The size of such search tree grows exponentially. However the suggested branch and bound algorithm restricts the exploration to within only a small subset of the nodes in the tree by means of pruning the search tree. A sub-tree can be pruned only if the algorithm can determine that this sub-tree contains no unit minor of size greater than that has been found so far.

A node of the search tree is called viable if the unit minor size is greater than the size of a minor found when traversing the tree before getting to this node. Only viable nodes need to be examined during an exhaustive search of the tree. When reaching a viable node, the rows with weights, less than or equal to the size of the unit minor corresponding to that node, should not be considered. This allows reducing the search space. If in any step a unit minor of the greatest possible size m is found then it is the desired maximal unit minor and the algorithm ends.

Backtracking is made if not viable node is reached. In that case a row, last included in C^l, is discarded and another one, not considered earlier, is selected.

Heuristic for choosing the traversing path of the tree. The algorithm suggested employs a simple heuristic to determine an order in which sub-trees, rooted in a node, are traversed so that a near optimal solution can be discovered quickly. Finding a near-optimal solution quickly is important since the sooner such a pair of sets C^u and C^l is found the sooner it can be used for pruning purposes.

The selection steps are important to produce a "good" solution the sooner the better, so to find a solution at the first branch of the search tree as close as possible to the maximum. It appears that justified strategy is greedy one. Therefore, a good heuristic is to select, at the first step, a row with maximum weight as candidate to be placed in the set C^l. Then a row of the matrix ***D*** is chosen that most intersects the set C^u: in other words it has the greatest number 1's in the columns of the set C^u.

Starting with reduced column disjoint matrix ***D*** depicted above, the algorithm finds maximal unit minor with C^u = {9, 10, 12} and C^l = {1, 2, 4, 5, 6}. It is highlighted in the matrix ***D*** in thick print. The following folding matrix ***F*** is associated with found maximal unit minor:

	7	8	9	10	12
1	1	1	1	1	1
2	1	1	1	1	1
4	1	1	1	1	1
5	1	1	1	1	1
6	1	1	1	1	1

The resulting bipartite folded PLA is given in Fig. 2. The area occupied by the bipartite folded PLA is 168 instead of 273 (Fig.1).

Finally, after algorithm completion we can get a unit minor that is defined by a pair of sets C^u and C^l of different cardinalities. In this case we have some extra (for the bipartite folding) PLA columns. That is another possibility to reduce the area of the PLA. Such extra members of C^u or C^l can be folded with any PLA columns that are disjoint from them. It is possible to find unit minors with row (or column) set containing only some of the extra PLA columns and with column (or row) set containing PLA columns not included in $C^u \cup C^l$.

An example of such a case of double folding is shown in Fig. 3 (that is example PLA from [4] with modified 13-th column) and Fig. 4. The area occupied by the double bipartite folded PLA is reduced to 147.

The following column disjoint matrix corresponds to the example PLA AND plane of Fig. 3:

	1	2	3	4	5	6	7	8	9	10	11	12	13
1	0	0	0	0	1	1	1	1	1	1	0	1	1
2	0	0	0	0	0	0	1	1	1	1	0	1	1
3	0	0	0	0	0	0	1	0	0	0	0	0	0
4	0	0	0	0	1	1	1	1	1	1	1	1	1
5	1	0	0	1	0	0	1	1	1	1	1	1	1
6	1	0	0	1	0	0	1	1	1	1	1	1	1
7	1	1	1	1	1	1	0	0	1	1	1	1	0
8	1	1	0	1	1	1	0	0	1	1	1	1	0
9	1	1	0	1	1	1	1	1	0	0	0	0	0
10	1	1	0	1	1	1	1	1	0	0	0	0	0
11	0	0	0	1	1	1	1	1	0	0	0	0	0
12	1	1	0	1	1	1	1	1	0	0	0	0	0
13	1	1	0	1	1	1	0	0	0	0	0	0	0
weight	8	6	1	9	9	9	10	9	7	7	5	7	5

There exists the only superfluous column (and row) in the above column disjoint matrix ***D***: 3, having weight 1. After its removing there are three groups of identical rows (and columns): 5 and 6; 7 and 8; 9, 10 and 12, they are united. Thus we have the following reduced column disjoint matrix ***D***:

	1	2	4	5	6	7	8	9	10	11	12	13
1	0	0	0	1	1	**1**	**1**	**1**	1	0	**1**	**1**
2	0	0	0	0	0	**1**	**1**	**1**	1	0	**1**	**1**
4	0	0	0	1	1	**1**	**1**	**1**	1	1	**1**	**1**
5,6	1	0	1	0	0	**1**	**1**	**1**	1	1	**1**	**1**
7,8	1	1	1	1	1	0	0	1	1	1	1	0
9,10,12	1	1	1	1	1	1	1	0	0	0	0	0
11	0	0	1	1	1	1	1	0	0	0	0	0
13	1	1	1	1	1	0	0	0	0	0	0	0
weight	7	6	9	9	8	9	9	7	7	5	7	5

The bipartite folding (Fig. 4) is induced from the maximal unit minor with C^u = {7, 8, 9, 10, 12, 13} and C^l = {1, 2, 4, 5, 6}. It is highlighted in the above PLA column disjoint matrix ***D*** in thick print. An extra PLA columns 7, 8 of the set C^u are disjoint from the PLA column 11 constituting one of the possible unit minors of size 2.

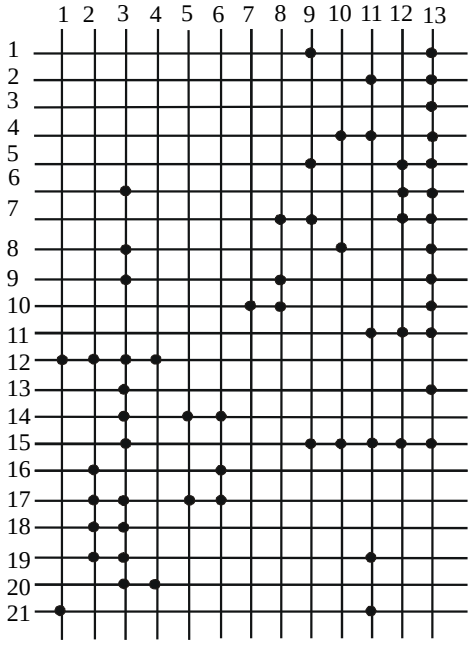


Fig. 3. The second example PLA

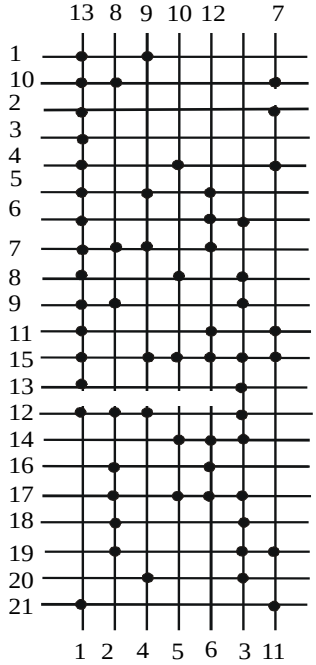


Fig. 4. A bipartite folded second example PLA

Conclusion

In this paper a new bipartite folding technique is presented. Compared with the other bipartite folding methods the suggested method has the following new features. The problem of bipartite folding is reduced to a search for a maximal unit minor of a Boolean matrix. The method contains some new procedures of reduction of Boolean column disjoint matrix that allows reducing the search space before functioning the basic bipartite folding algorithm. The approach presented in the paper may lead to the optimal bipartite folding without much wasteful computation.

References

1. G.D. Hachtel, A.R. Newton and A.L. Sangiovanni-Vincentelli, “An Algorithm for optimal PLA Folding”, IEEE Trans. Computer-Aided Design of Integrated Circuit Syst., vol. CAD–1, no 2, pp. 63–77, 1982.
2. G. DeMicheli and A. Sangiovanni-Vincentelli, “Multiple Constrained Folding of Programmable Logic Arrays: Theory and Applications”, IEEE Trans. Computer-Aided Design, vol. CAD-2, no 3, pp. 151–167, 1983.
3. J.R. Egan and C.L. Liu, “Bipartite folding and partitioning of a PLA”, IEEE Trans. Computer-Aided Design, vol. CAD-3, no. 3, pp. 191–199, 1984.
4. Chun-Yeh Liu and Kewal K. Saluja, “An efficient algorithm for bipartite PLA folding”, IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems, vol. 12, no 12, pp. 1839–1847, 1993.
5. J. E. Lecky, O.J. Murphy and R.G. Absher, “Graph theoretic algorithms for the PLA folding problem”, IEEE Trans. Computer-Aided Design, vol. 8, no 9, pp. 1014–1021, 1989.

Author information

Liudmila Cheremisinova, The United Institute of Informatics Problems of National Academy of Sciences of Belarus, Surganov str., 6, Minsk, 220012, Belarus, Tel.: (10-375-17) 284-20-76, E-mail: cld@newman.bas-net.by

ONE APPOACH TO INDIVIDUALIZED INTERFACE DESIGN

T. Gavrilova, E. Vasilyeva

***Abstract**: The paper reviews the main problems of the interface development for the distance learning systems. Several approaches for the interface development are described. The adaptive interface design is shown to be the best solution for the distance learning systems. The user model is considered to be the main adaptation criterion. The notion of user model comprises a set of representative information about the user. It includes demographic, cognitive, psychological, physiological and some other data characterized human-computer interaction peculiarities. The main adaptation criteria of the distance learning interface are determined. Some investigation's problems intended to design a special software tool for the user interface adaptation in the distance learning system are discussed. Specially designed software system to form user model InterTrivium is presented.*

***Keywords**: human-computer interaction, user model, adaptive interface, distance learning system.*

Introduction

The development of the user interface had become one of the main stages in the computer software design process. The great attention is paid for user interface to be user-friendly, comfortable and easy to learn. There is now a whole branch of science called usability in which specialists of different schools are working to build main principles of the user interface design. Psychologists, ergonomists, computer specialists try to find out basic criterions of the user interface development.

Usability engineers actively discuss problems of web-based design [Nielsen, 1999]. The main idea of there work is to think out fundamentals of user interface design as "User Interface for all". This approach is not well-posed from the psychologist's point of view. Users are individuals. They are differing by age, education, psychological and cognitive peculiarities. There is another approach to user interface design – developing of adaptive interfaces on the base of the user model. We suppose that the second approach is more fruitful for the user interface elaboration to such computer applications class as distance learning systems.

The user model development for the distance learning systems can also help to adapt learning process. Such user model characteristics as user education, his cognitive and psychological peculiarities can influence the learning material generation, navigation via learning course parts, type of students testing.

In this paper we report the first results of the current co-operative Russian-Byelorussian project intended to work out the methodology of distance learning systems interface adaptation and to develop special software tool to carry out interface adaptation. The initial project milestone was centered at the user and interface models' assembling for the distance learning system. The major distance learning interface properties able to adapt were combined into several special groups of the user interface model as well as key user features were arranged to the set of the user model groups. The preliminary propositions of the user and interface model's correlation were formulated to prove by the series of experiments. For the experimental part of research special software tool InterTrivium was developed. It carries out the user questioning and user model design. Some single-purpose questionnaires were composed to determine user's interface preferences.

Interface Model of the Distance Learning System

The distance learning system interface model involves four groups of interface parameters (see Fig. 1). There are functional, interactive, service and lay out features of the distance learning user interface. More than 50 different interface characteristics were primarily selected to include to the interface model. The investigation of all interface parameters adaptation attainability is a very difficult task. Thereby several interface characteristics, which have the significant influence the human-computer interaction and education process, were selected for our research.

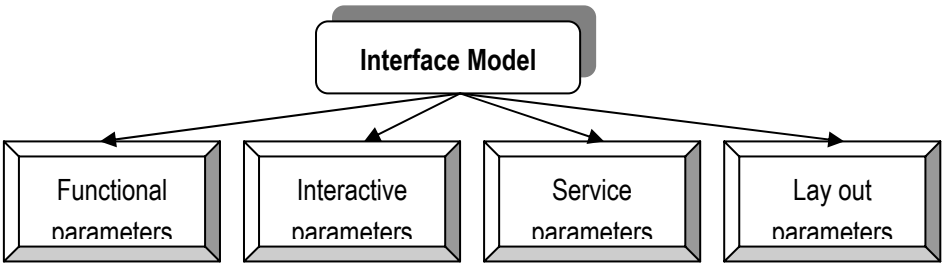


Fig. 1. Distance Learning System Interface Model Structure.

Functional interface parameters contribute greatly the system interaction behavior. It means difference in the representation of learning material and tests' performance. The examination of distance learning systems allowed to choose the following user functional parameters, which can be adapted:

- The set of the available working processes. This characteristic implies several distance learning system access modifications – for students, for administrator and for lecturer.
- The learning course material's structure. The user is allowed to study a certain limited suite of learning material according to his education, psychological peculiarities, interaction time and tests' results.
- Tests' content. Each user is provided by individual set of tests according to his user model.
- The navigation tool. The navigation tool adaptation supposes individual route for learning material study.

Interactive interface parameters determine the usability during the interaction with the system, the interaction scenario features. These characteristics group includes the information layout in the distance learning systems. it Is advisable to select the following parameters within this group:

- Dialogue type. The dialogue type management supposes providing different forms of tests.
- Controlling elements composition (menu configuration). The individual menu configuration should be formed for each user.
- Learning materials and tests' content being shown to the user at a time. The hypothesis of this parameter's adjustment according to the user's psychological peculiarities and his skills is put forward.
- Learning material performance format. Adaptation of this interface component assumes learning material presentation (font size, structure, graphic material availability) in the format most suitable to the user individual peculiarities.
- The highest possible hyperlink level (hypertext depth). According to the individual user model parameters' values the hyperlinks' number and hierarchy are generated to the distance learning system user.
- Navigation status. This characteristic's adaptation task is to follow the user navigation via learning material, to remind last visited page for starting with it during the next interaction with the system.

Service interface parameters include all objects participating in the reference and information dialog interface's functions. In our project we study adaptability of the reference information level, in other words we study individual user support system development feasibility.

Lay out interface parameters characterized information layout on the user display and level of user participation in it. In our project we tries to investigate adaptability of the following lay out interface parameters:

- **Current window set-up**. Distance learning system interface should be optimized to user screen sizes.
- Information lay out influences the learning material assimilation effect. Taking into consideration of psychological, physiological user's features will let to adapt this web-interface parameter.
- Menu appearance. The task of this parameter's adaptation is to design menu, which will be suit to user's psychological characteristics (as a text, icons, special images etc.).
- Background color.
- Text color.
- Hyperlinks color.

User Model

User model is not a new concept. First it was introduced in 1974 by the Institute of Informatics of the USA Congress. Now user model is interpreted as system's notion about the user, which generates either on the base of predetermined information about the user or on information acquired in the process of human-computer interaction. In spite of the fact that the user modelling is studying for a long time, there are no common principles of user model generation and it's implementation as a complex adaptation criterion.

The basic tendencies of the current user modelling research are:

- the number's increase and variety expansion of parameters included to the user model;
- user modelling use for adaptive systems development,
- user modelling implementation for the wide range of software systems development,
- attempts of generalized user models generation.

The network technologies expansion ensured new application fields for user modelling. Thus user model generation uses in adaptive hypertext navigation systems.

In the adaptive hypertext navigation systems the user model includes:

- users' goals,
- user's knowledge,
- user's hyperspace experience,
- user's background,
- user's preferences.

Also the more simple stereotype model (Rich model - [Rich, 1983]) is used for user knowledge representation. The stereotype model differentiates several groups of typical or «stereotype» users. For each user model measuring the system should offer several possible stereotypes.

The distinctive feature of our research is an attempt of user model generation process systematization and also including of psychological, physiological and cognitive features into the user model. Up to nowadays the major part of users modelling approach have comprised only the group of the human-computer interaction parameters (number of errors, main executed commands, visited pages) and also user knowledge about the subject domain. We suppose that including of psychological, physiological and cognitive features into the user model should improve human-computer interaction process quality as well as it will considerably increase user interface adaptation flexibility.

In our research we propose the concept of the user model [Rich, 1983; Wagner, 1982] as a set of formal representation of different factors, which affect the user's productivity in distance learning system environment. The user characteristics are grouped into several classes. The proposed distance learning system user model structure is shown on the Fig.2.

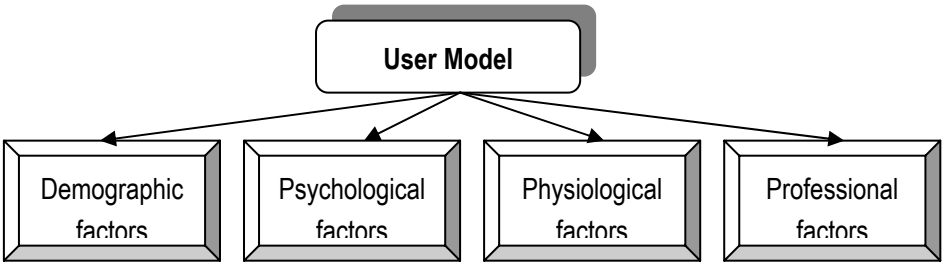


Fig. 2. Distance Learning System User Model Structure.

Demographic factors comprise such essential user's parameters as age, gender, first language, place of birth, social and cultural peculiarities. In our research we study two demographic factors – gender and age.

The group of the **psychological factors** in the distance learning system's user model consists of cognitive and communicative psychological peculiarities. The distance learning system's user model in our study includes the following psychological factors: *ability to study, conformism, the level of locus control, cognitive style, logical mentality style*.

The study of the psychological characteristics group is in the centre of attention in our research. For the distance learning systems tasks the user cognitive style [Witkin, 1981] is an extremely important factor. It considerably influences the problem solving way. The study of the user logical mentality style or deductive/inductive strategies can help to present the learning material more comfortable for the student, because those who are using deduction always perform their cognitive activity with the top-down strategy from the higher level of abstraction to more and more detailed schema and in the variant of induction the users ascend from the unconnected elementary concepts to metaconcepts.

The user's **physiological parameters** have the greatest influence the productivity of the human-computer interaction. We have included two factors into the distance learning system user model - *attention* and *mistakes frequency* - for the purposes of our research.

In our study **professional factors** group consists of the following user features: *expertise level, user professional experience in the subject domain, user education and user computer skills.*

In this part we have described the most common structure of the distance learning user model. In our project we study all factors mentioned above and some other factors to be included at the distance learning systems user model final structure.

Adaptation of the Distance Learning System

We consider the process of the distance learning system adaptation twofold – as interface adaptation and scenario adaptation. Scenario adaptation implies adjustment of the learning process scenario to the user peculiarities. Therefore we have included some characteristics, which in our opinion influence the learning materials navigation scenario, to the user model. Our adaptation comprehension corresponds the classical adaptation notion [Brusilovsky, 1996] (see Fig. 3).

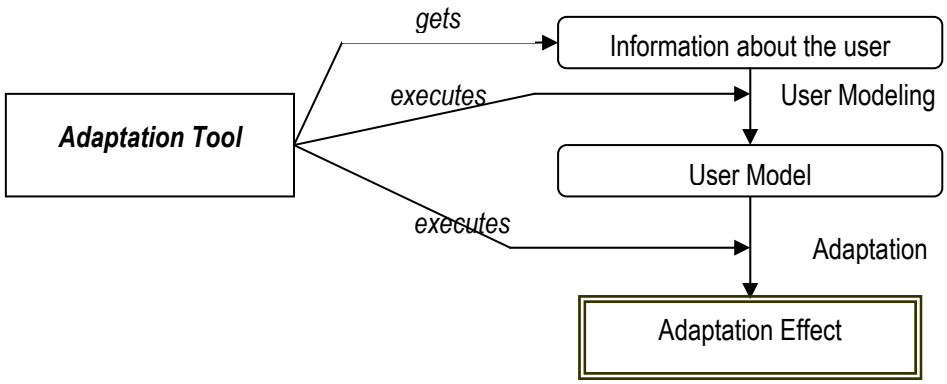


Fig. 3. Classical notion of the adaptation process in the adaptive systems

The major aim of the reported project is to develop interface adaptation tool for the distance learning systems. The process of the interface adaptation supposes inferences rules implementation to the interface generation. We are going to form knowledge base comprising user/interface models correlation rules. Now we carry out special experiments, which will allow to study correlation between user and interface models. The special software tool InterTrivium was developed to provide this experiments.

InterTrivium – User Model Acquisition Tool

InterTrivium is a specially designed software system to form distance learning system's student model (first version called TOPOS was developed by Voinov, second version TRIVIUM developed by Geleverya T.).

It is an application for multi-factor quiz's data interpretation developing and the user model generation. The system can work with all types of question-answer tests (graphical tests, multi-factors test etc.). InterTrivium includes tools for interactive visual editing of tests' descriptions and tests' scales.

The main InterTrivium major targets are:

- test development,

- quiz/questionnaire executing,
- the result data interpretation,
- user model generation.

In the system there is an intelligent tool for automatic verbal interpretation of test results for each respondent using rules, defined by experts-psychologists.

The prototype of InterTrivium is implemented in the framework of PHP scripting language and can store data in MySQL database or in text files.

The outcome of the described application may then be used both by Internet-based and standalone computer-aided learning systems. InterTrivium can serve as a user model generation tool and as an application for the different testing and queering support.

Now InterTrivium supports several tests on the user interface preferences and some professional psychological tests. Some experiments aimed to find correlation between the user model and distance learning interface components are providing. The interface adaptation tool for the distance learning system will be developed on the base of experiments results.

Acknowledgments

The research work reviewed in this paper has been carried out in the context of the Russian Foundation for Basic Research funded project “Adaptable Intelligent Interfaces Research and Development for Distance Learning Systems”(grant N 02-01-81019). The authors wish to acknowledge the co-operation with the Byelorussian partners of this project.

Conclusion

The importance of the user interface adaptability is evident. The intelligent interface development can noticeably improve distance learning systems outcome. The approach described in this paper can be titled as design and building of adaptive interfaces embedded in the distance learning systems via user modelling. This approach is based on the user-centred technology that puts stress at the usability, handiness and efficiency of human-computer interaction.

The described project is under active development. Currently, different system components are studied – up to considerable extent – separately. This is referred to, e.g., user modelling, distance learning, Internet programming, description of subject domain.

Bibliography

[Brusilovsky, 1996] Brusilovsky, P. Methods and techniques of adaptive hypermedia. User Modelling and User-Adapted Interaction, 6, 87-129.

[Nielsen, 1999] Nielsen J. Designing Web Usability: The Practice of Simplicity // New Riders Publishing; 1999. – 432 p.

[Rich, 1983] Rich E. Users are Individuals: Individualising User Models // Int. Journal of Man-Machine Studies, vol.18, 1983. – pp. 199-214.

[Wagner, 1992] Wagner E. A System Ergonomics Design Methodology for HCI Development. In: Proceedings of East-West Intern. Conference on Human-Computer Interaction EWHCI'92. - St. Petersburg.- pp. 388-410.

[Witkin, 1981] Witkin, H.A. & Goodenough, D.R. (1981). Cognitive Styles: Essence and Origins. NY: International Universities Press.

Author information

Tatjana Gavrilova - Saint-Petersburg State Polytechnical University, Intelligent Computer Technologies Dpt., Computer Science School Politechnicheskaya 29/9, 195251, Saint-Petersburg, Russia; e-mail: gavr@limtu.spb.su

Ekaterina Vasilyeva – Saint-Petersburg State Electrotechnical University, IIST, Prof. Popova St., bl.5, Saint-Petersburg, 198156, Russia; e-mail: vki@mail.ru

FROM THE MODEL-ORIENTED APPROACH TO USER INTERFACE DEVELOPMENT TO AN ONTOLOGY-ORIENTED ONE

Kleshchev Alexander; Gribova Valeriya

***Abstract**: The paper describes a new approach to user interface development which is an evolution of the model-based approach. The aim of the new, ontology-based approach is to eliminate the demerits of demerits of the model-based approach but to conserve its merits and, as a consequence, to lower more the cost of user interface development and maintenance. The main idea of our approach is to exchange models of different interface components for corresponding ontologies. The ontology models accessible by the Internet are used to form the models of there components.*

***Keywords**: Ontology, interface model, user interface development*

Introduction

A user interface is a central component of any modern software system. The efforts that are necessary to design, implement, modify and maintain a user interface add up to 70% of all the labor consuming for a software system development. Recent trends in development of software systems generally and of user interface in particular are applying the tools that free developers from low-level programming. These tools based on computer languages of the 4-th generation lower the cost of the development and maintenance for the applied software systems. There are a number of different tools for user interface development. But many user interfaces are implemented with interface builders as before. At the same time, there is a lack of tools that help designers to put all the pieces of an interface design together [I], no support for specifying the dynamic parts, and even for the static parts this support is not adequate [I, II]. These defects have given impetus to the development of a model-based approach for constructing user interface and now several model-based interface development tools have been built, for example [I, II, III, IV]. The main goal of this approach is an automatic translation of the declarative, high-level models of interface components into an executable program [I, II]. As a result, the number of procedural components developed in the course of designing an interface becomes considerably less, there is a possibility to reuse the knowledge making up a model, there are powerful tools supporting development [I]. Except these merits, this new technology has also a few demerits that are discussed in [I]. In addition to them it is possible to point out the following. First, up till now there has been no universally accepted standard of interface components. As a result, every model-based tool defines its specific model interface components. In the second place, the methods for implementation of different interface model components by the same model-based tool are different. For every model specific principles and mechanisms are used. This is a reason for difficulties in linking these models together. In the third place, many these tools require the description of application program in detail. This property makes the interface development and maintenance difficult. In the fourth place, different model-oriented tools are based on different declarative languages and data models. This fact makes also a transfer of the same models from one tool to another difficult. In the fifth place, a universally accepted terminology has not been formed within the model-oriented approach yet. As a result, the components, which are identical by meaning and effect, often have different names.

The aim of a new, ontology-based approach to the user interface development advanced in this report is to eliminate the existing demerits of the model-based approach but to conserve its merits and, as a consequence, to lower more the cost of user interface development and maintenance.

THE BASIC IDEAS OF THE ONTOLOGY-BASED USER INTERFACE DEVELOPMENT.

In this report four principal more precise definitions to the model-oriented approach are suggested.

1. A model of a user interface should be considered as a representation of such information about it that should be modified if some conditions of using the software system are changed. The information uniform by meaning should be combined into interface model components. Every component of the interface model should be

represented in the form of an ontology model [I, II, III, IV]. The representation of knowledge in the form of an ontology model is a universally accepted practice for development of knowledge based systems.

2. The ontology models accessible by the Internet should be used to form the interface model components for which it is possible.

The great current interest in ontologies is caused by the fact that the ontologies of different domains, represented in specific computer languages, can provide access of people as well as computer programs to a huge volume of information and knowledge stored in the Internet and give a possibility to software systems to use these ontologies and knowledge for solving different tasks. It is the ontologies that make possible the development the Internet of the second generation or semantic Internet [I]. The main problem of the semantic Internet is to give direct access of all comers to whole knowledge accumulated by the human civilization.

To implement the approach suggested in this report, it would be necessary to develop, store and maintain the user interface ontology in the Internet. This ontology could be used to form ontologies of particular user interfaces. The user interface ontology as well as the others has to be perpetually maintained.

3. The user interface and application should be designed and implemented as independent components that interact asynchronously through a set of common variables.

This improvement gives a possibility to do away with a task model description and a tool for linking the interface and the application. This idea can permit to decrease the cost of development and to make better interface maintainability.

4. The tool for interface development should be provided with a set of system functions. The interface designer should have a possibility to include necessary system functions in the developed interface according to the customer's requirements.

Any user interface has a set of possible functions according to the functions and tasks of user interface defined, for example, in [I]. These functions are to input data, to solve the task, to exit the application program, to view its results and so on

THE COMPONENTS OF AN INTERFACE MODEL.

The user interface model has to contain all the information about this interface that can be modified during the life circle of this interface. This model has also to be appropriate to automatic implementation of the interface (by translation or interpretation).

Since a dialog with an software system carried on within the framework of a domain concept system, and the concept system can be modified during the life circle of the software system according to user's wishes and also according to modifications of the domain and of the program functions, the user interface model has to contain some information about this concept system. The domain concept system has to be appropriate to express the input and output data of the system, the information about the application program control, about the interface control and also about an intellectual supporting user's actions.

Any user's dialog with the system is carried on using some display aspects of the interface such as methods and means for information transmission, for the control of user interaction with the application program, for dialog structures and so on. These display aspects can also be modified during the life circle of the system according to customer's wishes, to modifications of the program functions and to the development of our ideas about the display aspects of the interfaces. So this information about the display aspects has also to be a part of the interface model.

An software system consists of its user interface and application specific program (application program), which the user interface is closely connected with. The application program of the software system can be modified during its life circle according to modifications of the requirements to it. So the information about the application specific program has also to be a part of the user interface model. The less there is this information in the user interface model, the less it is probable that this information will have to be modified when the application specific program is modified, and so the better this part of the user interface model is.

The user interface presents the input and output information of the system to a user in terms of the domain concept system, but to the application specific program in the form of the values of its variables. In this manner there is a correspondence between the domain concept system and the set of the applied program variables.

This correspondence can be modified when the domain concept system and/or variables of the application specific program are modified. So the information about this correspondence has also to be a part of the user interface model.

The display aspects of the user interface are used in a dialog to present in a certain form the information that is transmitted from a user to the application program and from it to the user. The user understands this information within the framework of the domain concept system. In this manner there is a correspondence between the domain concept system and the display aspects used in the user interface. This correspondence can be modified according to a modification of the display aspects and of the domain concept system, and also according to user's wishes. So the information about this correspondence has also to be a part of the user interface model.

Any dialog is carried on according to a scenario. This scenario can be modified according to user's wishes, to a modification of the domain concept system and of the application program. So the information about the scenario of the dialog has also to be a part of the user interface model.

The customer's requirements to the set of system functions of the interface can also be modified in the course of using the application program. So the information about the set of system functions of the interface has also to be a part of the user interface model.

In this manner the model of any user interface of an software system can be considered as the set of the following models. They are the models of the domain concept system, of the display aspects for a presentation of these concepts in the interface, of the application program, of a correspondence between the domain concept system and the display aspects, of the scenario of a dialog and of the set of system functions.

The Domain Concept System Model.

The direction of attention towards the user means that the interaction between the user and the application program is realized in terms of the domain that the program is intended for. The domain is characterized by its concept system, which consists of concept definitions and of the descriptions for correspondences among them. An explicit representation of these definitions and correspondences is called domain ontology. Thus, a domain concept system model of every user interface is a formal model of a domain ontology.

The concept system used for an interaction between a user and an application program is a part of the concept system of an appropriate domain. It is possible to expect that in the near future formal ontology models for some domains will be presented in the Internet, and the number of models will be increased in time. The terminology of these ontologies will be standardized. If the domain ontology to which an software system relates has been formed, then the ontology of the concept system used by its user interface either is a part of this ontology or can be defined in terms of it.

Domain ontology often has the following property: many its components have the same structure. In this case it is convenient to form its description consisting of two levels. The first one is a reusable metaontology or an ontology of a domain class. It is the same for all the domains of the class. The second one is a domain ontology formed on the basis of the metaontology.

A Model for the Display Aspects of the Interface.

Now user interface development is an independent branch of software engineering, in which a rather stable concept system has been formed. The display aspects used in every specific user interface can be described as concrete definitions in terms of the concept system. Since designing the display aspects of a user interface is a professional activity, it should naturally be carried on within the framework of this professional concept system.

If this concept system is standardized, reduced to an ontology, formalized and made open to general use, designing the display aspects of the user interface will be considerably simplified. So practical use of the ontology-based approach to user interface development considerably depends on the fact, how quick the open to general use, standardized and formal and ontology model for the display aspects of the user interface will be formed. This ontology model has to contain descriptions for classes of objects having general structure and purpose. It should be possible to expand this ontology by adding new objects and their properties. The content of this branch of knowledge does not depend on a specific interface, but is determined by its achievements. Today a version of the graphic user interface ontology model is accessible by the Internet [I].

In that way, there are two possibilities for designing a model for the display aspects of the user interface in general case. The first one consists in extracting an appropriate concept system from the ontology model for the display aspects of the user interface had presented in the Internet. Extracting this concept system consists in defining the values of all the attributes in the definitions of general concepts. The second possibility is direct forming a specific model using the terminology accepted in this branch of knowledge.

An Application Program Model.

Any software system can be considered as consisting of two main components which are the application program and the user interface. The application program solves some tasks, and the user interface supports an interaction between a user and the application program. According to [XVII], the user interface is intended for supporting an interaction between a user and an application program, that is a process fulfilling a task. The functions of this interaction are transmission of information for a control of running the application program, transmission of the input data from a user to the application program, of the output data from the application program to the user. In this manner, the interface should transmit the input data and maybe some control information to an application program, and the output data to a user.

An application program model is the better the less it contains information. It is obvious that the minimum information about an application program is a description of all the application program variables by which the exchange of information between the user and application program takes place. It may be considered that every variable is either input or output one, and has an identifier and a possible value range, i.e. an application program model can also be represented in the form of ontology.

When an application program ontology is described, it is necessary to choose an appropriate concept system for the description of the variables. This description is possible:

- in terms of the domain concept system;
- in terms of the implementation language;
- in mathematical terms.

In the first case a transition from the application program ontology model to its implementation cannot be monosemantic. This fact will require an additional description of this transition.

In the second case using different implementation languages will require different models of the application program. This fact can considerably worsen their reusability.

In the third case the description of the variables in mathematical terms is monosemantically understood and reusable.

Thus, designing an application program model reduces to a description of its variables common with its interface in the form of ontology.

The Correspondence between a Domain Concept System and an Application Program Model.

There is a correspondence between domain concepts and application program variables. The input data should be transmitted from a user to the application program without information loss, as well as the output data from the application program to the user. In this way, this correspondence between the models of the domain concept system and of the application program has to be defined when an interface model is designed.

The Correspondence between a Domain Concept System Model and a Display Aspect Model of an Interface.

A specific ontology for the display aspects of an interface is a subset of the domain-independent ontology. It determines what interface elements are used in the concrete user interface, and what properties these elements have. The meaning or information components of these interface elements are domain terms described in the domain ontology. In other words, the domain term system forming the input and output data of the tasks is presented to a user in the user interface in the form of different interface elements such as lists, menus, tree control, edit field, graphical images and so on. Thus, there is a correspondence between the domain ontology and the specific ontology for the display aspects of the interface.

At the same time, different communications can exist which communicate the same information. They form a class of equivalent communications. Different users often need in the presentation of the same information in

different forms. And what is more, flexibility is an occurring everywhere requirement to the modern interface. It means a possibility of adjusting the interface to user's requirements and its adaptability, i.e. self-adjusting to the user.

Usually a concrete user interface consists of a subset of repeated interface elements, having different meaningful components. For example, they may be menus of the same type which have domain terms as their names and elements. In this case, there is no necessity to give a presentation in the interface for every domain term. It is enough to enumerate all the display aspects of the same type used in the interface and the domain term classes corresponding to them. The domain term classes are defined in the metaontology. But if there is no repetition in the interface, it is possible to assign explicitly every domain term its display aspect.

A Dialog Scenario Model.

In previous sections the knowledge branches are presented which are necessary for user interface model development. Besides the declarative model descriptions and the correspondences between them, it is necessary to define the functions for control of the interface by the application program and by the user. To do this, the tool for interface development should have a set of system functions determining user interface behavior. It is necessary to define in a dialog scenario model a correspondence between system functions and their presentation in the interface, and also to define attributes of appropriate system functions.

A DESIGN PROCESS AND IMPLEMENTATION TOOLS FOR THE INTERFACE MODEL

In general case, the design process and ontology-based tool architecture does not differ from the model-based ones. These tools also include design critics and advisors, automated design tools and so on for helping to interface developers.

The user interface design process begins with developing the ontologies or with editing them, i.e. with forming specific ontologies using general ontologies in the case, if the latter have been presented in the Internet. To do this, the tools should contain an ontology editor. The ontology editor gives a possibility to read an existing (for example, in the Internet) ontology and, using it, to extract a specific one. Since these ontologies may appear in the Internet in the near future, their inner representation format has to be standardized. This standardization gives the editor a possibility to read any ontology. If there has not been yet an ontology ready for use in the Internet, then it is possible to form this necessary ontology by the editor, too.

After forming the ontologies of the domain, of the display aspects of the interface and of the application program, models of all the correspondences between the different ontologies are defined by a linking editor. At last, a model of the whole user interface is formed by it. As well as in the case of the model-based approach, an implementation tool generates the source code either in a programming language or in the form of a certain format file, which can be read by an existing UIMS (User Interface Manager Systems). The interface model can also be interpreted at runtime.

Conclusion

The ontology-based approach to user interface development presented in this report permits to decrease the cost of the user interface development and maintenance. The first reason of this fact is reusing ontologies and their fragments both newly developed and presented in the Internet. The second one is using a library of system functions. The third one is using minimum linking between a application specific program and an interface. This property also simplifies the maintenance of software systems.

Bibliography

I Puerta, A.R., and Maulsby, D. Management of Interface Design Knowledge with MOBI–D. IUI97:International Conference on Intelligent User Interfaces, Orlando, January 1997, pp.249–252

II P. Castells, P. Szekely and E. Salcher. Declarative Models of Presentation. International Conference on Intelligent User Interfaces (IUI'97). Orlando (Florida), 1997, pp. 137-144.

III Puerta, A. R. Supporting User–Centred Design of Adaptive User Interfaces Via Interface Models. First Annual Workshop On Real–Time Intelligent User Interfaces For Decision Support And Information Visualization, San–Francisco, January, 1998.

IV P. Szekely, P. Sukaviriya, P. Castells, J. Muthukumarasamy, E. Salcher Declarative interface models for user interface construction tools: the Mastermind approach.. In Engineering for Humand-Computer Interaction, L. Bass and C. Unger Eds. Chapman & Hall, 1996 http://www.isi.edu/isd/Mastermind/mastermind-ia.htm

V Puerta A. the Mecano project: comprehensive and integrated support for model–based interface development. Computer–aided design of user interfaces, ed. by Jean Vandrdonckt. Pressed Universitaires de Namur, Belgium,1996, pp.19–25.

VI Lonczewski F., Schreiber The FUSE–system: an Integrated User Interface Design Environment, Proc. CADUI 96, J Vanderdonckt, ed., http://www.info.fundp.ac.be/~jvd/dsvis/cadui96.html

VII Foley J. History, results, and bibliography of the User Interface Design Environment (UIDE): An Early Model–Based System for user interface design and implementation. Proc. Eurographics Workshop design, specification, verification of interactive systems, F. Patern, ed., 1995, http://www.info.fundp.ac.be/~jvd/dsvis/dsvis94.html

VIII P. Szekely.User Interface Prototyping: Tools and Techniques. 1994 http://www.isi.edu/isd/humanoid-papers.html

IX da Silva, P.P., Griffiths, T. and Paton, N.W., Generating User Interface Code in a Model-Based User Interface Development Environment, Proc. Advanced Visual Interfaces, V. di Gesu, et al. (eds), ACM Press, 155-160, 2000.

X P.Szekely Retrospective and Challenges for Model-Based Interface. 1996 http://citeseer.nj.nec.com/szekely96retrospective.html

XI Puerta, A.R. Issues in Automatic Generation of User Interfaces in Model-Based Systems. Computer-Aided Design of User Interfaces, ed. by Jean Vanderdonckt. Presses Universitaires de Namur, Namur, Belgium, 1996, pp. 323-325.

XII N. Guarino, Formal Ontology in Information Systems.Proceedings of FOIS'98, Trento, Italy, 6-8 June 1998. Amsterdam, IOS Press.

XIII Kleshchev A.S. & Artemjeva I.L. Domain ontologies and knowledge processing. Techn. Report, Vladivostok: IACP, FEBRAS, 1999. 25 p.

XIV A. Kleshchev, I. Artemjeva A structure of domain ontologies and their mathematical models. In proceeding of the Pacific Asian Conference on Intelligent systems 2001. Korea Intelligent Information Systems Society. 2001. p.410-420

XV M. Uschold Knowledge level modeling: concepts and terminology. The knowledge Engineering Review, Vol.13:1.5-29

XVI www.ontoweb.org

XVII Coats R.B. and Vlaeminke I. Man-computer interfaces. Blackwell Scientific Publications 1987

XVIII http;//interface.es.dvo.ru/ontology.htm

Author information

Kleshchev Alexander, Professor, Head of the Expert System Department, Institute for Automation & Control Processes, Far Eastern Branch of the Russian Academy of the Sciences: Vladivostok, +7 4323 310424 kleschev@iacp.dvo.ru, http://www.iacp.dvo.ru/es

Gribova Valeriya, Ph.D. Senior Researcher of the Expert System Department, Institute for Automation & Control Processes, Far Eastern Branch of the Russian Academy of the Sciences: Vladivostok, +7 4323 314001 gribova@iacp.dvo.ru, http://www.iacp.dvo.ru/es

О ПРЕОБРАЗОВАНИИ OBDD, ПРЕДСТАВЛЯЮЩИХ КОНЕЧНЫЕ АВТОМАТЫ

Крывый С.Л. Гжывач В.

***Аннотация**: Рассматривается задача преобразования OBDD, представляющих конечные размеченные транзиционные системы, относительно некоторого отношения конгруэнтности. Преобразования ориентированы на получение OBDD минимизированной транзиционной системы по этому отношению конгруэнтности.*

***Ключевые слова:** конечный автомат, OBDD, отношение конгруэнтности, минимизация, преобразования.*

Введение

Упорядоченные бинарные таблицы решений или OBDD (*Ordered Binary Decision Diagrams*) [1] нашли широкое применение в различных областях *Computer Science*. Одним из наиболее важных приложений OBDD является возможность компактным способом представлять семантические сети больших размеров, представляющие базы знаний, а также расширять возможности методов верификации транзиционных систем (ТС) (например, методов *model checking* [2,3]). Важным классом размеченных семантических сетей и транзиционных систем является класс конечных автоматов. При большом числе состояний автомата возникает проблема его оптимального представления с целью упрощения анализа такого объекта и решение этой проблемы использует структуры данных, которые получили называние OBDD. Однако, при наличии информации об эквивалентных состояниях автомата, OBDD, его представляющую, можна сделать еще более оптимальной.

В данной работе решается следующая задача: дан конечный автомат A, его представление в виде OBDD и отношение конгруэнтности R на множестве состояний автомата *A*; требуется построить OBDD, которая представляет фактор-автомат *A/R*. Решить эту задачу можно традиционным способом: построить сначала фактор-автомат *A/R*, а потом построить OBDD, представляющую этот фактор-автомат. Однако часто отношение R появляется после того, как автомат *A* представлен OBDD и тогда проще преобразовать имеющуюся OBDD в OBDD, представляющую автомат *A/R*.

Необходимые сведения

Конечным X-автоматом называется четверка $A = (A, X, f, F)$, где A - конечное множество, элементы которого называются состояниями, $X = \{0, 1\}$ - алфавит входных символов или входной алфавит автомата, $f : A \times X \to A$ - функция переходов и $F \subseteq A$ - множество заключительных состояний.

Пусть $R \subseteq A \times A$ - некоторое отношение конгруэнтности относительно функции переходов на множестве состояний, т.е. $a\ R\ b \Leftrightarrow (\forall x \in X)(f(a, x)\ R\ f(b, x))$. Автомат, множеством состояний которого являются классы эквивалентности отношения R, называется фактор-автоматом автомата A и обозначается A/R. Важным отношением конгруэнтности является отношение автоматной эквивалентности R_A, которое для X-автоматов определяется следующим образом:

$$a\ R_A\ b \Leftrightarrow (\forall p \in F(X))(f(a, p) \in F \Leftrightarrow f(b, p) \in F),$$

где $F(X)$ означает полугруппу всех слов конечной длины в алфавите X, $a, b \in A$, а F - множество заключительных состояний. Автомат называется ***приведенным*** или ***минимальным***, если все его состояния попарно автоматно неэквивалентны. Известно, что фактор-автомат A/R_A является минимальным в классе всех автоматов, эквивалентных автомату A [4].

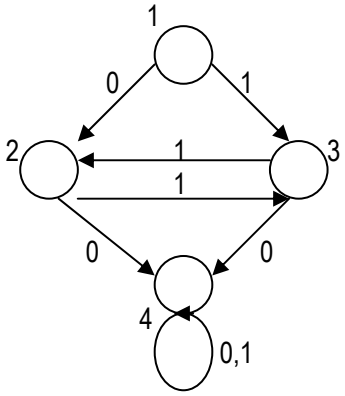
OBDD являются графическим представлением (в виде ациклического графа) булевой функции [1]. Использование OBDD для представления конечного автомата основывается на представлении его

функции переходов *f* в виде тернарного отношения R_f, определенного на кодах состояний автомата. Пусть, например, для состояний *a, b* ∈ *A* и некоторого *x* ∈ *X* имеет место *f(a, x) = b*. Тогда если *x, a, b* означают коды символа *x* ∈ *X* и состояний *a, b* ∈ *A*, то R_f *(x, a, b) = 1* для этих значений аргументов. Общее определение отношения R_f такое:

Теперь очевидно каким образом отношению R_f ставится в соответствие булевская функция g_f *(x, a, b),* значение которой равно 1 тогда и только тогда, когда R_f *(x, a, b)* = 1.

$$R_f(x, a, b) = \begin{cases} 1, & \text{если} \quad f(a,x) = b, \\ 0, & \text{если} \quad f(a,x) \neq b, \\ d, & \text{неопределено или несущественною} \end{cases}$$

Поясним это на примере. Пусть A = (A= {1,2,3,4}, X = {0,1}, f, F = {4}) – конечный X-автомат, функция переходов *f* которого определена нижеприведенным графом переходов.



Символы входного алфавита *X* кодируются естественным образом с помощью тождественного отображения, а состояния автомата пусть закодированы следующим образом (способ кодирования несущественен):

$$1 \to 00,\ 2 \to 01,\ 3 \to 10,\ 4 \to 11.$$

В этой кодировке отношение $R_f(x, a_1, a_2, b_1, b_2)$ имеет вид:

$$R_f(0,0,0,0,1) = 1,\ R_f(1,0,0,1,0) = 1,\ R_f(0,0,1,1,1) = 1,\ R_f(1,0,1,1,0) = 1,$$
$$R_f(0,1,0,1,1) = 1,\ R_f(1,1,0,0,1) = 1,\ R_f(0,1,1,1,1) = 1,\ R_f(1,1,1,1,1) = 1.$$

Этому отношению соотвествует булевская функция g_f:

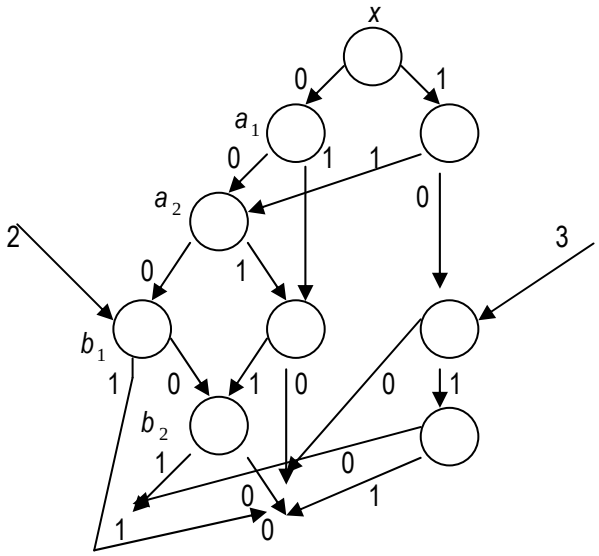
$$g_f(x, a_1, a_2, b_1, b_2) = \bar{x}\,\bar{a}_1\,\bar{a}_2\,\bar{b}_1 b_2 \vee x\bar{a}_1\,\bar{a}_2 b_1\,\bar{b}_2 \vee \bar{x}\,\bar{a}_1 a_2 b_1 b_2 \vee x\bar{a}_1 a_2 b_1\,\bar{b}_2 \vee$$
$$\vee\ \bar{x}\,a_1\,\bar{a}_2 b_1 b_2 \vee x a_1\,\bar{a}_2\,\bar{b}_1 b_2 \vee \bar{x}\,a_1 a_2 b_1 b_2 \vee x a_1 a_2 b_1 b_2,$$

где x, a_1, a_2, b_1, b_2 соотвествуют значениям входного символа из X и состояний a и b из A, соотвественно (a_i, если a_i = 1, и $\bar{a}_i$, если a_i = 0, аналогично и для b_i, i = 1,2).

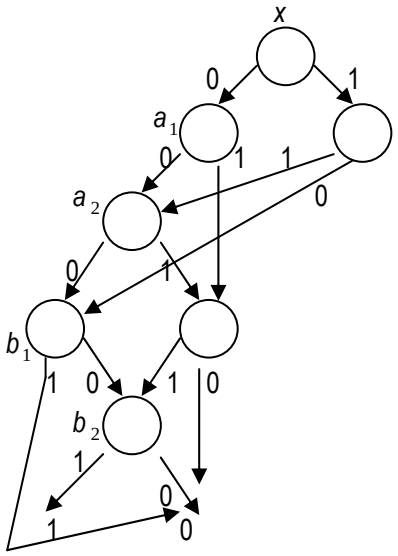
Преобразования OBDD неминимизированного автомата

Опишем преобразования OBDD, направленные на получение OBDD, представляющей минимальный автомат *A/*R_A. Пусть $K_1, K_2, \ldots, K_m$ - классы эквивалентности состояний автомата A по отношению R_A. Выберем по одному представителю из каждого класса K_i, i = 1,..., m. Пусть это будут состояния b_1, b_2, …, b_m, где $b_i \in K_i$. Этим состояниям соотвествуют коды $b_{11}b_{12}\ldots b_{1k}$, $b_{21}b_{22}\ldots b_{2k}$,…,

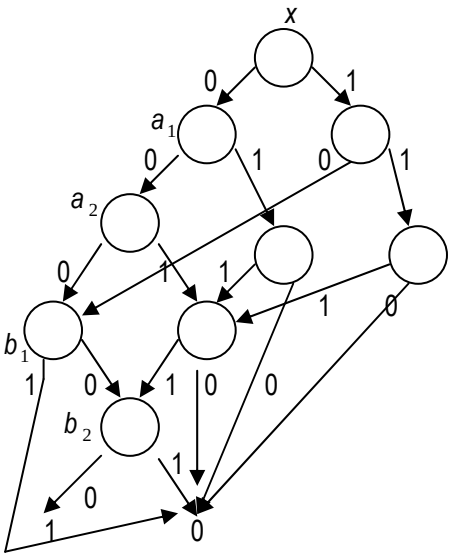
$b_{m1} b_{m2} \ldots b_{mk}$, где $k = log\ |A|$. Первый шаг преобразований состоит в переориентации всех дуг, ведущих в состояния из K_i (а точнее, на вершину являющуюся началом кода соотвествующего состояния), на вершину b_{i1}, i = 1,..., m. Второй шаг состоит в применении стандартных преобразований OBDD (см. [1]) с целью приведения полученной OBDD и удаления недостижимых вершин. Третий шаг преобразований состоит во введени вершин, если в этом есть нужда, с целью корректного доопределения переходов. Необходимость введения новых вершин заключается в том, что некоторые переходы в OBDD могут представлять состояния из K_i, i = 1, ..., m, которые эквивалентны состоянию b_i, но в минимизированном автомате эти состояния удалены. Удаление таких состояний может потребовать введения дополнительных вершин (которые были склеены и удалены в результате стандартных преобразований OBDD). Покажем всю последовательность преобразований на примере вышеприведенного автомата *A*. OBDD, представляющая автомат *A*, имеет вид:



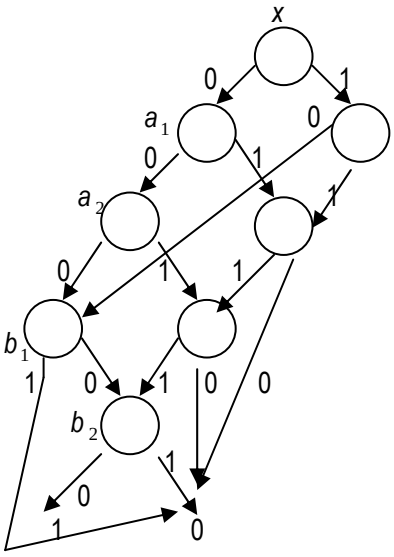
Автоматно эквивалентными состояниями в этом автомате являются состояния 2 и 3. Эти состояния имеют коды 01 и 10, соотвественно. Начальные вершины на OBDD этих вершин указаны стрелками. Выбираем представителем класса эквивалентности, например, состояние 2, тогда после первого шага преобразований, т.е. после переориентации дуг на вершину 2 и удаления недостижимых вершин, получаем такую OBDD:



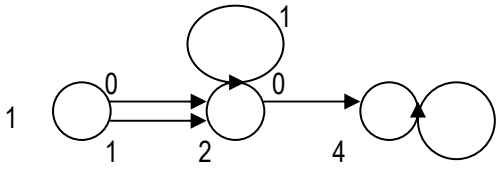
Эта OBDD еще не отвечает OBDD минимального автомата, поскольку в ней удалены все переходы в состояние 3, но не удалены переходы, ведущие из состояния 3. Для удаления этих переходов необходимо ввести новую вершину (в данном случае только одну). Это выполняется следующим образом. В OBDD имеется две вершины, соотвествующие несуществующему переходу $xa_1\overline{a}_2\overline{b}_1b_2$. Это вершины, помеченные символом a_1. Вот для этих вершин вводим новые вершины, помеченные символом a_2, и удаляем несуществующие переходы. Все эти преобразования показаны ниже на рисунках.



После очевидной редукции (склеивания двух новых вершин в одну) получаем окончательную OBDD, представляющую минимизированный автомат A/ R_A для автомата A.



Для того, чтобы убедиться в том, что эта OBDD действительно представляет автомат A/ R_A, приведем автомат, который получается из этой диаграммы.



В том, что полученный автомат есть приведенным автоматом для автомата A, можно убедиться непосредственно.

Из всего сказанного выше следует, что получение OBDD минимизированного автомата сводится к следующим преобразованиям:

а) удаления переходов в эквивалентные состояния, которое сводится к переориентации дуг, ведущих в эти состояния, на вершину, являющуюся началом состояния представителя данного класса эквивалентности;

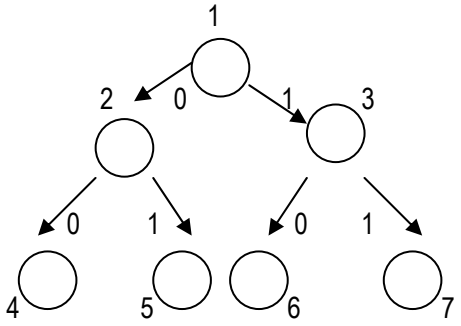
б) удаления переходов из эквивалентных состояний с возможным восстановлением необходимых вершин в OBDD;

с) удаления недостижимых вершин в графе OBDD.

Ациклические автоматы

Описанные выше преобразования OBDD, которые сводились к удалению эквивалентных состояний вместе с переходами, можно применить к преобразованию OBDD, представляющих ациклические автоматы. При этом отношение R может быть не объязательно конгруэнтностью, а только отношением эквивалентности и для построения приведенного автомата необходимо вычислять конгруэнтное замыкание отношения R. В этом случае конгруэнтное замыкание R^* отношения R определяется так: $a\ R^*\ b \Leftrightarrow (a\ R\ b) \vee (\forall\ x \in X)\ (f(a, x)\ R^*\ f(b, x))$.

Если отношение эквивалентности R задано, то построение приведенного автомата можно выполнить на OBDD. В общем случае отношение R^* может приводить к циклу. Для того, чтобы обеспечить ацыкличность отношения R^* будем рассматривать случай, когда R связывает состояния одного уровня. Такое ограничение гарантирует сохранения ацикличности отношения R^* и фактор-автомата A/ R^*. Продемонстрируем все сказанное на примере ациклического автомата A, граф переходов которого приведен ниже.



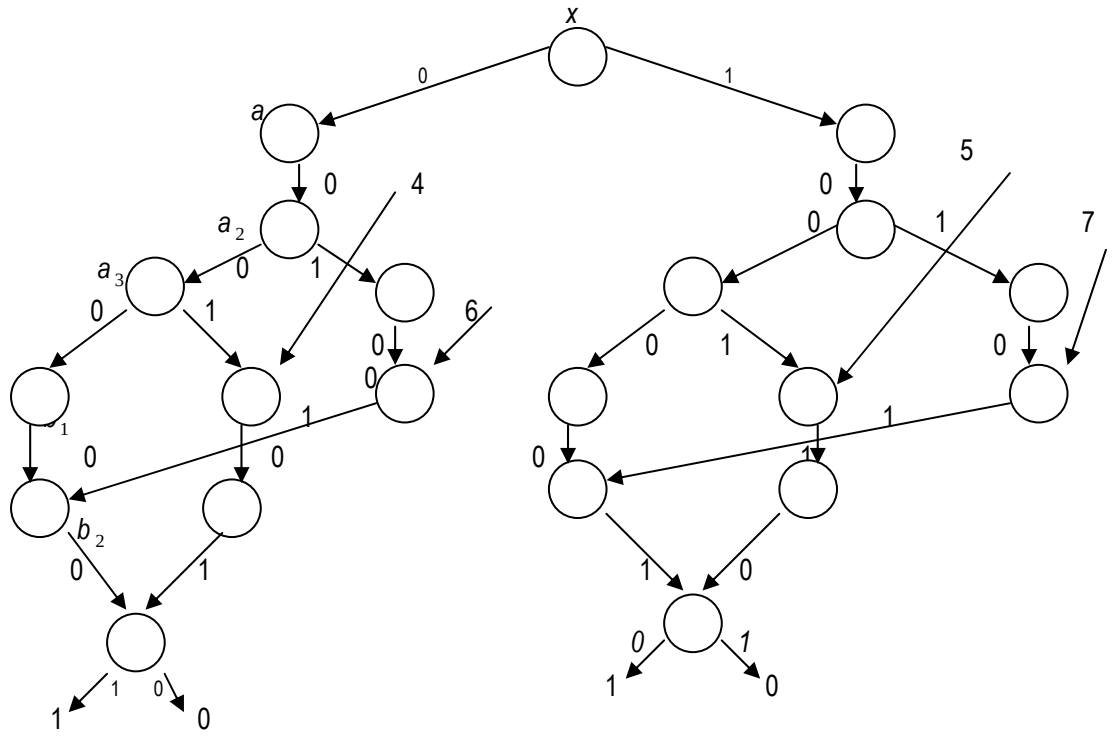
Пусть состояния данного автомата закодированы таким образом:

$$1 \to 000,\ 2 \to 001,\ 3 \to 010,\ 4 \to 011,\ 5 \to 100,\ 6 \to 101,\ 7 \to 110.$$

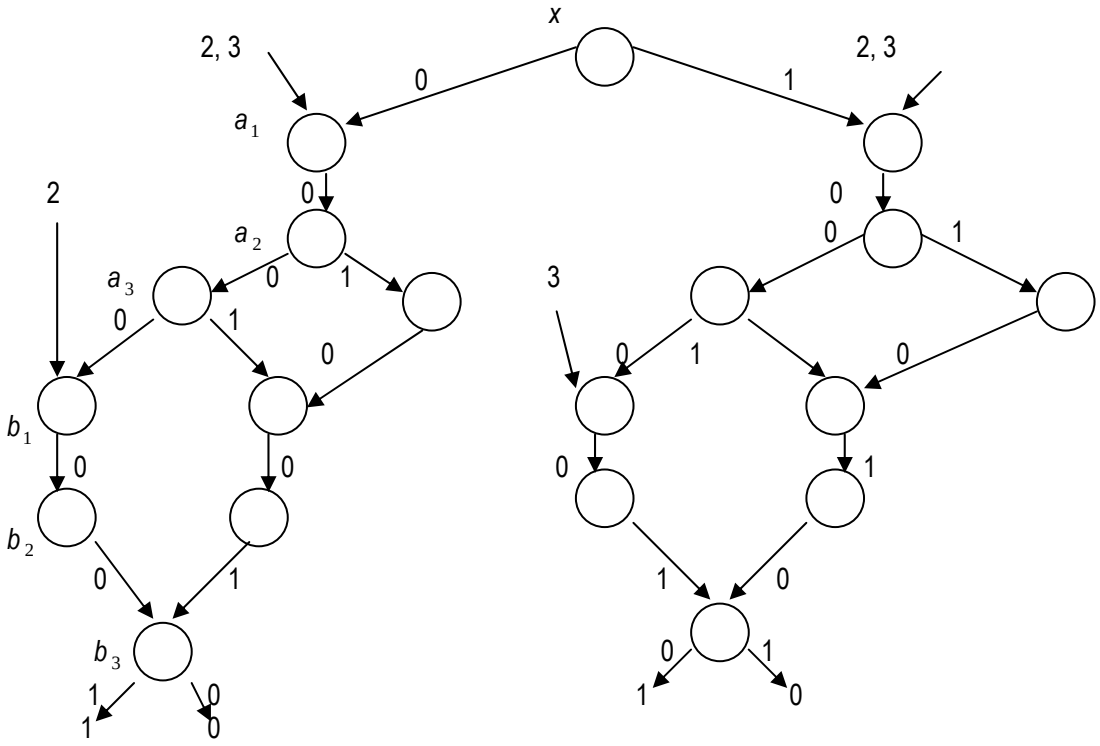
Булевская функция g_f, соотвествующая отношению переходов R_f имеет вид:

$$g_f(x, a_1, a_2, a_3, b_1, b_2, b_3) = \bar{x}\,\bar{a}_1\,\bar{a}_2\,\bar{a}_3\,\bar{b}_1\,\bar{b}_2\,b_3 \vee x\bar{a}_1\,\bar{a}_2\,\bar{a}_3\,\bar{b}_1 b_2\,\bar{b} \vee$$
$$\vee\ \bar{x}\,\bar{a}_1\,\bar{a}_2 a_3\,\bar{b}_1 b_2 b_3 \vee x\bar{a}_1\,\bar{a}_2 a_3\,b_1\bar{b}_2\,\bar{b}_3 \vee \bar{x}\,\bar{a}_1 a_2\,\bar{a}_3 b_1\,\bar{b}_2 b_3 \quad \vee$$
$$x\bar{a}_1 a_2\,\bar{a}_3 b_1 b_2\,\bar{b}_3.$$

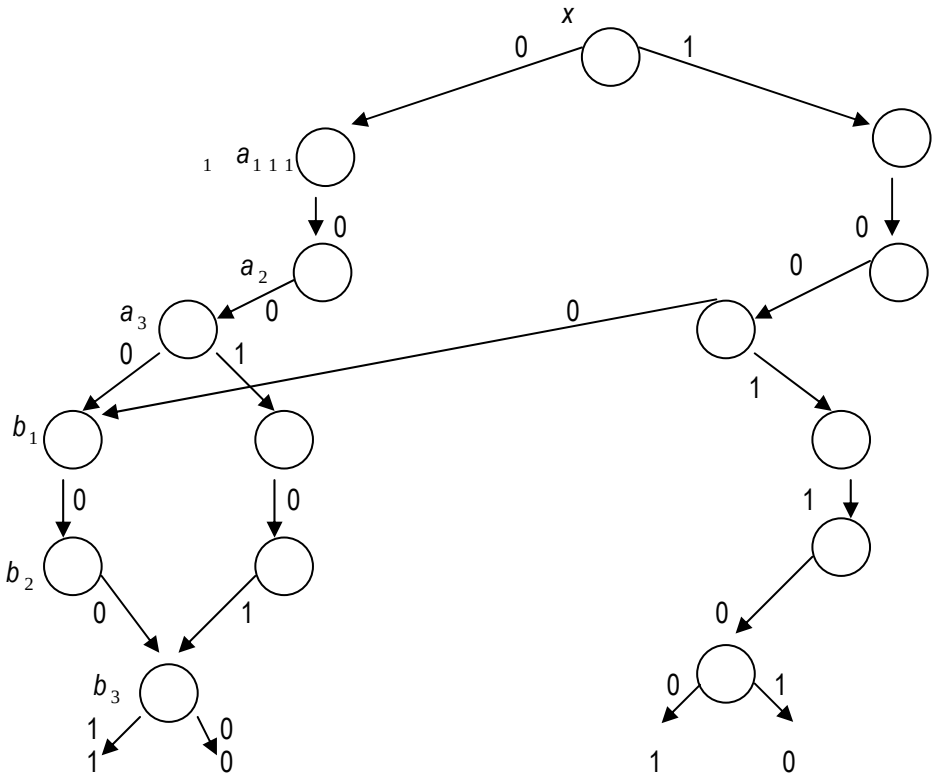
OBDD, которая соответствует этой функции, выглядит таким образом (в целях уменьшения громоздкости диаграммы, недостающие на ней дуги все ведут к вершине с пометкой 0):



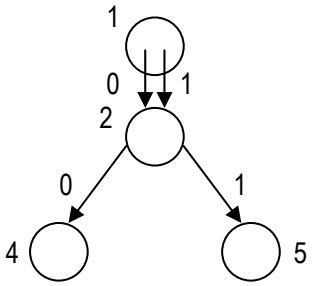
Пусть задано отношение R = {(4,6), (5,7)}. Тогда выбираем представителей в классах эквивалентности, например, состояния 4 и 5 и удаляем переходы в эти состояния. Следовательно, после выполнения первого шага преобразований получаем такую диаграмму:



В результате такого преобразования находим, что состояния 2 (001) и 3 (010) эквивалентны. Выбираем в качестве представителя класса эквивалентности состояние 2 и удаляем переходы из и в состояние 3. В результате повторения первого шага преобразований, получаем такую OBDD:



Эта OBDD соотвествует приведенному ациклическому автомату:



Заключение

Представленные в данной работе преобразования достаточно прозрачные и их применение ведет к уменьшению числа вершин OBDD. Приведенные ограничения на отношение эквивалентности, сохраняющее ацикличность автомата, имеют прямую связь с проблемой приведения общих подвыражений. Дальнейшие исследования преобразований OBDD связаны с другими алгоритмами теории конечных автоматов такими, как общий алгоритм минимизации автоматов, алгоритм детерминизации, алгоритмы синтеза и анализа и т.п. Цель этих исследований состоит в том, чтобы выяснить целесообразность проведения соответствующих преобразований на OBDD.

Библиография

[Bryant R.E., 1] Bryant R.E. Symbolic Boolean Manipulation with Ordered Binary Decision Diagrams. School of Computer Science, Carnegi Mellon University, Pittsburg. 1992 (june). - 34 P.

[Clarke E.M., Schlingloff B.-H., 2] Clarke E.M., Schlingloff B.-H. Model Checking. In ed. A. Robinson and A. Voronkov Handbook of Automated Reasoning, Elsevier Science Publishers B.V. - 2001. - P. 1369 - 1522.

[McMillan K.I.,3] McMillan K.I. Symbolic Model Checking: an approach to state explosion problem. PhD thesis. School of Computer Science, Carnegi Mellon University. -1992. - 212 P.

[Глушков В.М., Летичевский А.А., Годлевский А.Б., 4] Глушков В.М., Летичевский А.А., Годлевский А.Б. Методы математической биологии. Книга 6. Методы синтеза дискретных моделей биологических систем. Киев: Вища школа. - 1983. - 264 стр.

Сведения об авторах

Крывый Сергей – Институт кибернетики им. Глушкова НАН Украины, Украина, Киев, 03187, ул. Глушкова, 40, Институт кибернетики; e-mail: krivoi@i.com.ua

Гжывач Виолетта – Технический универсистет г. Ченстохов, Польша, e-mail: wiola@icis.pcz.czest.pl

AUTOMATIC TRANSLATION OF MSC DIAGRAMS INTO PETRI NETS

S. Kryvyy, L. Matvyeyeva, M. Lopatina

Abstract*: Development-engineers use in their work languages intended for software or hardware systems design, and test engineers utilize languages effective in verification, analysis of the systems properties and testing. Automatic interfaces between languages of these kinds are necessary in order to avoid ambiguous understanding of specification of models of the systems and inconsistencies in the initial requirements for the systems development.*

Algorithm of automatic translation of MSC (Message Sequence Chart) diagrams compliant with MSC'2000 standard into Petri Nets is suggested in this paper. Each input MSC diagram is translated into Petri Net (PN), obtained PNs are sequentially composed in order to synthesize a whole system in one final combined PN. The principle of such composition is defined through the basic element of MSC language — conditions. While translating reference table is developed for maintenance of consistent coordination between the input system's descriptions in MSC language and in PN format. This table is necessary to present the results of analysis and verification on PN in suitable for the development-engineer format of MSC diagrams. The proof of algorithm correctness is based on the use of process algebra ACP. The most significant feature of the given algorithm is the way of handling of conditions. The direction for future work is the development of integral, partially or completely automated technological process, which will allow to design system, test and verify its various properties in the one frame.

Keywords*: MSC diagram, MSC language, condition, automatic translation, Petri Net.*

Introduction

While designing and developing of either software or hardware it is of vital importance to detect and remove defects in product on its early stages in order to avoid time and resource losses. Development-engineers and test engineers (verifiers) use in their work different approaches and specification languages, that eventually leads to ambiguous understanding of the same portion of a project, to inaccuracies, incompletenesses or even to the inconsistencies in the initial requirements for the development. Development-engineers usually utilize languages intended for design purposes (as VHDL, MSC, SDL, UML and so on), while test engineers (verifiers) utilize languages effective in verification and testing (languages of mathematical logics, automata theory, algebraic and net languages). The way-out of this situation is a development of automatic interfaces between languages of these kinds. Given work is devoted to the development of automatic interface between the languages MSC (Message Sequence Chart) and PN (Petri Nets). The work suggests an algorithm of automatic translation of MSC diagrams compliant with MSC'2000 language standard [ITU-TS, 2000] into Petri Nets, which allow to automatically verify a lot of properties of the system under design. The algorithm works on a certain subset of MSC'2000 language.

1. Syntactic MSC constructions

MSC is a modelling technique that uses a graphical interface, which was standardized by ITU (International Telecommunication Union, earlier CCITT). It is usually applied to applications of the telecommunication domain, since they have properties of distributed reactive real-time systems, often in combination with SDL language [Grabowski, 1991]. These very properties of the systems make an MSC with possibility of scenario describing extremely suitable as for specification so for testing purposes. This means that MSC can be applied on every stage of system development, even on the stage of test case development. MSC describes message flow between the instances, which present asynchronously communicating objects of the system or system entities like blocks, services or processes of the system. One MSC diagram describes a certain portion of system behaviour or a scenario of communication between the instances.

MSC has two syntactical representations: textual and graphical, which are in one-to-one relation according to a standard. Basic elements of the language are those which define message flow, namely, *instance*, *message*, *action*, *set*->*reset*, *set*->*time-out*, *stop*, *create* and *condition*. An example of an MSC is presented on Figure 1(a). As far as this example is of illustrative kind, it only introduces a minimal set of possible MSC constructions as instances and messages. Let's describe basic elements of the language MSC'2000, which are considered in the given work.

1.1. Instances, messages and system environment.

Instance is a basic primitive of MSC, which in graphics is presented as vertical line with its name.

Message transmissions, which are acts of communication between instances, are presented by horizontal arrows with possible curve or tilt under angle for reflecting "overtaking" or "intersection" of messages. The beginning of the vector marks a sending of the message and its ending marks receiving of the message. Evens of sending and receiving of the messages are ordered along the instances so that sending of the message always happens earlier than its receiving. There is one more rule in standard MSC'2000 for ordering events along the instances: everything located above happens earlier than that located below. A MSC diagram imposes a partial ordering on the set of events being contained. A binary relation which is transitive, antisymmetric and reflexive is called partial order. The partial ordering can be described by its connectivity graph.

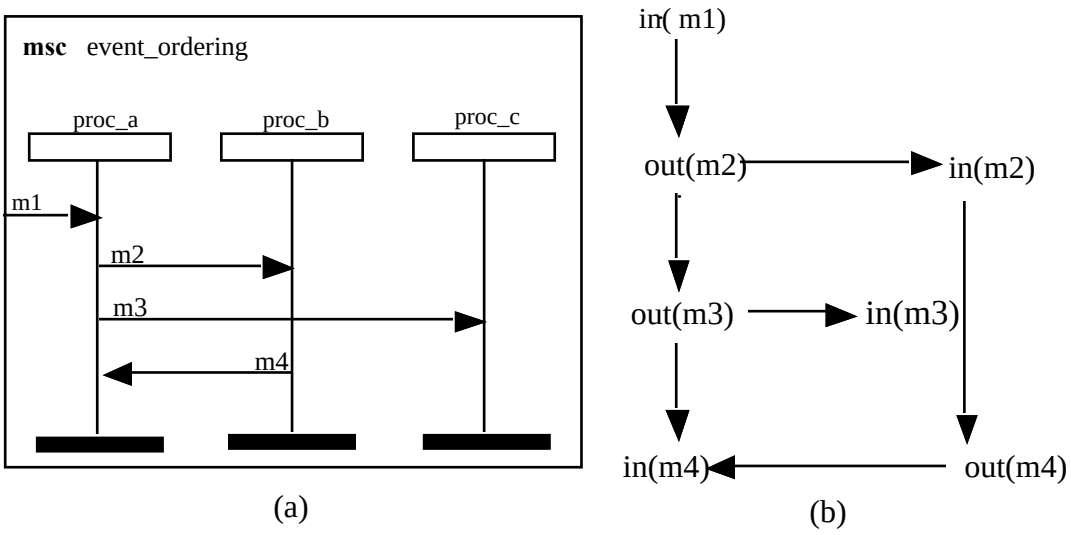


Figure 1

At the Picture 1(b) there is a graph, which reflects the order of events on instances in the diagram "msc event_ordering" (Figure 1(a)). Event out(m1) means sending of the message m1, in(m1) – receiving of message m1.

Environment of the system (the set of instances) is presented by the borders of MSC.

1.2. Conditions

Condition is used as for restricting or defining a set of MSC traces through indicating states of the system so for defining the composition of one MSC diagram from the several MSC diagrams. Condition can describe a global state of the system which is extended to all MSC instances existing in the time of this condition, it also can describe a state for a certain subset of MSC instances. In the first case condition is called global. Instances presenting dynamic objects can be began and finished, so far globality of a state considers dynamically changing set of instances.
Standard MSC'2000 [ITU-TS, 2000] defines conditions of two types: setting condition and guarding condition. Conditions of the first type are those which describe the current state of the system. Conditions of the second type restrict behaviour of the MSC to execution of events in a certain part of MSC depending on the value of the given guarding condition.
Besides of composition role, conditions according to the standard [ITU-TS, 2000] are the means of events synchronization. For example, if two instances share one and the same condition, then for each message between these instance its sending and receiving events shall happen both before or both after setting of the condition. If two conditions are ordered directly sharing the common instance, or indirectly through conditions on other instances, then this order must be respected on all instances that share these two conditions.

2. MSC semantics

The first version of MSC language standard defined the semantics incompletely and informally, however in ideal development of the language and its semantics shall go in parallel. Need of semantics standardization was becoming apparent, as even MSC experts could not always agreed on interpretation of the particular properties. Associated with this situation and extension of MSC language application in 1992 three new approaches of MSC semantics defining were submitted to standardization committee CCITT (now ITU-TS or Telecommunication Standardization section of the International Telecommunication Union).
The first approach was based on the theory of finite automata.
The second one was based on theory of Petri Nets using partially ordered sequences of events in the system. Given semantics is known to be extremely suitable for modelling of distributed asynchronous systems.
The third approach was based on process algebra ACP (Algebra of Communicating Processes) and interleaving model, when system is modelled by the sequence of transitions supposing that events are atomic and have no duration, and in every moment of time only one event can be executed. Semantics of interleaving simulates independence (asynchronism) among the subsystems (instances) through nondeterministic interleaving of independent parallel activities.
Every of the three approaches given for definition of MSC language semantics has its advantages and disadvantages, but the committee for standardization chose third approach to be the basis for formal definition of MSC language semantics. MSC language semantics based on process algebra [Bergstra, 1984] was defined for textual representation of MSC diagrams [ITU-TS, 1995] in expressions of process algebra that was called denotation semantics. Operational semantics is defined through addition of transitional rules to algebraic expressions. So operational semantics is reflection of MSC-specifications into transition system. Yet it should noted that operational semantics of MSC is not defined and standardized formally.

3. Algorithm of translation of MSC diagrams into Petri Net

The first progresses in defining MSC semantics basing on Petri Nets were presented in [Grabowski, 1993]. Nowadays research work on translating of MSCs into Petri Nets goes on and its results are covered, for example, in [Kluge, 2000], [Heymer, 2000], [Kluge, 2001]. However the authors develop semantics for the MSC'96 and MSC'2000 standards basing on Petri Nets, and, what's more, elaborate on MSCs into the Petri Nets translation, an automatic translation is not considered in these papers.

3.1. Description of algorithm

The following is the formalized description of the translation algorithm.
INPUT: A set of the MSC diagrams in the basic subset of the language MSC'2000.
OUTPUT: A Petri Net adequate to the set of initial MSC diagrams.

METHOD: Translation of every MSC diagram of the input set into a Petri Net is performed in two stages in parallel with composing of corresponding to each MSC diagram Petri Nets into one final combined Petri Net (synthesis).

Begin

Stage 1. Building of the partial-order graph to reflect events order in the initial MSC diagrams imposed by static requirements of MSC'2000 standard [ITU-TS, 2000] (see Figure 1).

Stage 2. Translating of the partial-order graph obtained at the stage 1 into the Petri Net.

End

Let's give more detailed description of the algorithm's stages and start with the description of how synthesis is proceeds.

Detailed description of the synthesis.

Each input MSC diagram is being translated into PN, the PNs are sequentially "glued" in order to compose (synthesize) a whole system in one PN, while the principle of such composition is defined at the level of MSC language - through *conditions*. Let's detail semantics of the basic element of MSC language *condition* as it was described in the MSC'2000 standard.

According to the MSC'2000 standard *condition* defines a system state of instance, which it covers . A system state of the instance is interpreted not as a current global state of the whole system, but as a set of the current values of a certain subset of attributes coherent with the instance (system object/entity), another words, *condition* is a precondition of occurring the event in the system. Let's explain this statement. MSC language was created to specify systems with local interconnections among the asynchronous parallel processes. Asynchronous models are typically built on cause-and-effect relation among events rather than on clocked sequence of system state changes. Asynchronous systems also present time moments or intervals as events. So event in this model is considered to be either atomic or compound with internal structure formed from "sub-events". Thus, *condition* is a precondition of occurring the event in the system and simultaneously a synchronizing action. The steps of the synthesis are carried out according to the following rules, on the assumption of fulfilling the following requirements.

The requirements:

1. We only consider a subset of MSC'2000, including the elements: instances, message inputs and outputs, setting conditions in textual representation. It is supposed, that all input diagrams are syntactically correct and satisfy the static requirements of the MSC'2000 standard. For example, for each event of a message sending the diagram shall have a pair event — a message consumption.
2. Naming of instances is complete and exact.
3. Naming of conditions is complete and exact, meaning of the conditions do not influence synthesis in any way.
4. Each instance shall have initial and final conditions either local or, probably, shared by several instances. The given requirement is not referred to a case of creation and termination of the instance within the scope of the given diagram.

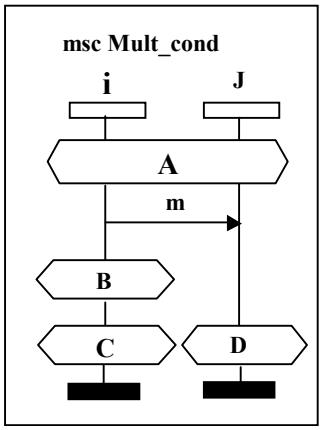


Figure 2

The condition is called *initial*, if the diagram has not any event or condition which precedes it, and *final*, if there is no event or condition after the given condition. Not only singular but also multiple initial and final conditions are

possible, in logic they can be represent as conjunction of all initial conditions, and, correspondingly, conjunction of all final conditions.

For example, at Figure 2 conditions **B** and **C** represent a multiple final condition, **D** — singular final condition or simply final condition, **A** is an initial condition.

5. Each diagram from the input set shall satisfy the following: every initial and final condition of MSC shall cover a minimum one instance, on which an event occurs (in given case, either sending or consumption of the message).

6. "Gluing" is forced and carried out according to the following rules.

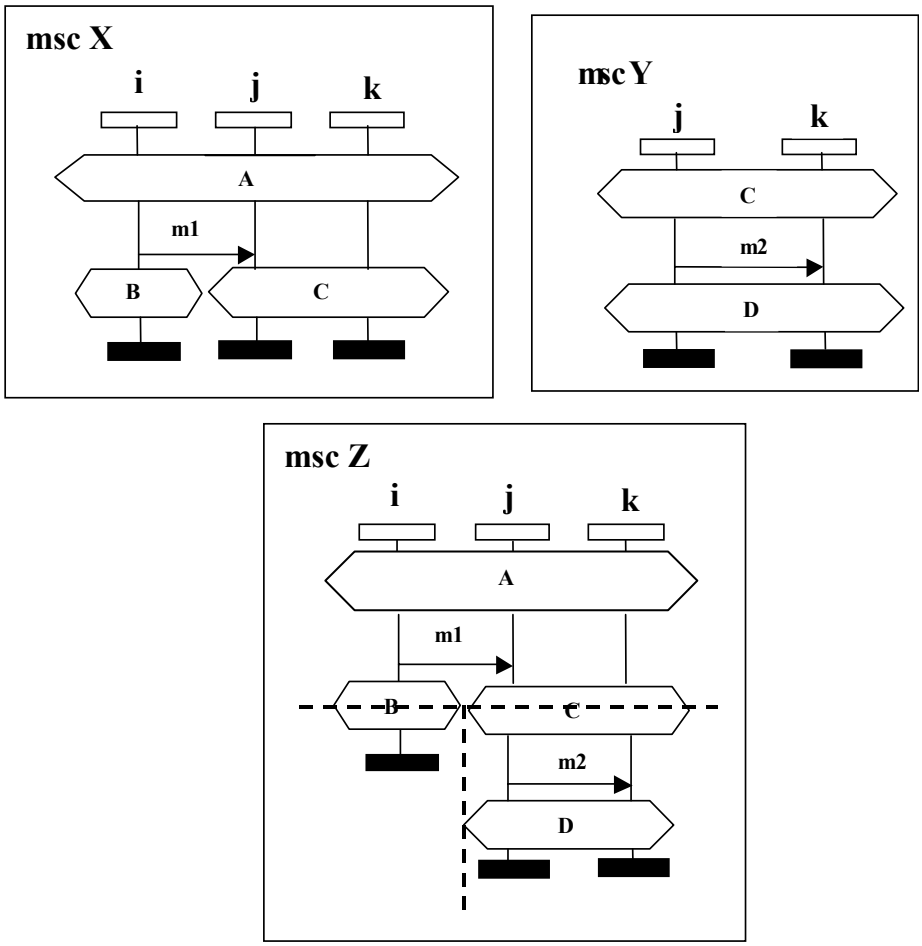


Figure 3

The Rules:

1. If the first MSC diagram has a final condition, which corresponds (has the same name) to the initial condition of the second MSC diagram, and these conditions cover the same set of instances in the both diagrams, than the first and the second diagrams can be “glued” together via the given condition regardless of the fact that this condition is either global or local for the diagrams. For example, at Figure 3 **msc Z** is a synthesis or composition of **msc X** and **msc Y**. The dashed line shows places of "gluing".

This rule also assumes the opportunity to “glue" by condition not only two diagrams in sequence, but also to “glue" together more than two diagrams at once (“multiple gluing”), this means that "gluing" is a non-deterministic alternative composition in opposite to the delayed choice operator for alternatives. As a result of gluing we receive an MSC diagram presenting a set of possible alternative traces in the system. Let's consider, for example, Figure 4. There are two possible continuations of the diagram **msc X**: it is **msc Y** and **msc Z**. And the choice is made when the condition **C** occurs and it is not delayed until the events presented on **msc Z** (**msc Result 1**) and on **msc Y** (**msc Result 2**) begin to differ from each other, namely, before passing of the messages **m** and **k**.

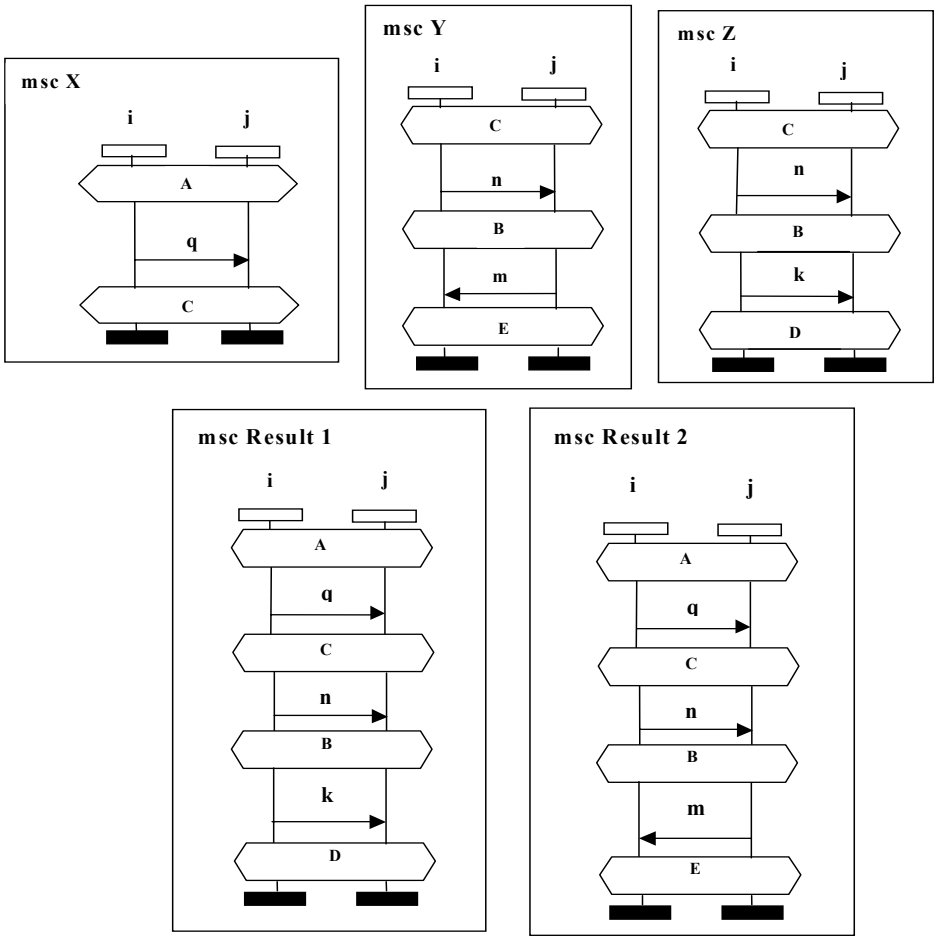


Figure 4

2. The following is a very important rule for "gluing" of MSC diagrams by **overlapped** and **multiple** conditions:

a) conditions occur simultaneously and are equal to the conjunction of these conditions,

b) the order of their enumeration in the diagrams is of no importance because of simultaneous occurrence of conditions,

c) these conditions can glue together independently from each other.

Let's consider an example at Figure 5. Each of MSC diagrams X1, X2, X3 is a possible continuation of **msc X**. The conditions **B** and **C** are overlapped in **msc X** and **msc X3**.

All requirements and rules mentioned above do not contradict the MSC'2000 standard, and only specify it. Composing of one common Petri Net is performed in accordance with semantics of the synthesis of MSC diagrams described above. A table of initial and final conditions of the initial MSCs plays a key role in the synthesis, as it contains all information necessary for composing. It is formed on the stage 1 during sequential processing of MSCs. Thus, the gluing is performed along with translation of MSCs applying the table of initial and final conditions.

Detailed description of stage 1. Partial-order graph for system events is built according to the following rules:

1. Directed edges of the graph set an order of the events.

2. Events of sending and consumption of messages and conditions correspond to the graph nodes. It is necessary to emphasize, that a condition is also represented by an event, namely, event of synchronization. Moreover the graph defines nodes of **two** types: a graph node which denotes intermediate condition or event of message consumption/sending, and a graph node which denotes initial or final conditions. They differ in the way of translation into the elements of Petri Net during the second stage.

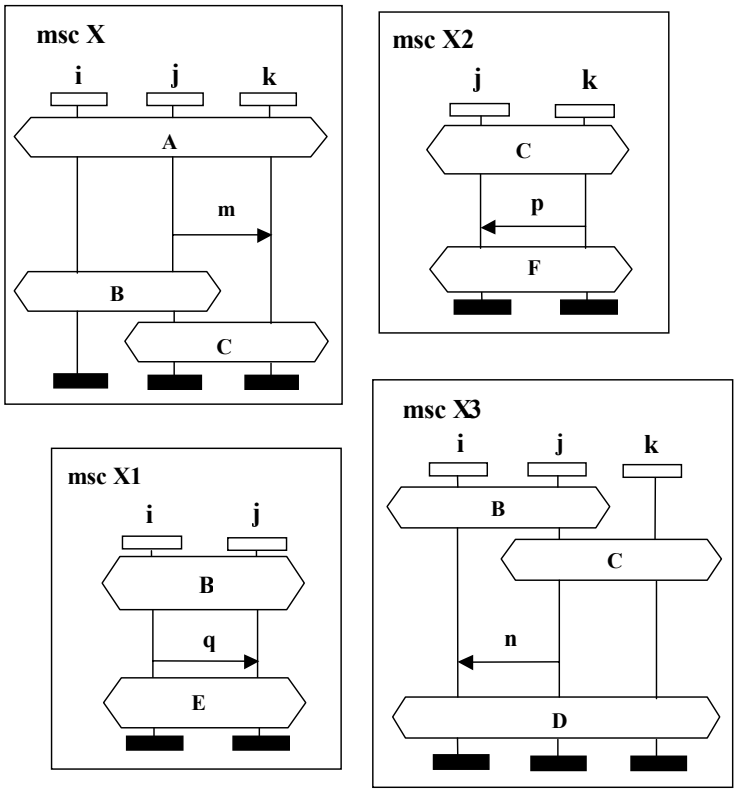


Figure 5

Detailed description of stage 2. Rules for the second stage of translation (translation of the graph into the Net) are the following:

1. Each directed edge of the partial-order graph is translated into a place of Petri Net.
2. An arrow of each directed edge of the graph corresponds to an arrow of Petri Net that directs tokens' flow in Petri Net.
3. A graph node of the **first** type (denoted intermediate conditions and events of an message input (consumption) and message output (sending)) is translated into a transition of Petri Net.
4. A graph node of the **second** type (denoted final and initial conditions) is translated into combination of transition and place of Petri Net. This place is a joint element for "gluing" into the common Petri Net (representing input system as a whole).

While translating reference table is developed for maintenance of consistent coordination between the input system's descriptions in MSC language and in Petri Net format. This table is necessary to present the results of analysis and verification on Petri Net in suitable for the development-engineer format of MSC diagrams.

Unfortunately, limited size of the paper does not allow to place here the proof of algorithm correctness even in the reduced version. We note only that the proof of algorithm correctness is based on the use of process algebra ACP.

Conclusion

Summing up, we note, that the algorithm of an automatic translation, presented in the paper, considers only the subset of MSC language, therefore extending of this subset to the maximum or even up to the whole MSC language is an obvious direction of the further research. The most significant feature of the given algorithm is the way of handling of conditions, since the literature indicates this problem in translation process as the most difficult . What is also important is obtaining of necessary experience for developing analog translators for the languages: SDL, UML, etc. The ultimate goal of this research is development of integral, partially or completely automated technological process, which will allow to design system, test and verify its various properties in the one frame.

Bibliography

[Grabowski, 1993] J. Grabowski, P. Graubmann, E. Rudolph. Towards a Petri Net Based Semantics Definition for Message Sequence Charts, In O.Fergemand and A.Sarma, editors, SDL'93 Using Objects, Proceedings of the 6th SDL Forum, Darmstadt, 1993. Elsevier Science Publishers B.V.

[Grabowski, 1991] J. Grabowski, P. Graubmann, E. Rudolph. Towards an SDL-Design-Methodology Using Sequence Cart Segment, SDL'91 Evolving Methods - O.Fergemand and A.Sarma, editors, North-Holland, 1991.

[CCITT, 1992]. CCITT Recommendation Z.120: Message Sequence Chart (MSC). Geneva,1992.

[Reniers, 1995] M. A. Reniers.Static semantics of Message Sequence Charts. In SDL'95 with MSC in CASE, Proceedings of the Seventh SDL Forum, Oslo, 1995. Elsevier Science Publishers B.V.

[ITU-TS, 1995] ITU-TS Recommendation Z.120 Annex B: Algebraic semantics of Message Sequence Charts. ITU-TS, Geneva, 1995.

[Котов, 1984] В.Е.Котов Сети Петри. М.:Наука, 1984, 157 стр.

[ITU-TS, 2000] ITU-TS Recommendation Z.120: Message Sequence Chart (MSC). ITU-TS, Geneva, 2000.

[Belina, 1992] F. Belina. SDL Methodology Guidelines, CCITT, Geneva, May 1992.

[Bergstra, 1984] J.A. Bergstra, J.W. Klop. Process Algebra for Synchronous Communication, Inf.&Control 60,pp.109-137,1984.

[Mauw, 1995] S. Mauw, M.A. Reniers. Thoughts on the meaning of conditions. Experts meeting SG10, St.Petersburg TD9016, ITU-TS,1995.

[Kluge, 2000] O. Kluge. Time in Message Sequence Charts Specifications and How to Derive Stochastic Petri Nets. In: Proceedings of the Third International Workshop on Communication Based Systems (CBS3), Berlin, March 2000.

[Heymer, 2000] S. Heymer. A Semantic for MSC based on Petri-Net Components, SIIM Technical Report A-00-12, Informatik-Berichte Humboldt-Universitдt zu Berlin, June 2000.

[Kluge, 2001] O. Kluge, J. Padberg, H. Ehrig. Modeling Train Control Systems: From Message Sequence Charts to Petri Nets, Technische Universitдt Berlin,http://cs.tu-berlin.de/SPP/index.html, 2001

Authors information

Sergiy Kryvyy - Institute of cybernetics, NAS of Ukraine, Pr. Glushkova, 40, Kiev-03187, Ukraine; e-mail: krivoi@i.com.ua

Lyudmila Matvyeyeva - Institute of cybernetics, NAS of Ukraine, Pr. Glushkova, 40, Kiev-03187, Ukraine; e-mail: luda@iss.org.ua

Mariya Lopatina - Institute of cybernetics, NAS of Ukraine, Pr. Glushkova, 40, Kiev-03187, Ukraine; e-mail: marichkay@mail.ru

ГЕНЕТИЧЕСКИЙ АЛГОРИТМ ОПРЕДЕЛЕНИЯ ПАРОСОЧЕТАНИЙ ГРАФА

Владимир В. Курейчик, Виктор М. Курейчик

***Аннотация**: Определение максимального паросочетания в графе является одной из важнейших NP-полных комбинаторных задач.*

В работе рассматриваются основные принципы генетического поиска. позволяющие эффективно управлять им. На основе этих принципов разработан комбинированный генетический алгоритм определения паросочетаний графа, позволяющий получать набор качественных решений за приемлемое время. Построен комплекс программ и проведены экспериментальные исследования, подтверждающие эффективность разработанного алгоритма.

***Ключевые слова**: Граф, генетические алгоритмы, паросочетания, генетический поиск.*

Введение

В настоящее время использование моделей эволюции во многом определяет качество решения инженерных и практических задач. Аналогия эволюционного развития естественных и искусственных систем позволяет развить подходы и методы эволюционного моделирования, генетических оптимизационных алгоритмов, распределенного искусственного интеллекта и искусственной жизни [1,2]. Использование графовых и гиперграфовых моделей дает возможность эффективного решения инженерных и практических задач. Одной из важнейших комбинаторных задач на графах является определение максимального паросочетания. Данная задача относится к классу NP-полных проблем [3,4]. Поэтому разработка эвристических алгоритмов является актуальной и важной задачей. В работе предлагается новый метод определения максимального паросочетания в двудольных графах, основанный на комбинированных генетических алгоритмах [5,6]. Данный метод в отличие от известных позволяет получать набор квазиоптимальных решений за приемлемое время.

1. Основные принципы генетического поиска

Процесс эволюции основан на анализе начальной популяции альтернативных решений и использует различного вида эволюционные, генетические и комбинированные алгоритмы. Генетические алгоритмы (ГА) начинают свою работу с создания исходного множества альтернативных решений. Затем эти «родительские» решения создают «потомков» с лучшими свойствами путем случайных, направленных или комбинированных преобразований. После этого оценивается эффективность каждого альтернативного решения и они подвергаются селекции. Во всех известных моделях эволюции используется принцип «выживания сильнейших» или его модификации, т.е. наименее приспособленные решения устраняются, а лучшие решения переходят в следующую генерацию. Затем процесс повторяется вновь и вновь до получения наборов оптимальных или квазиоптимальных решений [1,2,5,6].

При решении оптимизационных задач (ОЗ) на графах генетические алгоритмы дают много преимуществ. Одно из них – это приспособление к изменяющейся окружающей среде. При использовании ГА популяцией является база знаний, которую можно анализировать, дополнять и видоизменять применительно к изменяющимся условиям. Для этого не требуется полный перебор. Другое преимущество ГА для решения задач состоит в способности быстрой генерации достаточно хороших альтернативных решений.

Предлагается комбинированный ГА, состоящий из трех основных блоков. Первый блок назовем препроцессором. Здесь производится создание одной или некоторого множества начальных популяций на основе различных методов локального поиска [5,6].

Второй блок состоит из четырех этапов:

- выбор представления решения;
- разработка операторов случайных, направленных и комбинированных изменений;
- определение законов выживания решения;
- рекомбинация.

Третий блок назовем постпроцессором. Здесь реализуются принципы эволюционной адаптации к внешней среде (лицу, принимающему решения) и самоорганизации. Схема такого поиска показана на рис.1.

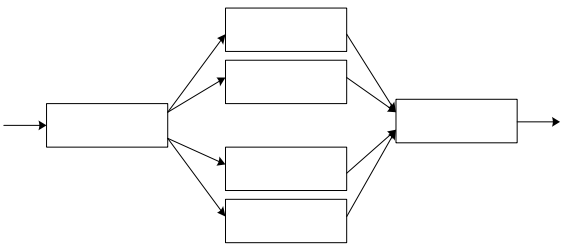


Рис.1. Архитектура генетического поиска

Это горизонтально организованная архитектура генетического поиска (ГП). Ее преимущество состоит в том, что в ней все уровни связаны с уровнем внешней среды и могут общаться между собой. Недостаток

горизонтальных архитектур - это сложность координации работы отдельных уровней. Приведем основные принципы, которые эффективно используются при генетическом поиске [7]:

- Принцип целостности. В генетических алгоритмах значение целевой функции альтернативного решения не сводится к сумме целевых функций частичных решений.
- Принцип дополнительности. При решении ОЗ на графах в ГА возникает необходимость использования различных не совместимых и взаимодополняющих моделей эволюций и генетических операторов.
- Принцип неточности. При росте сложности анализируемой задачи уменьшается возможность построения точной модели. Здесь используется теория нечетких графов.
- Принцип соответствия. Язык описания исходной задачи должен соответствовать наличию имеющейся о ней информации.
- Принцип разнообразия путей развития. Реализация ГА многовариантна и альтернативна. Существует много путей эволюции. Основная задача найти точку бифуркации и выбрать путь, приводящий к получению оптимального решения.
- Принцип единства и противоположности порядка и хаоса. «Хаос не только разрушителен, но и конструктивен», т.е. в хаосе области допустимых решений обязательно содержится порядок, определяющий искомое решение.
- Принцип совместимости и разделительности. Процесс эволюции носит поступательный, пульсирующий или комбинированный характер. Поэтому модель синтетической эволюции должна сочетать все эти принципы.
- Принцип иерархичности. ГА могут надстраиваться сверху вниз и снизу вверх.
- Принцип «Бритвы Оккама» Нежелательно увеличивать сложность архитектуры ГА без необходимости.
- Принцип спонтанного возникновения Пригожина. ГА позволяют спонтанно генерировать наборы альтернативных решений, среди которых с большой вероятностью может возникнуть оптимальное.
- Принцип гомеостаза. ГА конструируются таким образом, что любое полученное альтернативное решение не должно выходить из области допустимых. Операторы в ГА должны позволять получать реальные решения.

С учетом сказанного базисную структуру комбинированного генетического алгоритма для решения ОЗ на графах запишем:

Препроцессор

1. Создание начальной популяции решений для ОЗ на графах.
2. Моделирование популяции (определение ЦФ для каждой хромосомы) на основе приведенных принципов.
3. Отбор «лучших» хромосом (альтернативных решений) для реализации генетических операторов.
4. Пока заданный критерий не достигнут, реализация модифицированных генетических операторов.
5. Редукция, т.е. приведение размера популяции к заданному виду.
6. Рекомбинация родителей и потомков для создания новой генерации.
7. Постпроцессор
8. Реализация новой генерации.
9. Конец работы алгоритма.

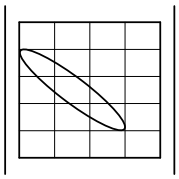
2. Комбинированный генетический алгоритм определения паросочетаний

Пусть G=(X,U) – неорграф. Паросочетанием (ПС) называется подмножество ребер $M \subseteq U$, не имеющих общих концов. Причем каждое ребро $ui \in U$ смежно одному ребру из М. Максимальное паросочетание – это паросочетание М, содержащее максимально возможное число ребер [3,4]. Известно, что число ребер в ПС графа G=(X,U) |X|=n не превышает [n/2], где [n/2] ближайшее большее целое.

Рассмотрим новые эвристики определения паросочетаний в двудольных графах $G=(X1 \cup X2,U)$, $X1 \cup X2$, $X1 \cap X2=\emptyset$.

Пусть задан двудольный граф, показанный на рис.2. В нем можно определить паросочетание M1={(1,6), (3,8)}. В этом графе максимальное паросочетание (МПС) M2={(5,8)(3,7)(2,6)}, выделенное жирными

линиями, показано на рис.3. В этом графе можно построить еще одно МПС: М2'= {(4,8)(3,7)(2,6)}. Для нахождения МПС в двудольном графе будем использовать специальную матрицу смежности R.



Строки матрицы соответствуют вершинам Х1, а столбцы вершинам Х2 . На пересечении строк и столбцов ставится значение 1 или 0 в зависимости от наличия или отсутствия соответствующего ребра. Такая модель требует в 4 раза меньше ячеек памяти для представления матрицы в ЭВМ. Это особенно существенно при анализе графов на сотни тысяч вершин.

Предлагается следующая эвристика. В матрице R ищется диагональ с наибольшим числом элементов. Если таких диагоналей несколько, то выбирается любая. Например, в матрице R диагональ с наибольшим числом элементов имеет вид Д = {(2,6)(3,7)(4,8)}. Следовательно, определено МПС М2' для графа рис.3.

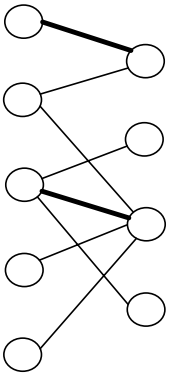


Рис.2. Двудольный граф G

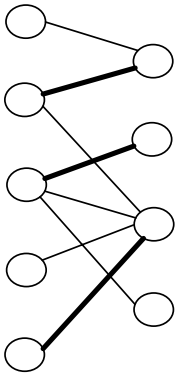


Рис.3. Максимальное ПС

Сформулируем следующую гипотезу. Главная диагональ матрицы R, полностью заполненная элементами, соответствует МПС. Суммарное число единиц соответствует суммарному числу ребер МПС. Каждая единица главной диагонали определяет ребро МПС.

Доказательство следует из способа построения специальной матрицы по заданному двудольному графу, т.к. каждой вершине из Х1 главной диагонали ставится в соответствие одна и только одна вершина из Х2. Отметим, что перед работой алгоритма необходимо упорядочить вершины двудольного графа по возрастанию элементов.

Опишем основную стратегию определения МПС в двудольном графе.

1. Определить вершины подмножеств Х1 и Х2 двудольного графа G.
2. Упорядочим вершины Х1 и Х2.
3. Построить специальную матрицу R и определить в ней главную диагональ.
4. Если главная диагональ заполнена элементами полностью, то построено МПС и переход к шагу 7. Если нет, то переход к 5.
5. В матрице R определить все диагонали и выбрать диагональ с наибольшим числом элементов. Если таких диагоналей несколько то выбирается любая.
6. Выполняется процедура агрегации и преобразования на основе генетических операторов. В результате строится МПС.
7. Конец работы алгоритма.

Например, дан двудольный граф рис.5. Построим специальную матрицу R этого графа.

Согласно алгоритму определяем главную диагональ Д= {(1,6)(2,7)}. Она не заполнена элементами полностью. Определяем другие диагонали: Д1={(1,8)(2,9)(3,10)}; Д2={(2,10)}; Д3={(2,8)(4,10)}; Д4={(5,6)}.

В качестве базовой для определения МПС выбираем Д1 , как диагональ содержащую наибольшее число элементов. Выполняем процедуру агрегации: Д1$\cap$ Д2$\neq$ Ø, Д1$\cap$ Д3$\neq$ Ø, Д1$\cap$ Д$\neq$Ø, Д1$\cap$ Д4=Ø.

R =

6					
1	1				
2	1				
3					
4					
5					

1

6

2

7

3

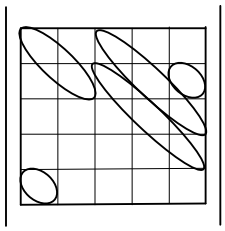
8

4

9

5

Следовательно, элемент (5,6) из Д4 добавляется к Д1 . На этом построение МПС завершено M1={(1,8)(2,9)(3,10)(5,5)} и |M1| = 4. Данная методика может быть применена и для графов, не являющихся двудольными. Для этого необходимо в графе выделить максимально двудольную часть. Для нее определить МПС, а затем на основе агрегации и применения ГО построить МПС для произвольного графа [8].



Например, пусть задан граф G (рис.6). Для его двудольной части МПС равно M={(2,6)(3,7)(4,8)}. Применяя процедуру агрегации получим МПС графа рис.3.37 M={(2,6)(3,7)(4,8)(5,9)}.

Отметим, что для построения МПС можно использовать жадную стратегию. Пример такого построения имеет вид:

1° (1,6)
2° (1,6)(3,7)
3° (1,6)(3,7)(4,8)
4° (1,6)(3,7)(4,8)
5° (1,6)(3,7)(4,8)(5,6)
6° (1,6)(3,7)(4,8)(5,8)
7° (1,6)(3,7)(4,8)(5,9)

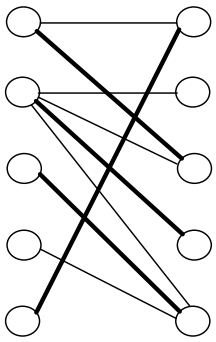


Рис.5. Двудольный граф G

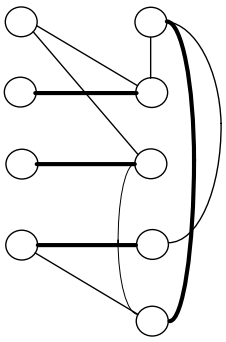


Рис. 6. Граф G

R = 1 2 3 4 5

В результате построено МПС, M={(1,6)(3,7)(4,8)(5,9)}, |M| = 4.

Для графов n<100 можно построить семейство всех ПС и выбрать из них наибольшее.

Например:
1 (1,7)
2 (1,7)(2,6)
3 (1,7)(2,6)(4,8)
4 (1,7)(2,6)(4,8)(5,9)

Построено новое МПС M1={(1,7)(2,6)(4,8)(5,9)}, |M1| = 4.

Применим жадные ГО для графа (рис.6), длина хромосомы в котором n/2=9/2=4,5. Следовательно, L=5. Сгенерируем случайным образом популяцию, как набор хромосом состоящих из различных ребер графа P={P1, P2, P3, P4}.

P1 =(1,6)(2,6)(3,7)(4,8)(5,9),
P2 =(2,6)(4,9)(5,8)(3,7)(1,7),
P3 =(7,9)(5,8)(1,7)(2,6)(5,6),
P4 =(4,9)(5,9)(2,6)(4,8)(3,7).

Применяя оператор сегрегации [5,6] из Р1 выбирается подмножество ребер (строительный блок) СБ1= {(4,8)(5,9)}, из Р2 выбирается СБ2= {(1,7)}, из Р3 выбирается СБ3= {(2,6)}. На этом формирование МПС закончено М={(1,7)(2,6)(4,8)(5,9)}. Временная сложность алгоритма в лучшем случае O(nm), в самом худшем случае O(n!). В среднем для реальных графов O(n2) – O(n3), где n- число вершин, а m- число ребер графа.

Таблица 1

Размер графа, n	Число поколений G	Размер популяции	Вероятность кроссинговера	Вероятность мутации	ЦФ	t, у.е.
100	10	100	30	84	8438	4
100	20	100	30	84	8864	6
100	30	100	30	84	8887	10
100	40	100	30	84	8906	14
100	50	100	30	84	9017	16
100	60	100	30	84	8969	18
100	70	100	30	84	9033	20
100	80	100	30	84	9012	23
100	90	100	30	84	8978	27
100	100	100	30	84	9003	31

Построен комплекс программ решения О3 на графах. Для этого использовались пакеты Borland C++, Builder, Visual C++. Отладка и тестирование проводилось на ЭВМ типа IBM PC с процессором Pentium-IV, AMD Atlon A(0)-1500 с ОЗУ-512Мб. В результате проведенных экспериментальных исследований были получены зависимости значений ЦФ и времени решения от количества поколений приведенные в табл. 1.

Заключение

Для решения оптимизационных задач на графах необходимо учитывать зависимость качества получаемых решений от исходных данных. Поэтому при реализации конкретных генетических алгоритмов, авторы предлагают учитывать влияние внешней среды принципы генетического поиска и знания о решаемых задачах. Разработаны новые эвристики определения максимальных паросочетаний в двудольных графах. Использование принципов агрегации фракталов и поисковых методов позволяют получать набор максимальных паросочетаний за время, сопоставимое с реализацией последовательных алгоритмов.

Литература

Goldberg David E. Genetic Algorithms in Search, Optimization and Machine Learning. USA: Addison-Wesley Publishing Company, Inc., 1989.

Koza J.R. Genetic Programming. Cambridge/MA:MIT Press, 1998.

Кормен Т., Лейзерсон И., Ривест Р. Алгоритмы: построения и анализ. М.: МЦМО,2000.

Кристофидес Н. Теория графов. Алгоритмический подход. М.: Мир,1978.

Курейчик В.В. Эволюционные, синергетические и гомеостатические методы принятия решений. Монография. –Таганрог: Изд-во ТРТУ, 2001.

Курейчик В.М. Генетические алгоритмы и их применение: Монография. –Таганрог: Изд-во ТРТУ, 2002.

Тарасов В. Б. От многоагентных систем к интеллектуальным организациям: философия, психология, информатика. М.: Эдиториал УРСС, 2002.

Курейчик В.М., Курейчик В.В. Фрактальный алгоритм разбиения графа // Известия АН. Теория и системы управления, № 4, 2002, с.65-75.

Информация об авторах

Владимир Викторович Курейчик - e-mail: vkur@tsure.ru
Виктор Михайлович Курейчик e-mail: kur@tsure.ru

Таганрогский государственный радиотехнический университет, Некрасовский 44, г. Таганрог, ГСП-17А, Ростовская обл., Россия, 347928

USING THE SIMULATION MODELING METHODS FOR THE DESIGNING REAL-TIME INTEGRATED EXPERT SYSTEMS

G. Rybina, V. Rybin

Abstract. *Certain theoretical and methodological problems of designing real-time dynamical expert systems, which belong to the class of the most complex integrated expert systems, are discussed. Primary attention is given to the problems of designing subsystems for modeling the external environment in the case where the environment is represented by complex engineering systems. A specific approach to designing simulation models for complex engineering systems is proposed and examples of the application of this approach based on the G2 (Gensym Corp.) tool system are described.*

Keywords: *integrated expert systems, real-time, simulation modeling, object-oriented model, rule, complex engineering systems, electrophysical complex.*

Introduction

The more and more sophisticated character of modem software systems is due to the fact that their architecture comprises a great number of subsystems and components with different functional characteristics, which interact with different groups of users. Because many components are created and developed autonomously, without any provision made for the possibility of joint operation and support of integration processes in the course of evolution of systems, this results, as a rule, in a substantial deterioration in the reliability of such systems. On the other hand, tendencies towards the integration of investigations in different fields, which have prevailed over the last ten years, have necessitated the integration of semantically dissimilar objects, models, methods, concepts, and technologies. This circumstance has inevitably led to the emergence of new classes of systems, such as integrated intelligent systems, integrated expert systems, integrated information systems, integrated manufacturing systems, etc.

Thus, research and development in the field of advanced integrated systems are of particular importance at present. Among the studies in this field, we can mention certain results on the theory and technique of designing integrated expert systems (IES) for static application domains (AD) obtained by the author in the course of conducting the AT-TECHNOLOGY research project (see, e.g. [Rybina,1997]).

However, the integration problems are most conspicuous in the construction of dynamic IES operating in a real-time mode (real-time integrated expert systems (RTIES)), because in this case it is necessary to ensure the following (see [Rybina,1998]): simulation of the external world and its various states; representation, storage, and analysis of time-varying data incoming from external sources; simultaneous temporal reasoning about several distinct asynchronous processes (tasks), support in functioning the inference mechanism under conditions of resource (time and memory) limitations, and other capabilities.

In this connection, special software tools (ST) are needed. These tools must make it possible to design and develop RTIES that can operate in dynamic AD, including the case where the correction of search strategies and knowledge acquisition are possible directly in the process of searching for a solution. The most widely known ST of such a kind are G2 (Gensym Corp.) and RT works (Talarian Corp.).

Over recent years, the author has accumulated certain experience in designing RTIES on the basis of the G2 tools for diagnostics and control problems, such as the control of modem electrophysical complexes [Rybin,Rybina,1998a, Rybin,Rybina,1998b], the diagnostics of complex engineering systems [Rybina,1998], the launch readiness verification of carrier rockets (prelaunch monitoring of carrier rockets) [Rybin,Rybina,1999], radioecological monitoring of areas adjacent to nuclear power plants [Kosterev et al, 1998].

By and large, despite the external dissimilarity of AD, complex engineering systems (CES) were studied. These systems are objects of a technical nature characterized by the following [Rybin,Rybina,1998]: their parameters constantly vary (in real time); they comprise from several hundred to several thousand functionally and structurally interrelated components, subsystems, modules, units, etc.; the diagnostics of these objects can be considered as a specific control process with the goal of determining the technological state of objects at each

current instant (the general task of diagnostics of the object status) and, in addition, the task of fault finding (as a special case of the general diagnostic task); the functioning of these objects is a complex technological process accompanied by a multitude of abnormal conditions, rapid changes in the environment, and the lack of time for decision-making in response to abnormal conditions; a high price is paid for errors made by operators.

Therefore, any RTIES for diagnostics and control of CES (which are discrete and discrete-continuous for the main part) must ensure, in the general case, support for the execution of the following tasks: the dynamic modeling of all processes of functioning the of CES; monitoring the CES operation, detection of deviations from the prescribed regime, prefailure alerting and abnormal condition warning, emergency cut-out, etc.; studying the actions of the operators who control CES and training of personnel; a convenient graphic user interface for monitoring variations in the basic parameters characterizing CES operation, etc.

The architecture of an IES that is designed for real-time operation undergoes substantial changes due to these circumstances, because practically all basic components of a static IES are modified and two new subsystems are added—one of them is intended for the environment simulation, and the other supports the interface with the physical equipment.

The primary emphasis in the present paper is made on the problems of designing one of the most important components of the RTIES, the subsystem for simulation of the environment, which is represented by CES.

1. Statement of the Problem

The problems of real-time modeling of the external environment and its various states are very important in designing any RTIES. The methods and tools of simulation modeling (SM) are the most appropriate here, because the dimensionality of problems being solved and the unformalizability of complex objects and systems for which the RTIES are designed do not allow one to use rigorous mathematical methods.

On the whole, the SM is quite efficient, but, at the same time, it is a rather labour-consuming method and entails a number of problems such as the necessity to provide an adequate description of systems and processes in these systems, a correct interpretation of the results obtained, questions of stochastic convergence of simulation processes, the necessity to overcome difficulties caused by the problem of dimensionality, etc.

As was noted above, the application of methods and tools of SM in RTIES is caused primarily by the necessity for real-time modeling of the external environment and, in particular, to include the corresponding subsystems that adequately reflect all the processes and laws of functioning of the CES in the architecture of RTIES.

In [Rybina,1997] the author consider and solve these problems within the framework of the general-purpose problem-oriented methodology (POM). This methodology is intended for use in static and dynamic applications; it is a set of models, methods, algorithms, and procedures for designing of applied IES.

Since IES are implemented by integrating the methods and technologies of ES with those of conventional programming, the basic problem of integration within the framework of POM should be considered in the following way (see [Rybina,1997]): integration in the framework of IES of different components that execute formalized and unformalized tasks and determine the specific character of functioning of the entire IES (the top integration level); integration (functional, structural, and conceptual) related to basic system designs and concepts of development and design of particular classes of IES and their components (the medium integration level); integration (informational, software, and hardware) related to the technologies, ST, and platforms used (the bottom integration level).

Analyzing the problem of the top-level integration problem, we proposed the classification of IES and introduced the concepts of IES with the superficial and deep integration of components. It was also shown that the methodology for developing simple ES can be used only for designing IES with superficial integration and is completely inapplicable for IES with deep integration. In this case, we propose to apply the approach that improves ES by incorporating functions of a certain component N (where N is a database management system, an application package, SM system, etc.) that are unconventional for such ES, which is an important conceptual basis of the POM.

Here, the problems of integrating ES with SM in the framework of RTIES with deep integration of components are of primary interest, because, in this case, it is necessary to ensure the following (see [Rybin,Rybina,1998a]): the conceptual uniformity of approaches, models, and methods being used; the combination of rigorous mathematical methods of search for solutions with unformalized heuristic methods based on expert knowledge; due regard for the time factor both in the construction of models of AD and in the search for solutions, and other capabilities. The problems of designing RTIES that integrate simulation, conventional ES, as well as other constituent components

of IES, are not clearly understood yet; therefore, the present paper deals with these problems. The modular principle of designing RTIES on the basis of POM, as well as the similarity of certain concepts used in ES and SM, make it possible to integrate these technologies.

Below, we describe the results of experimental approbation of the elaborated models and methods of POM that ensure the integration of ES with SM in designing prototypes of RTIES for problems of control and diagnostics of CES; moreover, we focus mainly on one of the most complicated problems in the construction of RTIES, namely, the control problem for CES.

2. **Construction of Simulation Models of CES**

As noted above, the complexity of CES under study does not allow one to apply rigorous mathematical (analytical and numerical) models when describing these systems. There is no way in which this problem can be solved except by constructing a simulation model (MSM); moreover, preference should be given to the application of methods of the intelligent SM. Now, we consider an example of designing the simulation model MSM for the life-support and survival system (LSS) of the electrophysical complex (EPC) [Rybin,Rybina,1998a].

It should be noted that, due to their complexity, modern EPC are designed to eliminate the effects of subsystems on each other to the greatest possible extent; hence, each particular subsystem can be considered as a CES, and EPC can be treated as the set of CES, i.e., CES = (CES_1,..., CES_N}. Therefore, the control system (CS) for a given object is a hierarchical CS whose control object at the top hierarchy level is the control system of a particular subsystem of the LSS.The simulation model M^{SM} can be subdivided into the model of internal stochastic perturbations (M_{SD}) and the model of the control system (Mcs). To simplify the description of the structure of the simulation model M^{SM}, we assume that the LSS comprises only one subsystem, for instance, the subsystem of vacuumizing. In this case, the simulation model M^{SM} of the life-support and survival system of the EPC will be represented by the simulation model M^{SM} of the subsystem for vacuumizing.

Thus, from the standpoint of SM, the life-support and survival system of the EPC is a discrete-continuous system. In the general case, the set-theoretic model of this system has the following form:

$$M^{SM}_{LSS} = \{M_{CO}, M_{CONTR}, M_{SD}, V^x, V^u, V^E, V^Y, V^Z, S, F^{Y\to U}, F^{XEU\to YZ}\},$$

where M_{CO} is the model of a control object, M_{CONTR} is the model of a controller; M_{SD} is the model of internal stochastic perturbations; $V^x = \{ v_i^x \}$, $i = (1, m)$, is the set of monitored uncontrolled inputs; $V^u = \{ v_j^u \}$, $j = (1, s)$, is the set of monitored controlled inputs of the M_{CO}; $V^E = \{v_h^e\}$, $h = (1, k)$, is the set of stochastic perturbations, $V^Y=\{v_l^y\}$, $l = (1, r)$, is the set of output parameters of the Mco (this set is used in the controller); $V^Z = \{ v_g^z \}$, $g = (1, q)$, is the set of output parameters of the Mco; $S = \{s_c\}$, c = (1, n) is the set of possible (admissible and abnormal, i.e., inadmissible) states; $F^{Y\to U}$ is the function generating the control vector $u(t_{i+1})$ on the basis of the incoming output vector $y(t_i)$; $F^{XEU\to YZ}$ is a function mapping the input of the CO into its output.

Let I denote the input of M_{CO} (V^x, V^u, V^e), and let 0 denote the output (V^y, V^z); then, we have $o(t) = F^{xeu\to yz}(i(\tau), \forall\tau \in [v, t], o^{(k)}(v), k = \varnothing; (n - 1), t)$, where $[o^{(k)}(t)(t=v)= o^{(k)}(v), k = \varnothing; (n - 1)$, are the initial conditions (IC). Thus, at any instant t, the output is a certain function of the input and the IC.

The output of M_{CO} has the dimension $(r + q) \le n$; therefore, $O \subseteq S$, and each particular state of the CO is described by the set of selected (on the basis of different criteria) properties (characteristics), i.e.,

$$C^{\sim} = \{c_1^{\sim},...,c_\lambda^{\sim}\}$$

where $C^{\sim}$ are the valuated properties of the CO. Thus, the set $C^{\sim}$ can be used to describe the set of states $S \subseteq S^{ad} \subseteq S^{ab}$, where S^{ad} is the set of admissible states and S^{ab} is the set of abnormal states.

This formalized representation of the simulation model M^{SM} of LSS of the EPC describes the operation of the entire system, but it is still too abstract for further implementation. In the present paper, in order to make the obtained model more specific and universal, we have used the Rational Rose Real-Time 6.0 CASE tool and the UML with enhancements for the support of real-time system engineering (UML-RT) is used as a language of model designing; these enhancements include the structural elements of the UML-RT-like Capsule and behavioral elements of type of Protocol. The idea is to represent all units using the diagram of classes and to write the diagram of states and transitions for each class. For instance, all units of the equipment making up M_{CO} and M_{CQNTR} are represented in the form of capsules (abstract representations of real-world objects, i.e., the equipment of the LSS of the EPC) with a necessary set of ports (abstract representations of data-transfer channels of real-world objects) through which the messages from other elements (from capsules in abstract declarations, from EPC's equipment in the real world) are incoming.

Since $O \subseteq S$, the output (V^y, V^z) of the M_{CO} is completely described by the attributes of capsules that represent M_{CONTR}; the component (V^u) is described by the capsules of the component representing the generator of stochastic perturbations; and each arrow (V^e) is represented by a message (information) transfer channel, i.e., by a protocol.

For each capsule, a diagram of states and transitions is constructed; the sets of these diagrams for all capsules are defined (represented) by the functions $F^{y \to u}$ and $F^{xeu \to yz}$. Thus, the representation of the simulation model M^{SM} in terms of UML-RT has the following form:

$$M^{SM}_{LSS} = <C, P, S, T, E, R_C, R_p, A>,$$

where C is the set of capsules; P is the set of ports, S is the set of states of capsules; T is the set of transitions; E is the set of events initiating a transition to another state; R_C is the set of relations between capsules, R_p is the set of relations between protocols; A is the mechanism for event tracing and the initiation of transitions (actions) corresponding to an event.

The representation of the model M^{SM} obtained above allows one to pass to its implementation at the level of tools; moreover, no constraints are imposed on the choice of ST for this purpose. In the present paper. the G2 system is used as a toolkit for implementing the simulation model M^{SM} of the life-support and survival system of the EPC and as a software development environment of the entire RTIES. In the G2 object-oriented environment, the above model is unessentially modified, namely: the diagram of classes, which was developed in the Rational Rose RealTime, is turned into an analogous diagram in the G2 environment; the hierarchy of protocols transforms into the hierarchy of connections and relations; the logic of the diagram of transitions and states is described using the G2 rules for the *whenever* construction; the application of these rules allows one to form the mechanism of *event tracing.*

An *event* means that the system is in one of the following a priori known states: a variable, a parameter, or an attribute of an object received a new value; an error occurred when a value was assigned to a variable; a variable lost its significance (the value is no longer significant); an object of some class was created; an object was moved (changed its coordinates) on the desktop; an object passed into an active or dormant state; two objects became related by a certain relation; two objects was connected to each other.

Therefore, M^{SM} in the G2 environment can be represented as the following set:

$$M^{SM}_{LSS} = <CL, O, C, E, RL_E, R_{C1}, R_0>,$$

where CL is the set of system classes; 0 is the set of objects; C is the set of connections between objects; E is the set of model events; RL_E is the set of rules of the event-tracing machine; R_{C1} is the set of relations between classes; R_O is the set of relations between objects.

3. Methodology for Designing Simulation Models of CES

Thus, the following particular methodology for constructing the simulation model MSM of a CES, which is oriented towards use in the G2 environment, was developed within the framework of POM.

1. An AD is analyzed; the basic concepts of the AD, as well as the characteristics and operations of functioning of these concepts, are specified.
2. The abstractions of these concepts are described as classes in G2. The characteristics of concepts of a CES are represented by attributes, and the operations are represented by methods.
3. A powerful visual editor is employed to construct a scheme of the equipment of the real-world CES using the program objects and instances of described classes. Moreover, the interrelations of objects of the real-world CES are assigned by connections and relations.
4. As a result, one obtains the scheme $S = <0, R>$, where 0 is the set of objects of the scheme and R is the set of relations-between these objects.
5. Then, a set of model and temporal events, $E = \{e_i\}$ is constructed. A model event is an a priori specified state, i.e., the set of valuated attributes of one or several particular objects. A temporal event is a priori preset model or real time.
6. The set $D = \{d_j\}$ of actions, which are associated with the set of events, E, is constructed (an action is the totality of methods of objects of the scheme S), as well as the Scheduler (sequence monitor) of their joint functioning (sequential, concurrent, or with time delay). An action has a duration, which is realized in G2 through the use of the *wait for t* construction, where t is the delay time.

7. Based on Items 4 and 5, an event-tracing machine is constructed. This machine is applied for scanning the states of the system and for the initiation of an action associated with an event, when the system passes into a state for which this event is described in the set *E*. This mechanism is implemented in the G2 environment through the use of the rules of the *whenever* construction.

8. The set *B* of initial states is determined. The initialization of the scheme *S* with one of the initial states is carried out by executing constructions of the type *initially*.

4. Example of Application of Methodology for the Construction of Simulation Models of CES for RTIES

We use the example of modeling the LSS of the EPC for an operational prototype of the RTIES [Rybin, Rybina, 1999] to illustrate the elaborated methodology for designing simulation models of CES in the G2 environment. The LSS of the EPC, the charged-particle accelerator in the case under consideration, comprises interrelated subsystems for electric power supply, water supply, vacuumizing, magnet cooling, tunnel ventilation, high-frequency electric power supply, radiation protection, and fire safety.

We restrict ourselves to the consideration of the vacuumizing subsystem designed for the development and maintenance of vacuum in vacuum chambers. When the accelerator is started up, the air is evacuated in two stages; moreover, different types of pumps are used. At the first stage, a low vacuum is developed; at the second stage, the air is evacuated so as to develop a high vacuum. If an abnormal condition arises in the process of operation of the accelerator, i.e., if the pressure is at variance with that existing under high vacuum, then a specific control signal is sent to the emergency system and the vacuum chamber is blocked; the corresponding message is generated and sent to the control desk with the aid of the RTIES (see Fig. 1).

The construction of the simulation model M^{SM} of the vacuumizing subsystem is started simultaneously with the object-oriented analysis of the AD when the basic concepts of the AD are specified and the relations (their type, multiplicity, etc.) between objects are refined. In the case under consideration, one can distinguish the following: the vacuum chamber, the vacuum sector, and the exhaust units (VN1-MG, NEM-300, TMN-200). As has already been noted, the model is designed using, the Rational Rose for RealTime (although this is not obligatory and the designing can be carried out directly in the G2 environment). A diagram of classes is constructed; in our case, we apply the UML-RT, so this diagram is a diagram of capsules; moreover, the relations between capsules can be of the following four types: association, utilization, aggregation, inheritance.

Then, the interacting capsules and the message flows that describe their interaction are specified. For each such interaction, the concept of the message transmission channel and, as a consequence, that of the protocol of transmission of these messages are introduced abstractly. Thus, the diagram of capsules is refined by the diagram of protocols. Further, the sets of attributes that characterize the abstract state of a given capsule as a consequence of the state of the real-world equipment are specified, the conditions for transition from one state to another and the corresponding actions (i.e., the variation of a particular variable, the start-up of the capsule method, the transmission of a message to another capsule) are specified. On the basis of the information thus obtained, a diagram of transitions and states is constructed with the use of the UML-RT tools.

After this, the complete source information for the implementation of the obtained model in the G2 environment is available. In order to generate a list of events, it is sufficient to simply write out the conditions for transitions between states. The actions that are performed when a certain event occurs are also represented in the diagram of states and transitions using Rational Rose for RealTime. It only remains to connect the events and the actions associated with them by *whenever* constructions; these constructions represent one of the types of rules of the G2 system:

<*whenever rule*> :: = whenever <*event declaration*>

[or <*event declaration*>] [and when <*logical expression*>] then <*list of actions*>.

Then, a set of system states is constructed. To simplify the further description of the model, we assume that the vacuumizing subsystem can be in one of the nominal states only, for instance, the system is in the "off" state, which is described by the condition *none of the pumps is powered;* the system is at the stage of maintaining a low vacuum, which is described by the condition *the vacuum is in the range from 10^{-3} to 10^{-2} mm Hg;* the system is at the stage of maintaining a moderate vacuum, which is described by the condition *the vacuum is in the range from 10^{-3} to I0^{-6} mm Hg;* the system is at the stage of maintaining a high vacuum, which is described by the condition *the vacuum is in the range from 10^{-6} to 10^{-9} mm Hg.*

Then, a set of actions is constructed in the G2 environment. The actions in this case are the G2 procedures that initiate and suspend the methods of objects and the methods themselves;

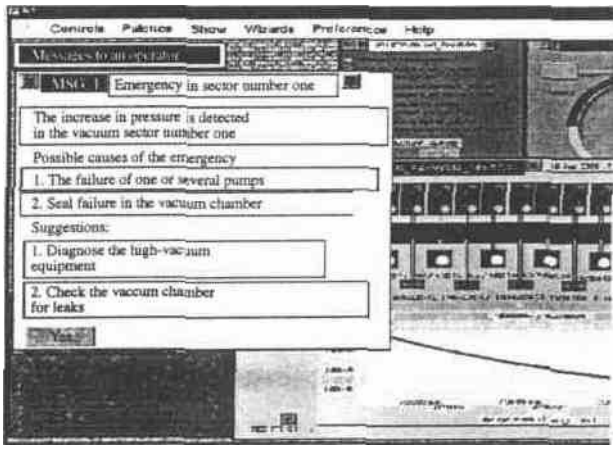


Fig. 1. An example of operation of the prototype of the RTIES for the control of the EPC

for instance, the procedure that initiates all the methods of air evacuation used by pumps of the second type is declared as follows: *power_all_p2(VC),* etc.

And, finally, the event-tracing machine is constructed. In our case, the machine for event tracing and initiation works is as follows:

whenever the pressure of any vac_chamber VC receives a value and when the pressure of VC < 10 and the status of VC /= 2 and the status of VC /= 3 and the status of VC /= 4 then conclude that the status of VC = 1;

whenever the status of any vac_chamber VC receives a value and when the status of VC = 1 then conclude that the status of VC = 2 and start power_all_p2(VC);

whenever the pressure of any vac_chamber VC receives a value and when the pressure of VC < 10e-5 and the status of VC /=4 then conclude that the status of VC = 3;

whenever the status of any vac_chamber VC receives a value and when the status of VC = 3 then conclude that the status of VC = 4 and start power _all_p3(VC).

The initial conditions are determined using the rules of the *initially* type, which invoke the procedures of the system initialization when the applications are started up, i.e., *initially start initial_top_value (top_value) and start initial_bottom_yalue (bottom_value) and start initial_stub posts (stub_posts).*

An example of operation of the current version of the prototype of the RTIES for control of the EPC is presented in Fig. 1. This figure presents the case of detection of an abnormal condition in the operation of the vacuumizing subsystem. This condition is simulated with the aid of the simulation model M_{LSS}^{SM}; it is detected by the RTIES; and, using the rule-based inference, a message indicating possible causes of this condition and suggesting remedies for the trouble is issued to the operator.

Conclusion

Thus, a simulation model M^{SM} of any CES in the G2 environment is an object-oriented model; i.e., it is the set of program objects that simulate the dynamics of the behavior of the real-world CES described with the use of these objects; these latter are interrelated both by data transfer channels and by logical circuits. The real-time simulation process as such is supported by the G2 Scheduler (sequence monitor), which coordinates the processing of model and temporal events; this circumstance substantially facilitates the elaboration of the simulation model M^{SM}.

It should be noted that, if necessary, the MATLAB system or its analogs can be additionally used for modeling complex continuous processes described, in particular, by differential equations of the second or higher order; in this case, integration with G2 is carried out on the basis of tools of the GSI interface (G2 Standard Interface).

Bibliography

[Rybina,1997] G.V.Rybina. Task-Oriented Methodology for Computer-Aided Construction of Integrated Expert Systems for Static Problem Domains. Izv.Ross.Akad.Nauk, Teor.Sist.Upr. 5. 125-137. 1997.

[Rybina,1998] G.V.Rybina. Specific Features and Principles of Design of Integrated Expert Systems for Diagnosis of Complex Industrial Systems. Pribory i Sistemy Upravleniya. 9. 12-16. 1998.

[Rybin,Rybina,1998a] V.M.Rybin., G.V.Rybina. G2 Real-Time Expert System for Control of Electrophysical Complex. Proc. 2nd IMACS Int. Multiconf. CESA '98. Computational Engineering in Systems Applications, vol.1. Tunisia. 659-663. 1998.

[Rybin,Rybina,1998b] V.M.Rybin, G.V.Rybina. Using the Tools Complex G2 For Control of an Electrophysical Complex. Accelerator and Large Experimental Physics Control Systems. Proc. Int. Conf. on Accelerator and Large Experimental Physics Control Systems, Beijing (China), Zhaanel, J and Daneels, A., Eds.. Cern: Science Press. 107-109. 1998.

[Rybin,Rybina,1999] V.M.Rybin, G.V.Rybina. Dynamic Real-Time Integrated Expert Systems: Analysis of the Experience of 'Study and Development. Pribory Sistemy Upravleniya. 8. 4-8. 1999.

[Kosterev et al, 1998] V.V.Kosterev, E.A.Kramer-Ageev, G.V.Rybina, et al. A Prototype of Real-Time Expert Systems for Radio Ecological Monitoring of Areas Adjacent to Nuclear Power Stations. Proc. VI Nat. Conf. with Int. Participation II'98. Pushchino: RAII. V.2. 433-439. 1998.

Author information

Galina Rybina - Moscow Engineering Physics Institute (State University),Kashirskoe shosse, 31, 115409, Moscow, Russia,Email: galina@ailab.mephi.ru

Victor Rybin - Moscow Engineering Physics Institute (State University),Kashirskoe shosse, 31, 115409, Moscow, Russia,Email: rybin@aie.mephi.ru

ON HANDLING REPLAY ATTACKS IN INTRUSION DETECTION SYSTEMS

A. M. Sokolov, D. A. Rachkovskij

***Abstract**: We propose a method for detecting and analyzing the so-called replay attacks in intrusion detection systems, when an intruder contributes a small amount of hostile actions to a recorded session of a legitimate user or process, and replays this session back to the system. The proposed approach can be applied if an automata-based model is used to describe behavior of active entities in a computer system.*

***Keywords**: intrusion detection, replay attack, probabilistic finite automata, dynamic programming, IDS, PFA, DP*

1. Introduction

Intrusion detection systems (IDS) are aimed at detecting and preventing intrusive activities that were not detected by common security mechanisms of a computer system. These are ill-intended activity of legitimate users, outsider's attacks that have passed through a firewall, the use of stolen passwords, and any other activity that was not prevented by authentication, authorization, or other security subsystems.

Two categories of IDS are usually distinguished – *misuse detection systems* and *anomaly detection systems*. The former make use of traces or templates of known attacks, while the latter build profiles of non-anomalous behavior of computer system's active subjects. Both types of IDS have their advantages and drawbacks. Misuse detection systems are perfect in detecting attacks that match one of the predefined templates. However, in case of an unknown attack or a slight variation of a known one, they usually fail. Anomaly detection systems, on the

contrary, learn to recognize non-intrusive behavior and try to detect deviations from it, i.e., anomalies. They are more suitable to alarm at a previously unseen attack, because, as is hypothesized, hacker's activity is different from that of a legitimate user. However, there always remains a possibility that an attack is detected while there is really no intrusion (*false positive*), or a real intrusion activity is unnoticed (*false negative*).

In this paper, we consider the problem of handling a kind of attacks the majority of IDS are vulnerable to. These are the so-called replay attacks – when an intruder acts as if she is a legitimate user, though still performing some intrusive activity. This can be done by contributing a negligible number of hostile actions to a recorded log of a legitimate user or process.

The rest of the paper proceeds as follows. In section 2, we briefly survey existent approaches to the modeling of behavior of active entities in computer systems. Section 3 explains what is a replay attack, and the main idea of our method. Three types of possible intruder's actions and their handling are described in section 4. In section 5, we make conclusions.

2. Methods of anomaly detection

Usually, four major methods of anomaly detection techniques are distinguished: instance-based learning methods, frequency methods, neural-network approaches, and finite automata methods.

Instance-based approaches are the simplest because they just memorize all seen subsequences of elements from the training data. Then, an audit sequence to be checked is considered anomalous if it contains subsequences not present in the previously memorized set. Examples of this approach are given in [1,2,3]. The authors claim, that because of the restrictions operating systems impose on the variety of possible signals in the audit logs, the data sets of behavior examples will not be too large. Otherwise, it is still possible to implement some of the size-reducing techniques [3].

Frequency-based approaches are a development of instance-based methods. They check whether the statistics gathered about a particular event falls into a predefined interval or is equal to a predefined value. The interval or the value may be estimated from the training data without anomalous traces or set by an expert. This approach was used in one of the early papers that considered the need for IDS [4].

Lately, *neural-network* approaches have emerged in the field of IDS. Usually, perceptron-like networks trained by backpropagation are used [5,6,7]. The input to the network is composed of selected system parameters, and the output may be either an *anomalous/normal* behavior indicator or one of the system parameters. If, during the process of testing, the output of the network deviates from the current value of the parameter it represents, an anomaly is signaled.

Numerous methods infer a structure and parameters of a sequence-processing finite automata. These automata can be nondeterministic, with their structure obtained from analyzing computer programs' sources [8] or from sequences of system calls [9]. Non-deterministic automata can be used to model subjects with a limited behavior (e.g., programs making system calls). Probabilistic automata usually infer their structure from data originating from humans [10,11,12], that are of probabilistic nature (e.g., shell commands). One of the simplest kinds of probabilistic automata that can be used to model sequences of audit events are Markov chains [10,11]. A more powerful type of probabilistic automata are Hidden Markov Models (HMM), which were also tested in the field of anomaly detection [12].

3. Modeling replay-attacks

IDS, and particularly anomaly detection systems, are known to be vulnerable to replay attacks. If an intruder gains access to the system audit logs or is able to eavesdrop user sessions, a replay attack can be carried out. Usually, an intruder replays the recorded session back to the system, with only a few inserted, deleted, or altered elements that serve her aim. Due to the negligible quantity of hostile elements in the sequence, this replayed session will most likely be marked by an IDS as legitimate. Almost every IDS can be fooled in such a way, especially those that mainly rely on frequency-based methods.

In [12], there was an attempt to tackle replay attacks by introducing an upper limit on the possible value of the probability of a session. If the probability of a session is higher (a "too similar" session), it is considered an anomalous repetition of some previously seen session, i.e., a replay attack. Accordingly, sessions with the

probability lower than the limit are not considered as containing replay attacks. This method has some serious flaws. In computer systems, namely in UNIX-based ones, there usually exists a plenty of pseudo-users running in the background – the so-called daemons. Daemons tend to generate highly regular log records, because they usually perform fixed and simple tasks. So, the probability of a daemon's session could evaluate to high values, even in the case when there is no replay attack present. Therefore, marking those sessions as anomalous would result in high false positive error rate. The second drawback is that this method does not allow us to identify added, altered, or deleted elements in a log sequence. Thus, even if we succeeded in detecting an attack, we would not be able to point the intruder's contribution out.

We will further describe our method for detecting and analyzing replay attacks that deals with the above drawbacks.

3.1 Underlying model

Unlike the situation in pattern recognition tasks, observed sequences of audit events are usually not distorted. For example, in speech, the same words manifest themselves differently due to individual characteristics of a speaker or peculiarities of the environment. But in case of computer audit events, we usually observe events their creator meant to create. Hence, it is not obligatory to introduce unobservable states having a meaning of "what really occurred", as in HMM [12], or to use advanced structure-processing methods that handle sequence components possessing a gradual degree of similarity [13,14].

Our method is based on recovering the sequence of states that led to generation of a given log session and comparison with the most probable sequence of states passed to generate the same session with the consideration of intruder's contribution. We take Deterministic Probabilistic Finite Automaton (DPFA) [15] as our basic model. In DPFA, states emit symbols probabilistically, but the next state is uniquely determined by the current state and the symbol. With this case of automata, it is possible to recover the sequence of states that generated given output symbol sequence, provided we know the initial state. A DPFA with observable states having direct interpretation can suffice, e.g., Markov chains with either fixed or variable memory length [16] (applications of Markov models to the anomaly detection task were mentioned in section 2). Then, every state of an automaton will have a sequence of symbols (events) associated with it. Those symbols are observable in the log sequence, just like in the common Markov chains.

Let $M=\langle Q,\Sigma,p_0,p\rangle$ be a DPFA, where Q is a set of internal states, Σ is an output alphabet, p_0 is the initial probability distribution over the starting states, and $p(q^b,\sigma|q^a)$ gives the joint conditional probability of transition from state q^a to state q^b generating output symbol σ. In case of user session logs, Σ is naturally thought of as a set of shell commands or processes a user launches. Q can be associated with Σ^L (Markov chains of fixed order) or with a subset of Σ^L (Markov chains with variable memory length). Let the total number of states in Q be N.

Here we will not be interested in methods of learning automata M with examples of legitimate behavior. We just assume that it has already been built during a learning phase (see [15] for a survey of such learning techniques) and describes behavior of a legitimate user.

Thus, the probabilities $p(q^b,\sigma|q^a)$ are considered to be known. For example, for Markov chains of fixed order L, these probabilities are equal to zero for states q^b whose associated strings s^b are not suffixes of string $s^a\sigma$, where s^a is an associated symbolic sequence of state q^a. In the following, lower indices of states will denote their positions in a sequence, while upper indices will be used to distinguish between different states in Q. The lower index of y has the meaning of position in the considered sequence.

To introduce a notion of the beginning (login) and the end (logoff) of a session, we assume that the automaton always starts from the special state q^α and always ends in the special state q^ω. So, $q_0=q^\alpha$ and $q_{n+1}=q^\omega$, where n is the length of a session. The fact that the initial state q^α is fixed will allow us to unambiguously recover the sequence of states given the output sequence of symbols or events (follows from the definition of DPFA).

The automaton operates as follows: at time step k an output symbol $\sigma_{k+1} \in \Sigma$ and next state q_{k+1} are chosen according to joint conditional probability distribution p. So, the probability of generating session $\boldsymbol{r}$ consisting of output symbols $r_0,r_1,\ldots,r_n$ by a walk over a sequence of states $\boldsymbol{q}=q_0,q_1,\ldots,q_n$ is $p(\boldsymbol{q},\boldsymbol{r})=\prod_{k=0,n} p(q_{k+1},r_k|q_k)$.

3.2 Detecting hackers' input

Usually, it is not easy to obtain examples of intrusive behavior and labeled replay attacks in particular. So, it is rather difficult to train some classifier to distinguish between "clean" and "abused" sessions using their examples.

More likely, only expert knowledge will be available about characteristics of intruder's possible actions (e.g., corruption likeliness and possible number of consequently contributed commands, see section 4).

We will think of an intruder's contribution as of corrupting noise applied to eavesdropped session $\boldsymbol{r}$ and distinguish several types of corruption, the intruder is able to make. The first type is when a replay-session is formed from an original session by only changing commands, insertions and deletions of commands are not allowed. The second type further allows insertions, and the third type includes all types of corruptions – changes, insertions and deletions of commands.

Let us denote the corrupted version of session $\boldsymbol{r}=r_0,r_1,...,r_n$ as $\boldsymbol{y}=y_0,y_1,...,y_n$. In general, m is not equal to n, since the number of elements in the sequence may change due to corruption. Let us assume that we have an already learnt automaton M. Given $\boldsymbol{y}$, we can get $\boldsymbol{q}(\boldsymbol{y})$, the sequence of states through which M passed to generate $\boldsymbol{y}$, because M is a DPFA and we know the initial state q^{α}. If we are not able to accomplish this because of, e.g., the absence of some symbols in the alphabet of the legitimate user, or the absence of transitions between certain states of M, it means we have already detected an attack.

Because of the absence of *a priori* information about legitimacy of a given session, we assume replay attack is possibly present. Our idea of detecting a session containing a replay attack is to replace M with another automaton M^+ (see section 4), that will represent the user's and the intruder's contributions simultaneously. Then, we have to search for the most probable sequence of states $\boldsymbol{q}^{\#}$ of M^+, by a walk over which $\boldsymbol{y}$ could be obtained:

$$\boldsymbol{q}^{\#} = \underset{\boldsymbol{q}\in Q^{+m}}{argmax}\, p(\boldsymbol{q}|\boldsymbol{y}) = \underset{\boldsymbol{q}\in Q^{+m}}{argmax}\, \Pi_{k=0,n}\, p(q_{k+1},r_k|q_k). \quad (1)$$

Here Q^+ is the set of states of M^+. Having found $\boldsymbol{q}^{\#}$, we compare it with the sequence of states $\boldsymbol{q}(\boldsymbol{y})$. If $\boldsymbol{q}^{\#}$ and $\boldsymbol{q}(\boldsymbol{y})$ are not equal, we say that at the states where they differ an intruder had changed, inserted or deleted commands.

As an illustration, consider a series of objects (Fig. 1). An object corresponds to the possible set of states of the automaton at a particular time step and can be in any of N states. The edges between states are assigned weights equal to the probability of transition between them. So, the task of finding the most probable sequence of states becomes the task of finding a path with the largest product of weights in the obtained graph.

The algorithm of detecting a hacker's input can be summarized as follows.

First, find a sequences of states $\boldsymbol{q}(\boldsymbol{y})$ of M that lead to generation of the given log sequence $\boldsymbol{y}$. If this step fails for some reason – signal an anomaly.

Second, obtain automaton M^+ from M taking into account type of corruption and stochastic characteristics of the corruption noise. Note that M is converted into M^+ in such a way that the sequence $\boldsymbol{q}(\boldsymbol{y})$ also exists in M^+. In the structure corresponding to M^+, find the most probable sequences of states $\boldsymbol{q}^{\#}$ through which it passed to generate the considered log sequence.

Third, compare the state sequences $\boldsymbol{q}(\boldsymbol{y})$ and $\boldsymbol{q}^{\#}$ of M^+ to find out whether they coincide. If they don't, report their differences – possible hacker's contribution.

Evaluation of (1) can be done efficiently if we use a dynamic programming (DP) scheme (see, e.g., [17]). It may be fruitful to consider several most probable sequences of states as the sequences to compare with $\boldsymbol{q}(\boldsymbol{y})$. This

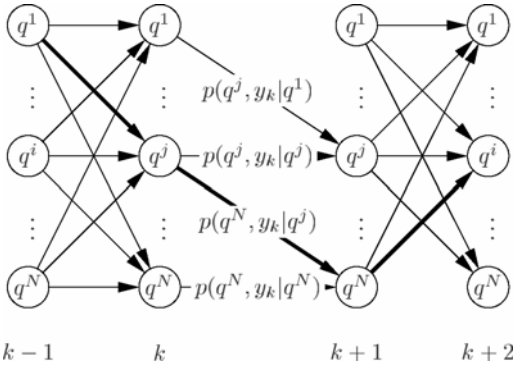


Fig. 1. A part of a graph that represents an automaton, in which the most probable sequence of states $\boldsymbol{q}^{\#}$ is searched. Objects are columns of circles (states). A part of the possible most probable path is given in bold.

option is especially worth considering, if their probabilities are close to each other.

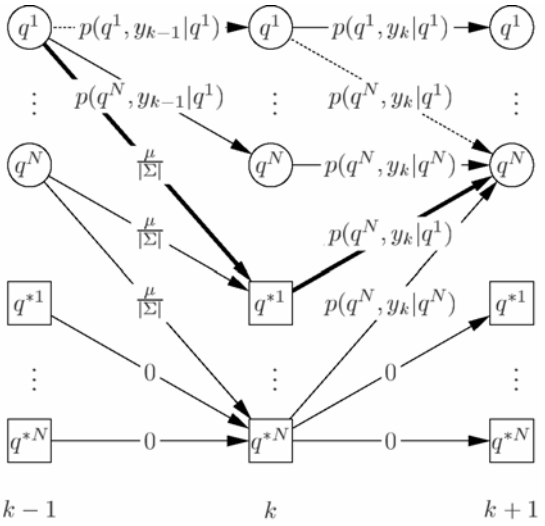


Fig. 2. A part of modified graph to handle insertions. For all i, states q^{*i} are in one-to-one correspondence to states q^i. Note that probabilities of transition from added states $q^{*i} \in Q^*$ to states from Q are equal to those between corresponding states $q^i \in Q$ and the same destination states. The differences between path $\boldsymbol{q}(\boldsymbol{y})$ (dashed) and the most probable path $\boldsymbol{q}^{\#}$ (bold) found in the graph give possible places of insertions.

4. Handling three types of corruptions

4.1 Changes

Let us assume that we have an expert estimation of the probability ρ of changing a command. That is, for each command in a log there is the probability ρ that it will be changed by an intruder and the probability 1-ρ that it won't. For the sake of simplicity, ρ does not depend on the place in the session log where this change occurs, on the current context, and on the command being replaced. We assume y_k that will replace command r_k in a sequence is drawn by an intruder from a uniform distribution over $\Sigma\backslash\{r_k\}$.

Then, in (1) the probability $p(q_{k+1},y_k|q_k)$ becomes dependent on whether symbol y_i coincides with symbol r_k that is necessary to emit to transit to the state q_{k+1} from q_k.

$$p(q_{k+1},y_k|q_k) = p(q_{k+1},r_k|q_k)((1-\rho)\delta_{q_k y_k} + \rho(1-\delta_{q_k y_k})/(|\Sigma|-1)), \quad (2)$$

with $p(q_{k+1},r_k|q_k)$ being the original uncorrupted probability of generating r_k in state q_k and going into state q_{k+1}. We evaluate expression (1) and compare $\boldsymbol{q}^{\#}$ with $\boldsymbol{q}(\boldsymbol{y})$. In the places where they differ, we say that the corresponding commands were changed by an intruder.

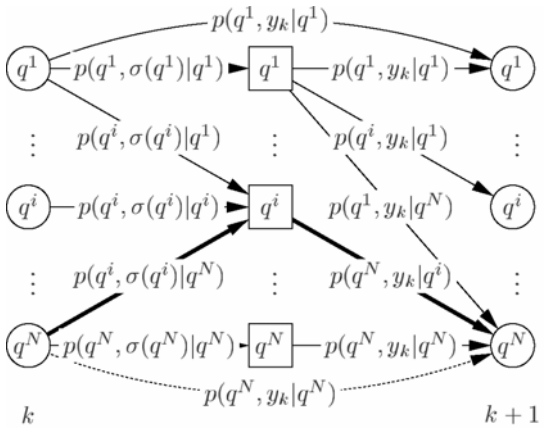


Fig. 3. A part of a graph to handle deletions. If the found most probable sequence of states $\boldsymbol{q}^{\#}$ (bold) passes through added, "jumped-over" objects, it means that the symbols in the log sequence corresponding to those objects were possibly deleted by an intruder. Dashed line gives a part of the sequence $\boldsymbol{q}(\boldsymbol{y})$ recovered from log.

4.2 Insertions

Besides changing commands, we will additionally allow insertions. Let us again assume that we have an expert estimation of the probability μ of inserting an intruder's command at any place in a session. We complement the set of states Q with the same quantity of special states Q^*, having one-to-one correspondence with the states from Q. Thus, the whole set of the automaton's states becomes $Q \cup Q^*$.

The probability of entering any of these additional states from any $q \in Q$ with generating an output symbol $y \in \Sigma$ is set equal to $\mu/|\Sigma|$. The probability of transition to $q^i \in Q$ from some $q^{*j} \in Q^*$ with generating an output symbol $y \in \Sigma$ is set to be equal to $p(q^i,y|q^j)$ – the probability of transition from the state $q^i \in Q$ (corresponding to $q^{*i} \in Q^*$) to $q^i \in Q$ with generating the same symbol y. Transitions between additional states are prohibited. Formally,

$$p(q^{*i},y|\, q^j) = \mu/|\Sigma|, \quad p(q^i,y|\, q^{*j}) = p(q^i,y|\, q^j), \quad p(q^{*i},y|\, q^{*j}) = 0, \quad y \in \Sigma,\ q^i,\ q^j \in Q,\ q^{*i},\ q^{*j} \in Q^* \qquad (3)$$

where $p(q^i,y|q^j)$ is given by the expression (2) that takes changes into account. The rest of probabilities of transition (between states from Q^*) and the whole graph structure (except for added states) remain unchanged (Fig. 2). In Fig. 2, every object corresponds to a command in a considered log including inserted commands.

Passing over a state from Q^* means insertion of a command that appears to be generated from this state. After having searched for the most probable sequence of states using a DP scheme, we check if states from Q^* appear in the obtained sequence $\boldsymbol{q}^{\#}$. If so, symbols generated by these states could be inserted by an intruder.

Equation (3) prohibits transitions between additional states. It means that only one command at each time step can be inserted. If, however, we allow such transitions, several consequent insertions can be handled. This would require additional expert knowledge about the likeliness of neighboring inserted commands. Expressed in the form of probability, this knowledge would participate in the above expressions.

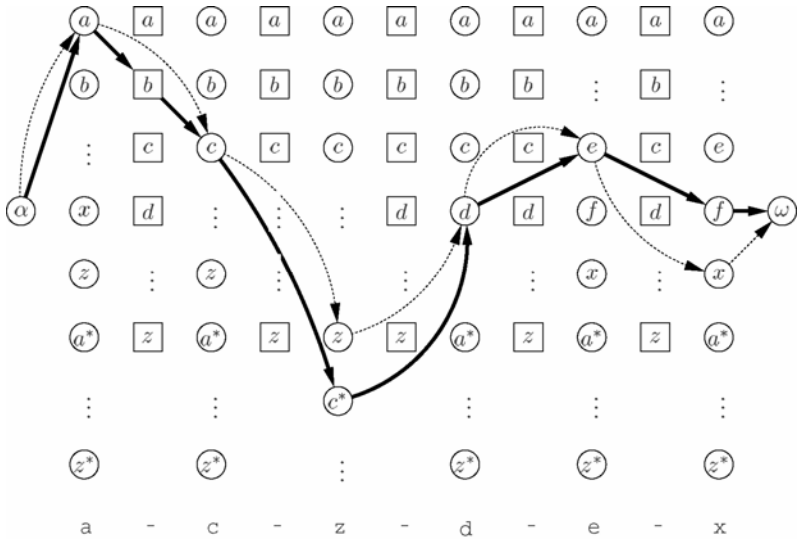


Fig. 4 An illustrative example to detecting different types of corruption. Dashed lines mark the recovered sequence of states $\boldsymbol{q}(\boldsymbol{y}) = \alpha aczdex\omega$ corresponding to the sequence $\boldsymbol{y}=aczdex$. The found sequence most probable sequence $\boldsymbol{q}^{\#}=\alpha abcc^*def\omega$ is given in bold. Circles correspond to the states from $Q \cup Q^*$. Square nodes denote states in objects that were added to handle deletions. We compare $\boldsymbol{q}(\boldsymbol{y})$ and $\boldsymbol{q}^{\#}$ to find differences – *b* could be deleted, *z* – inserted, and symbol *f* was replaced with *x*.

4.3 Deletions

Using deletions as additional means of performing some ill-intended activity is somewhat trickier than using just insertions and changes of commands, but nevertheless possible. Let us suppose that two adjacent deletions do not occur. We change the graph in Fig. 1 to allow connections not only to the neighboring object, but also to jump over one object. These "jumped-over" objects represent deleted components of the observed sequence and are inserted to the graph after each existing object, thus doubling the total number of objects. In Fig. 3, $\sigma(q^j)$ is the last symbol of the string that corresponds to the state q^j. That is, the automaton must emit $\sigma(q^j)$ to change from some q^i into q^j. We can still search for a path in the obtained graph with the largest product of edges' weights, so

calculating expression (1). Again, by comparing $\boldsymbol{q}^{\#}$ with $\boldsymbol{q}(\boldsymbol{y})$ we can find out the deleted elements of the audit sequence. The set of states used to represent objects can be extended as in case of handling insertions. Thus, we will obtain a structure handling all three types of corruption.

4.4 Illustrative example

Consider the illustrative example in Fig. 4. Let Q be a set consisting of symbols {*a*,*b*,*c*,*d*,*e*,*f*,*x*,*z*} and L=1. Then, Q^*={*a*,b*,c*,d*,e*,f*,x*,z*}.* Suppose that session $\boldsymbol{y}$=*aczdex* arrived to be tested. Then, $\boldsymbol{q(y)}=\alpha aczdex\omega$. After creating a corresponding graph, depicted in Fig. 4, we search for the most probable path on it. Suppose the found most probable sequence is $\boldsymbol{q}^{\#}=\alpha abcc^*def\omega$ (bold lines in Fig. 4). The most probable sequence $\boldsymbol{q}^{\#}$ passes through state *c**, what means the corresponding symbol *z* in $\boldsymbol{y}$ was inserted by an intruder and should not be included in the resulting recovered sequence $\boldsymbol{r}$ =*abcdef.* Comparing with $\boldsymbol{y}$=*aczdex*, we obtain that symbol *b* could be deleted, *z* – inserted, and symbol *f* was replaced with *x.*

5. Discussion

We proposed a method that can be applied to handling replay attacks in which a DPFA-based model is used to model legitimate active entity's behavior in a computer system. It allows one to specify the intruder's contributions to legitimate sessions. All three types of corruption (changes, insertions, deletions) can be handled simultaneously, as the changes their handling introduces to the underlying automaton are independent of each other.

It is feasible that an intruder will not insert her commands at the beginning of a session and would try to hide them somewhere in the middle of it, that some command context or type of change will be more advantageous for her. If so, the probabilities ρ and μ may be assumed dependent on the place in an audit sequence, context, commands being replaced or inserted, etc. The search procedures for the most probable sequence of states will not change. An adequate model of adversary that carries replay attacks out (e.g., which type of distortions the attacker uses, feasibility of deletions, whether to allow several consequent corruptions, mixtures of different types of corruptions, and so on) would facilitate implementation of the presented method and increase its performance.

In the proposed approach, an anomaly is signaled if at least one difference between state sequences occurs, but it may be worth considering an option of signaling it after encountering several differences (or differences of particular kind). A definite alarm policy should be determined after taking into account specific knowledge about users, their behavior, system peculiarities, assumed hacker's capabilities, etc.

The difficulty of obtaining an adequate expert estimation of the probabilities ρ and μ, and their dependencies (e.g., on the context) may limit the method's performance and accuracy. If we have data containing replay attacks, the option of estimating these probabilities from those data has to be considered.

An inaccurate estimation of probabilities and model (automaton) M used can lead to higher false positive and/or false negative rates. In case of sudden change of user's behavior (change of tasks, misprints, etc.), this method may also signal replay attacks. But the fact that the method reports the essence of alleged anomaly may help a security officer to make an appropriate decision.

Classification and recognition systems often provide a confidence value along with the result of classification or recognition. In the task of replay attack detection, the usefulness of an analogous empirical coefficient can be considered. As a variant, a ratio R between $P(\boldsymbol{q}^{\#})$ in M^+ and $P(\boldsymbol{q}(\boldsymbol{y}))$ in M, or between $P(\boldsymbol{q}^{\#})$ in M^+ and $P(\boldsymbol{q}(\boldsymbol{y}))$ in M^+, can be proposed. The obtained value of R can be brought to an expert's attention as an auxiliary indicator of an anomaly in the session, or can be used as an automatic threshold filter of anomalous sessions.

Further research and experiments with real data are required to estimate how the proposed approach and its modifications will improve handling of replay attacks.

Bibliography

[1] A. Somayaji. Automated response using system-call delays. In USENIX Security Symposium 2000, pages 185–197, 2000.

[2] S. Forrest, S. A. Hofmeyr, A. Somayaji, and T. A. Longstaff. A sense of self for Unix processes. In Proceedings of the 1996 IEEE Symposium on Research in Security and Privacy, pages 120–128. IEEE Computer Society Press, 1996.

[3] T. Lane and C. E. Brodley. Temporal sequence learning and data reduction for anomaly detection. ACM Transactions on Information and System Security, 2(3): 295–331, 1999.

[4] D. E. Denning. An intrusion-detection model. In Proc. IEEE Symposium on Security and Privacy, pages 118–131, 1986.

[5] J. Ryan, M.-J. Lin, and R. Miikkulainen. Intrusion detection with neural networks. Advances in Neural Information Processing Systems, pages 254–272, 1998.

[6] D. Endler. Intrusion detection: Applying machine learning to solaris audit data. In Proc. Annual Computer Security Applications Conference (ACSAC'98), pages 268–279, Los Alamitos, CA, December 1998. IEEE Computer Society Press. Scottsdale, AZ.

[7] A. Ghosh, A. Schwartzbard, and M. Schatz. Learning program behavior profiles for intrusion detection. In Proceedings 1-st USENIX Workshop on Intrusion Detection and Network Monitoring, pages 51–62, Santa Clara, California, April 1999.

[8] D. Wagner and R. Dean. Intrusion detection via static analysis. In Francis M. Titsworth, editor, Proceedings of the 2001 IEEE Symposium on Security and Privacy, pages 156–169, Los Alamitos, CA, May 14–16 2001. IEEE Computer Society.

[9] C. C. Michael and A. Ghosh. Two state-based approaches to program-based anomaly detection. ACM Transactions on Information and System Security, 5(2), 2002.

[10] B. D. Davison and H. Hirsh. Predicting sequences of user actions. In Predicting the Future: AI Approaches to Time-Series Problems, pages 5–12, Madison, WI, July 1998. AAAI Press. Proceedings of AAAI-98/ICML-98 Workshop, published as Technical Report WS-98-07.

[11] N. Ye. A markov chain model of temporal behavior for anomaly detection. In Proceedings of the 2000 IEEE Systems, Man and Cybernetics, Information Assurance and Security Workshop, pages 171–174, 2000.

[12] T. Lane. Hidden markov models for human/computer interface modeling. In IJCAI-99 Workshop on Learning About Users, pages 35–44, 1999.

[13] E. M. Kussul and D. A. Rachkovskij. Multilevel assembly neural architecture and processing of sequences. In A. V. Holden and V.I. Kryukov, editors, Neurocomputers and Attention: Vol. II. Connectionism and Neurocomputers, pages 577–590. Manchester University Press, 1991.

[14] D. A. Rachkovskij. Representation and processing of structures with binary sparse distributed codes. IEEE TKDE, 13(2): 261–276, 2001.

[15] K. P. Murphy. Passively learning finite automata. Technical Report 96-04-017, Santa Fe Institute, 1995.

[16] D. Ron, Y. Singer, and N. Tishby. The power of amnesia: Learning probabilistic automata with variable memory length. Machine Learning, 25(2-3): 117–149, 1996.

[17] M. I. Schlesinger and V. Hlavác. Ten lectures on statistical and structural pattern recognition. Kluwer Academic Publishers, Dordrecht/Boston/London, 2002.

Author information

Artem M. Sokolov, Dmitri A. Rachkovskij – International Research and Training Center of Information Technologies and Systems; Pr. Acad. Glushkova, 40, Kiev, 03680, Ukraine; e-mails: sokolov@ukr.net , dar@infrm.kiev.ua

Section 5: Knowledge Discovery and Engineering

ALGORITHMS FOR DATA FLOWS[1]

L. Aslanyan, J. Castellanos, F. Mingo, H. Sahakyan, V. Ryazanov

Abstract: *Data analysis is a regular massif task of applied sciences and businesses. A huge number of algorithms were developed for different kinds of data and for particular types of data analysis. Traditional theories work with traditional databases and data structures, although the paradigm of Internet doesn't want to wait, requiring novel technologies, able to work effectively with huge amounts of data, with data flows and uncertainties. The two current research projects, INTAS 397 and 626 are devoted to development of these issues. The paper gives the general statement, current results and examples of these researches.*

Keywords: *Data analysis, data flows, complexity, logic separation.*

Introduction

The novel issues of advanced knowledge-based data analysis algorithmic frameworks are under the development. Traditional theories, such as pattern recognition and combinatorial algorithms, usually work with the given input data sets, delivering the appropriate knowledge in conditions of optimized computational resources – mostly the time and memory [Zhur, 1978]. The nowadays work environment often is the distributed databases and networks. Here the concept "input data set" is dynamic. Where appropriate, data might arrive by portions, with delays and uncertainties. Some algorithms, e.g. statistical, can treat this data, delivering the required estimates of parameters. But the main procedures of intelligent data analysis, which are known in mathematical statistics, pattern recognition theory, logic deduction theories and experimental and theoretical heuristics, are to be revised for the needs of these new conditions:

- Large and increasing input data amounts and dynamic flows (data warehouses, networks, uncertainties),
- Cross-cooperating algorithms and information theories, - converging to the general reference model of intelligent data analysis and data mining,
- The complementary group of tasks – development of newly theoretical postulations (distributed intelligence, accumulative (additive) knowledge scheme, complexity and approximation of algorithms), practical realization (searches as WEB surf, Internet automation, diverse business models over the Internet).

The practical part of research is in developing prototype systems with real applications, particularly, for e-business management and for intrusion detection problems [Pasic, 2000]. The technological components used include mobile and intelligent software agents, neural nets, logic-combinatorial and algebraic recognition, association rules induction and heuristics harmonization. The result is a convergence of technologies into the hybrid technologies of data analysis in conditions, mentioned above: – consequent input data flows by complementary portions, data sizes are very huge, the networked distributed and intelligent analysis provides the new and powerful business mechanisms [Asl, 2001, Gim, 2001]. **Algorithm Incubator** is one more description of research result which means theories and technologies to near-to-market state, providing ultimately leading edge technologies for wealth creation, thereby advancing cross-European competitiveness.

The described problem is studied for several particular models. A good description of these results is surveyed in [Prov, 2000] which discussed the possibility of scaling of inductive algorithms. In part these studies are related to

[1] The research was supported by INTAS 00-397 and 00-626 Projects.

data mining algorithms but some particular cases are also important, e.g. the work of algorithms analysing huge data in a limited operational memory space. Without going deep into the comparisons it is sufficient to note that our case is related to the situation, when some data is available, more data are possible but uncertain, and we have to have a temporal data analysis result in any time ready to be outputted by request.

General Notes and the Simplest Examples

The simplest example of algorithms, working effectively with data flows is e.g., calculation of main values by the series. Given two blocks of numbers: $\alpha_1, \alpha_2, ..., \alpha_m$ and $\delta_1, \delta_2, ..., \delta_n$. Then the overall main value equals $M = \frac{\sum \alpha_i + \sum \delta_j}{m+n}$. If blocks are arriving consecuently, then we may have the main value $M_\alpha = \frac{\sum \alpha_i}{m}$ for the first block in arrival of the second block. So the calculation $M_\delta = \frac{\sum \delta_j}{n}$, and then $M = \frac{mM_\alpha + nM_\delta}{m+n}$ might be preferable of the direct calculation of M. Besides the computational counterpart, it is important, that in some cases and in some applications it is not known or certain, that after $\alpha_1, \alpha_2, ..., \alpha_m$ the new data block will arrive. Thia is why we suppose M_α calculated in arrival of the next block.

More meaningful is the following cluster analysis scheme [Duda, 1973]. Given a set $\Xi = \{x_1, x_2, ..., x_n\}$ of numbers (or vectors), the question is in partitionning it into the c clusters, minimizing the sum of square errors (distances) J_i in clusters. Let us consider a temporal (current) partitionning: $\Xi_1, \Xi_2, ..., \Xi_c$ and the reallocation scheme of search of a local optimum by the given criterion. If an element $\hat{x} \in \Xi_j$ is reallocated to the Ξ_i then the value of the optimization criteria in this part equals to $\sum_{x \in \Xi_i} \left\| x - m_i + \frac{\hat{x} - m_i}{n_i + 1} \right\|^2 + \left\| \frac{n_i}{n_i + 1} (\hat{x} - m) \right\|^2$. The reallocation strategy is based on consolidation of differences of these changes, which may use the simplified calculation $J_i + \frac{n_i}{n_i + 1} \left\| \hat{x} - m_i \right\|^2$, where $n_i = |\Xi_i|$ and m_i is the main value in Ξ_i [Duda, 1973]. Similar and still simple are the formulas of reallocating of a group of elements with elements – numbers or vectors. This algorithm also fits with the input data flow scheme, when new arrived data are to be integrated into a current partitionning (reallocation from outside). The idea is in easy update of the current (temporal) results and constructions, instead of analysing the whole input data set again, in each iteration. It is not evident that there are many algorithms suitable for this re-engineering strategy. For example, the discrete isoperimetry problem and it's solutions are a specific case, not fully extendable in this sense.

Let E^n be the n-dimensional unit cube: $E^n = \left\{ (x_1, x_2, \cdots, x_n) \middle/ x_i \in \{0,1\}, i = \overline{1,n} \right\}$, and let $A \subseteq E^n$. We say that $x \in A$ is an interior point of A, if $S_1^n(x) \subseteq A$, where $S_1^n(x)$ is the unit Hamming sphere of E^n, centered at x. We denote by a the size of A, and by $I(A)$ - the set of all interior points of A. A obeys the isoperimetry property, if $|I(A)| = \max_{B \subseteq E^n, |B| = a} |I(B)|$.

Let us agree to omit the trivial cases $a = 0$ and $a = 2^n$ and then $a = \sum_{t=0}^{k} C_n^t + \delta$, for some k and δ, where $0 \le k < n$ and $0 \le \delta < C_n^{k+1}$. Define the standard placement $L^n = \{x_1, x_2, ..., x_{2^n}\}$ of vertices, in the following way: for any vertices α and β of E^n $\alpha \prec \beta$ in L^n, iff

1) $\|\alpha\| < \|\beta\|$, or

2) $\|\alpha\| = \|\beta\|$ and α lexicographically precedes β (here $1 \prec 0$).

Let L_a^n denotes the initial a-segments of L^n. It is well known [Asl, 1979] that for a prefixed size a the L_a^n is a solution of the discrete isoperimetry problem. It is also known, that the arbitrary solutions are semisperical subsets, which are the slight modifications of the structure L_a^n [Asl, 1979]. So the algorithm of discrete isoperimetry is extendable for the basic solution through L^n, and can't be extendable in all cases, because of the sum of semispheres is not a semisphere [Asl, 1979]. That is, when we are given numbers a_1 and a_2, then the fusion of solutions for a_1 and a_2 into the solution for $a_1 + a_2$ is not evident. The same time, the direct construction of the basic solution for $a_1 + a_2$ is as simple, as the construction of the structure L_a^n.

Logical Separation

Let us follow by the Logical Separation (LS) pattern recognition model [Asl, 1976], which based on implementation of several logically expressed suppositions above the elements of the learning sets. These are some formalisms or additional properties (bias) of classification, expressed in terms of Boolean functions and especially – of the Reduced Disjunctive Normal Forms (RDNF) of Boolean functions.

Let us consider a set of logical variables (binary properties) $\chi_1, \chi_2, \cdots, \chi_n$, and the case of two disjoint classes K_1 and K_2. Let $\beta \in K_1$, $\gamma \in K_2$ and let α is an unknown object in the sense of classification. We say that γ is separated by the information of β for α if $\beta \oplus \gamma \leq \beta \oplus \alpha$ where $\oplus$ the bit-vice mod2 summation is. In simple words, this means that the information difference between β and α is larger than of β and γ (the first includes the second). As a consequence of this assumption we get, that the RDNFs of the pairs of complementary partially defined Boolean functions describe the complete structure of information enlargements, started from the learning set.

Let $L \subseteq E^n$ be the learning set; $L_1 = L \cap K_1 = \{\beta_1, ..., \beta_{l(n)}\}$ and $L_2 = L \cap K_2 = \{\gamma_1, ..., \gamma_{k(n)}\}$. Several structures serve the work of LS algorithms. Let us denote by $\mathrm{N}(\beta_i, L_2)$ the set of all maximal subcubes, containing β_i and - out of the set L_2. $\mathrm{N}(\gamma_i, L_1)$ is the similar structure for the second class. If f is the partial Boolean function, determined as "one" in L_1 and "zero" in L_2, then $N_1 = \bigcup_i \mathrm{N}(\beta_i, L_2)$ (in fact – it's simplification) is the RDNF of f. In terms of supervised classification N_1 is the area of the possible extensions of the learning subset L_1. The intersection $N_1 \bigcap N_2$ is logically reachable from both classes. These are the base structures of LS, which are related to different theoretical and applied problems.

In theories LS is the instrument for optimization [AslZhur, 2001]. It is given a partial Boolean function f. f might be evaluated arbitrarily on the area of indetermination. The aim is to minimize the complexity of the targeted extension function.

In pattern recognition LS is a powerful addition to the "compactness hypotheses"-based models [Zhur, 1978, Asl, 1979]. It is important the role of LS in advanced data mining solutions. LS is an universal interpreter of IREP class of data mining algorithms. While IREP is an approximate scheme through the rules growing and pruning stages, the LS is the exact construction of class separations by rules. We do not need to go now deep into the comparison of these models and it is to check whether the LS constructions are well suited to the data flow analysis algorithms.

The key structure is the set construction (bunch of subcubes) $\mathrm{N}(\beta_i, L_2)$ for an arbitrarily fixed β_i and for L_2. The question is in constructing the $\mathrm{N}(\beta_i, L_2)$ in conditions, when we are given the L_2 in the consecutive blocks $L_2^{'} + L_2^{''}$. The use of traditional algorithms of synthesis of RDNFs is one way. The complexity is evidently high for this algorithms, because of they are of growing type – they are going from vertices to subcubes. The natural way is the pronging approach. This is effective to apply consequently by the input vertices (L_2), starting from E^n. This is a monotone process of breaking E^n, and then - the consecutive results of pronging. When $\mathrm{N}(\beta_i, L_2^{'})$ and $\mathrm{N}(\beta_i, L_2^{''})$ are constructed, then their fusion - into the $\mathrm{N}(\beta_i, L_2)$ is through the intersection

of pairs of subcubes from $N(\beta_i, L_2^{'})$ and $N(\beta_i, L_2^{''})$. If $S^{'} \in N(\beta_i, L_2^{'})$ and $S^{''} \in N(\beta_i, L_2^{''})$, then $S = S^{'} \bigcap S^{''}$ belongs to the $N(\beta_i, L_2)$. This is evident, because of $\beta_i \in S$ and S is out of $L_2^{'} + L_2^{''}$. Now, let S be an arbitrary subcube of $N(\beta_i, L_2)$. First, it is clear that $\beta \in S$. Due to the mentioned monotonity, the intersections $S \bigcap N(\beta_i, L_2^{'})$ and $S \bigcap N(\beta_i, L_2^{''})$ are none empthy. Let $\delta \in S$ is the vertex, opposite to β. δ must be covered by these bunches so that there exist $S^{'} \in N(\beta_i, L_2^{'})$ and $S^{''} \in N(\beta_i, L_2^{''})$ containing both β and δ. The subset, containing both - β and δ have to contain the whole S. This intersection gives the proof of the statement.

The conclusion of the above paragraph is that the $N(\beta_i, L_2)$ may be constructed effectively through the intersections of pairs of elements of $N(\beta_i, L_2^{'})$ and $N(\beta_i, L_2^{''})$.

Conclusion

Data flows, as a special case of input information formation, are known and/but re-estimated by the appearance of Internet and other distributed databases. Data analysis algorithms, mainly constructed for use with fixed data sets, are to be revised for the new conditions. Some of the well known algorithms are easy to apply to data flows. Others are conditional or not optimal. The data mining algorithms, as the approximate branch of the Logical Separation pattern recognition model, are a special case of studies. The huge data sets imply to complex calculations, which is unavoidable, partially. The way is in systematization and it was demonstrated, that the general scheme of Logical Separation is extendable.

Bibliography

[Duda, 1973] C R. O. Duda, P. E. Hart, Pattern classification and scene analysis, Wiley, New York, (1973).

[Asl, 1976] L. H. Aslanyan, The pattern recognition algorithms with logical separators, in “Collection of works on Theoretical Cybernetics”, Computer Center of USSR Academy of Sciences, Moscow, (1976), 116-131.

[Zhur, 1978] Yu. I. Zhuravlev, On an algorithmic approach to the problems of recognition and classification, Problemi Kibernetiki, 33, (1978), 5-68.

[Asl, 1979] L. H. Aslanyan, The discrete isoperimetric problem and the related extremal problems in discrete spaces, Problemy Kibernetiki, 36, Moscow, (1979), 85-128.

[Pasic, 2000] Pasic A., Kulin A., Salmonsen G. and Aslanyan L., Security Policy Adaptation Reinforced Through Agents, International Seminar “Conversion Potential of Armenia and ISTC Programs”, 2-7 October, (2000), Yerevan.

[Prov, 2000] Provost F., Kolluri V., A survey of methods for scaling up inductive algorithms. Kluwer Academic Publishers, Boston, pp. 1-42.

[AslZhur, 2001] Aslanyan L. and Zhuravlev Yu., Logic Separation Principle, CSIT Conference, Yerevan, September 17-20, (2001), 151-156.

[Asl, 2001] Aslanyan L., Castellanos J., Georgiou P., Tsagas G. and Riazanov V., Concurrent heuristics in data analysis and prediction, Pattern Recognition and Image Analysis, vol. 11, No. 1, (2001), 8-10.

[Gim, 2001] Gimenez-Martinez V., Aslanyan L., Castellanos J., Riazanov V., Distribution Function as Attractors for Recurrent Neural Networks, Pattern recognition and image analysis, vol. 11, No. 3, (2001), 492-497.

Author information

Levon Aslanyan – Institute for Informatics and Automation Problems, NAS Armenia; P. Sevak 1 Str., Yerevan-14, Armenia; e-mail: lasl@sci.am

Juan Castellanos - Departamento de Inteligencia Artificial, Facultad de Informática, Universidad Politécnica de Madrid, Campus de Montegancedo s/n, Boadilla del Monte, 28660, Madrid, jcastellanos@fi.upm.es

Fernando Mingo - Departamento de Inteligencia Artificial, Facultad de Informática, Universidad Politécnica de Madrid, Campus de Montegancedo s/n, Boadilla del Monte, 28660, Madrid

Hasmik Sahakyan - Institute for Informatics and Automation Problems, NAS Armenia; P. Sevak 1 Str., Yerevan-14, Armenia; e-mail: hasmik@ipia.sci.am

Vladimir Ryazanov - Computer Center of Russian Academy of Sciences, Vavilov, 40, Moscow, rvv@ccas.ru

PARONYMS FOR ACCELERATED CORRECTION OF SEMANTIC ERRORS *

I. A. Bolshakov, A. Gelbukh

***Abstract**: The errors usually made by authors during text preparation are classified. The notion of semantic errors is elaborated, and malapropisms are pointed among them as "similar" to the intended word but essentially distorting the meaning of the text. For whatever method of malapropism correction, we propose to beforehand compile dictionaries of paronyms, i.e. of words similar to each other in letters, sounds or morphs. The proposed classification of errors and paronyms is illustrated by English and Russian examples being valid for many languages. Specific dictionaries of literal and morphemic paronyms are compiled for Russian. It is shown that literal paronyms drastically cut down (up to 360 times) the search of correction candidates, while morphemic paronyms permit to correct errors not studied so far and characteristic for foreigners.*

***Keywords:** error correction, correction candidates, semantic errors, malapropisms, paronyms, literal paronyms, morphemic paronyms, paronymy dictionaries.*

Introduction

Various errors made by authors in theirs natural language texts can be categorized as follows:

- Orthographic errors transform a correct word to a senseless letter string, e.g., *interesting vook* (instead of *book*);
- Syntactic errors transform one real word to another, thus violating syntactic correctness of the texts concerning agreement of adjectives with their ruling nouns in gender and/or number in Slavic or Romance languages, e.g. Rus. *маленький мальчики* 'little$_{SG}$ boys$_{PL}$' instead of *маленькие*$_{PL}$; grammatical cases of the valence dependent noun in Slavic languages (Rus. *довольный правительству*$_{DAT}$ lit. 'content to the government' instead of *правительством*$_{INS}$), personal verb forms (*he go* for *goes*) (SG, PL are singular and plural; DAT, INS are dative and instrumental case), etc.
- Semantic errors leave the text orthographically and syntactically faultless, but make it senseless or absurd (*inculpation period* for *incubation period, massy migration* for *massive migration,* etc.).

All modern text editors have tools for error detection. Purely orthographic errors are detected always, and lists of potential correction candidates are given out similar to the suspicious string in letters and/or sounds. Grammatical errors are not always detectable because of deficiencies of modern syntactic analyzers, and variants of syntax corrections are rare so far. Semantic errors are not detected at all.

Meanwhile, methods are already proposed of how to correct one type of semantic errors. For this type, one real word is replaces by another "similar" to the intended one in literal or sound content. If such errors violate semantic correctness of texts, they are referred to as malapropisms.

In [Hirst & St-Onge, 1998; Hirst & Budanitsky, 1998] detection and correction of malapropisms use paradigmatic semantic links between words occurring in adjacent paragraphs and sentences. These are links between direct word repetitions, a word and its hyperonym (*appliance* Vs. *vacuum cleaner*), a part and the whole (*steering wheel* Vs. *car*), etc. For several languages, the links are recorded in thesauri, among which EuroWordNet is well known [Vossen, 2000]. For example, the replacement of *wheel* by *weal* semantically isolates *weal* from words *car*, *brakes* or *gas* within a text.

In [Bolshakov, 2002] malapropism processing uses syntagmatic links between words in a sentence. Malapropisms destroy stable syntactically linked and semantically admissible combinations of content words (=collocations). E.g., *massive migration* is collocation, whereas the syntactically correct *massy migration* is not, cf. [Bolshakov & Gelbukh, 2001, 2002]. Thus malapropisms make some content word(s) in a sentence semantically isolated concerning collocations.

* Work done under partial support of Mexican Government (CONACyT, SNI), IPN (CGPI, COFAA), and RITOS-2.

For any method of malapropism detection, a generator of correction candidates is necessary. They should be somehow "similar" to the intended words. Such generation is analogous to candidate search for orthographical errors but it differs in the rational search strategy.

Indeed, word forms of natural language are rare interspersions in the space of literal strings. For approximate evaluation of this rarefaction, let us take into account that in such highly inflexional language as Russian there exist ca 1.2 million of different word forms, whereas in low inflexional English, say, four times less. To calculate the number of all possible strings above a given alphabet of A letters, suppose their length equal to the mean length L of real words in a corresponding dictionary. Then the total string number equals A^L, i.e. $32^9 \approx 3.5*10^{13}$ in Russian and $26^8 \approx 2.1*10^{11}$ in English. This means that in Russian a real word form contrasts 29 millions of senseless strings, while in English, contrasts 700,000. The change of the mean length of word form in a dictionary to the mean textual value decreases this contrast, still leaving it striking.

Would word forms as letter strings be absolutely stochastic in structure, the probability to meet two forms at a short distance were inconsiderable. In fact, words are built of few thousands of radixes and even fewer prefix and suffix morphs (they are few hundreds in the whole functional morphemarium). Some semantic and morphonological restrictions are imposed on the sets of radixes, prefixes, and suffixes, since not all combinations are reasonable and not all reasonable ones are utterable.

Just this circumstance facilitates the candidate search for replacement of one real word by another. Whereas for an orthographical error a wrong string can be arbitrary and the task to gather beforehand, for each string, literally close real words seems impractical, the environments of a real word form, as our evaluations show, contains only few other real words. Hence, the close words can be gathered for each word that has them. Being put in a special dictionary, they could be used for malapropism correction, to cut down the search of candidates. Indeed, for correction of one-letter error in a string with the length L, it is necessary $A(2L+1)+L-1$ tries, that for a word of nine letters equals 616. For two-letter errors already ca. 360,000 tries are necessary. In the same time, forehand gathered one-letter-apart candidates are numbered few units, for two-letter-apart ones, numbered few tens. For words that are not in the dictionary of substitutes, the candidate search is unneeded, and this also cuts the search.

This work has the objective to classify semantic errors in some detail and to propose for malapropism correction dictionaries of paronyms, i.e. of words similar to each other in some specific sense. Paronyms can be introduced of the following intersecting types:

- ***Literal paronyms*** [Гусев & Саломатина, 2000, 2001] differ in few letters, so they are within easy distance in the space of letter strings, e.g., Rus. *ожижать* 'to liquidize' Vs. *ожидать* 'to wait,' *рок* 'doom/rock' Vs. *срок* 'period.' They are intended for correcting errors characteristic for careless and/or poorly literate persons.
- ***Sound paronyms*** differ in few sounds, so they are within easy distance in the space of phonological records of speech, e.g., Rus. *проектировать* 'to design' Vs. *проецировать* 'to project'). They are indispensable for poorly literate persons.
- ***Morphemic paronyms***, known in Russian lexicography as paronyms proper [Бельчиков & Панюшева, 1994], have the same radix, pertain to the same part of speech (POS), and differ only in prefixes and/or suffixes. E.g., *sens-ible* Vs. *sens-itive* differ in one suffix; *re-volu-tion Vs. in-volu-tion,* in one prefix; *sens-ation-al* Vs. *sens-itive,* in two suffixes. Such paronyms can be close in the space of strings of morphemic symbols. They are important for poorly educated native speakers and for foreigners.

This work reports on compiling Russian dictionaries of one-letter and morphemic paronyms. The dictionaries' fragments and general statistical parameters are given. Literal paronyms cut the search trials by approximately 360 times, while morphemic paronyms permits to quickly detect the errors not yet discussed anywhere but really occurring in texts and speech.

Sources of semantic errors and their effect

Let us classify semantic errors against their sources, giving minimal contexts.

1. Random error directly giving a real word. This could occur by the following reasons:
   - A writing slip immediately gives another real word, e.g. Rus. *испытательный рок* 'trial rock' instead of *срок* 'period.'

- A slip gives senseless string that is falsely "corrected" based on a spellchecker menu, since the author took incorrect candidate among those proposed by text editor. If we enter Rus. *испытательный* ***мрок***(?), the menu of spellchecker will contain for the highlighted string the items *мирок, мрак, прок, рок, срок*, *урок,* along with some non-nouns*,* and the careless author can select a wrong item.
- A correct but very rare word is entered, for which the spellchecker contains one or more alternatives. For example, in the sentence *Ethology of these animals is not studied* spellchecker will propose to replace *ethology* by the more known *etiology* or *ethnology,* and the author can hastily accept such corrections.
- An entered rare word is automatically corrected by a special utility of autocorrection embedded in the text editor. In this case the user transfers a power to the software to make some amendments without any consultations.

2. Ignorance or imprecise knowledge of the intended word, so that instead of it a different word is entered similar to the intended one in sound, e.g. *scientific hypotenuse* instead of *hypothesis*.
3. Imprecise knowledge of meaning for words with the same radix (which really can have the same semantic components), e.g. *sensual news* instead of *sensational news.*
4. Wrong facts or incorrect logic of reasoning transferred in the text. This rarely implies an error in one word, and if so, the resulting word frequently differs from the correct one: *His mother died in* ***infancy*** (for *youth*?); *Hendel was* ***half*** (for *partially*?) *German,* ***half*** *Italian, and* ***half*** *English*. Every human (but not a computer) knows that if a female died in infancy she had no children; that no dividable entity can have three halves, etc.

Hereafter, we deal with the errors of the types 1 to 3. In contrast to errors of the type 4, they violate purely linguistic knowledge on how to commonly use words within the same text. The textual word proved to be:

- similar to the intended one in letters, sounds or morphs,
- preserving syntactic correctness of the utterance, and
- essentially deforming its meaning.

Just such errors are called malapropisms [Encyclopædia, 1998]. Linguistic knowledge is violated by them in the aspects of :

- Syntagmatic semantic links in the texts. The resulted word combinations are not collocations but are syntactically correct. The examples were given above (except of p. 4). More examples are: *polling* ***company*** (for *campaign*); ***hysterical*** (for *historical*) *center;* ***dielectric*** (for *dialectic*) *materialism*; *travel about the* ***word*** (for *world*); *equal* ***excess*** (for *access*) *to school.*
- Paradigmatic semantic links in the texts. Here is an example fit for a single sentence: *Total* ***garniture*** (for *furniture*) *was ruined*: *tables, chairs, armchairs*. Tables, chairs, and armchairs really are related to furniture (not of garniture!), and this is also linguistic knowledge: interrelation of parts and the whole. However, furniture never form collocations with tables, chairs, and armchairs.

The task of candidate search is the same for both type of violation of linguistic knowledge.

Literal paronyms

One literal string of the length L can be formed from any other with the series of editing operations [Kashyap & Oomen, 1981; Mays *et al.*, 1992; Wagner & Fisher, 1974]. Let us take strings under an alphabet of A letters. Elementary editing operations are: replacement of a letter with any other letter in any place within source string ($(A-1)L$ options); omission of a letter (L options); insertion of a letter ($A(L+1)$ options); permutation of two adjacent letters ($L-1$ options).

The string obtained with any of $A(2L+1)+L-1$ operations mentioned, is at the distance 1 from the source string, i.e. on the sphere of radius 1 in the string space. Making another elementary step off, we form a string on the sphere with radius 2—with regard to the source one, etc. Points obtained with minimum R steps are on R-sphere, points of r-spheres with $r < R$ and the source point are not here. Among previous examples,

- *word* Vs. *world, ethology* Vs. *etiology, ethology* Vs. *ethnology* are at the distance 1,
- *hysterical* Vs. *historical, dielectric* Vs. *dialectic, excess* Vs. *access, garniture* Vs. *furniture* are at the distance 2,
- *company* Vs. *campaign*, *massy* Vs. *massive, sensible* Vs. *sensitive, hypotenuse* Vs. *hypothesis* are at the distance 3 or more.

Though the mean distance between word forms is large in any language, they proved to be disposed in clusters. Firstly, such clusters contain elements of morphological paradigms of various lexemes, word forms within them being usually distanced 0 to 3 from each other. Just such a cluster is lexeme, and one of the composing forms is its dictionary name. Secondly, paradigms of various lexemes with similar morphs can be close to each other, sometimes even with intersection.

For our purposes, the paradigm pairs with the same number of elements and correlative elements at the same distance are of interest. E.g., all four elements of paradigms of Eng. verbs *bake* and *cake* differ in the first letter only. Let us call such paradigms parallel. If the distance equals 1, let us call them close parallel.

Thus, any element $\lambda(\chi)$ of the paradigm of λ (χ is a set of intra-lexeme coordinates, i.e. morphological characteristics selecting a specific word form) can be obtained from the correlated element of the parallel paradigm by use of the same editing operator $R_i()$, where i is cardinal number of the operator in an effective enumeration of such operators. Then the relation between dictionary names (they correspond to $\chi = \chi_0$) and specific word forms of parallel lexemes can be represented by the proportion

$$\lambda(\chi) : \lambda(\chi_0) = R_i(\lambda(\chi)) : R_i(\lambda(\chi_0)). \quad (1)$$

The formula (1) means that, for any suspicious form $\lambda(\chi)$ in text, it is necessary to find its dictionary form $\lambda(\chi_0)$, and, if a close parallel $R_i(\lambda(\chi_0))$ for it exists, $R_i(\lambda(\chi))$ should be tried as a correction candidate. For such try, the syntactic correctness pertains as a rule, and some try can correct the error.

The parallelism permits to unite sets of word forms, storing in the dictionary only one their representative, i.e. dictionary name of lexeme. However, strictly parallel paradigms are not so frequent in highly inflectional languages. More usually the parallelism between subparadigms can be found. As such subparadigms, it is reasonable to take grammemes corresponding to fixed combinations of characteristics χ.

For example, noun lexemes of European languages have grammemes of singular and plural. They play the same role in a sentence but differ in the sets of collocations they can be in. The division by grammatical number permits to describe easier Slavic declension and well serves for our purposes. E.g., the subparadigms of singular for Russian *метель* ‘blizzard’ and *мотель* ‘motel’ are not parallel, whereas they do—in plural.

Russian verbs have grammemes of personal forms (we join the infinitive to them), of active and passive participles in all grammatical cases, and of gerund. These grammemes differ in their role in a sentence, so that their separate use keep syntactic correctness of text after the substitution. It is also reasonable to divide each Slavic verb grammeme to its perfect and imperfect aspects, morphonologically rather different.

Each grammeme has its own dictionary name, e.g., a participle is represented by the singular form of nominative case. For the dictionary names and specific forms, the formula (1) pertains. Note that it is not obligatory to require strict parallelism within whole grammemes. E.g., formula (1) applied to Rus. *метры* ‘meters’ и *меры* ‘measures’ fails in genitive case. However such failed tries are not too burdensome.

The idea to divide morpho-paradigms to grammemes is not taken at random. The CrossLexica system elaborated by authors [Bolshakov & Gelbukh, 2001] operates just with grammemes, and paronyms dictionaries under questions are oriented primarily to systems of this kind.

Let us call ***literal paronyms*** any two grammemes that:

- are of the same part of speech;
- concern to the same grammeme type, e.g., both are participles;
- have (close) parallel forms; and, only for nouns,
- have the same gender in singular or are both plural.

With such definition, we have searched close parallel literal paronyms among rather frequent content Russian words. The pairs with at least one member being functional word (pronoun, preposition, conjunctions, etc.) were omitted. A large preliminary version of dictionary was compiled first, and then a special utility proofreads this version for repetitions, omission of inverted pairs, larger distances, wrong orders, etc. Note that in [Гусев & Саломатина, 2000, 2001] the same task has been performed for lexeme names, thus giving less information (see above).

In the current version, there are more than 6,000 paronym groups each having a item-head grammeme to be replaced and the rest grammemes as substitute candidates. The mean number of candidates stably equals 2.25, while the mean name length is 6.75.

Functional words, the shortest in any language, were excluded. Nevertheless, the mean word length in our dictionary proved to be two letters shorter than the mean dictionary length value. So grammemes in our dictionary are seemingly the shortest among the content words, and probably the most frequent among them.

Below, we give a fragment of our dictionary. Note that homonyms like *болеть*1 'to be ill' Vs. *болеть*2 'to ache' or *белки*1 'squirrels' Vs. *белки*2 'proteins' enter separately, but the group for one of them does not include the others. The number of candidates varies from 1 to 12. The maximum number is for the shortest words, i.e. of three letters.

бездомный	белеть	белка	белки2
бездонный	белить	булка	балки
бездумный	болеть1	елка	бели
бездумный	болеть2	челка	булки
бездомный	велеть	щелка	челки
безумный	мелеть	белки1	щелки
безумный	белея	балки	
бездумный	болея	бели	
бекон	мелея	булки	
бетон	белить	челки	
	белить	щелки	
	делить		

The main gain in candidate search is reached owing to looking only candidates given in our dictionary. Using the total number of tries for a 9-letter Russian word, we get the gain coefficient $G_1 = 616/2.25 = 274$.

ClossLexica contains ca. 100,000 one-word grammemes. Even if after further replenishments the total number of groups would reach 6500, this will be only 6.5% of the whole systemic dictionary. Nevertheless, the revealed paronyms are supposedly the most frequent among content words. With the reasonable assumption that the rank distribution of all words in systemic dictionary conforms to Zipf law, these paronyms cover approximately 80% of all word occurrences in texts, and we have the additional gain coefficient $G_2 = \ln 100000 / \ln 6500 = 1.31$ owing to that all other 93,500 word are ignored in the candidate search. The global gain is $G_1 \times G_2 \approx 360$.

Morphemic paronyms

Several errors of a different nature were demonstrated above: *massy* Vs. *massive, sensible* Vs. *sensitive, revolution* Vs. *involution*. They are of the same POS and have the same radix (*mass-, sens-, -volu-*). In Russian linguistics, only this similarity is called paronymy. Confusions of morphemic paronyms are usual errors, especially for foreigners. For example, it is rather difficult to explain to them how to use Rus. paronyms *вислый* 'slouching', *висящий* 'hanging,' *висячий* 'bangled,' and *повисший* 'flagging' that differ only in one suffix and one prefix.

We have gathered morphemic paronyms into groups with the following additional requisites:

- Grammemes are taken as units of the dictionary, so that, e.g., *бок* 'side' and *бока* 'sides' are put into the same group;
- Grammemes of participles are considered as adjectives;
- All grammemes with homonymous radixes are pu to the same groups, e.g., adjectives *бур-ный* 'roaring,' *бур-овой* 'boring,' and *бур-ый* 'brown';
- Homonymous lexemes are given in the groups separately, however none of them can replace another;
- Two-radix words are involved, one radix considered as the radix proper and another as the so-called suffixoid or a prefixoid. The negation *не* is a common prefix, the inseparable reflexive particle *–sja* is considered as suffix after the ending.

All in all, a morphemic paronym can be represented as a string $P_1...P_mRS_1...S_nE$, where $P_1,...,P_m$, m = 0, 1..., are symbols of prefixes; R is radix: $S_1,...,S_n$, n = 0, 1..., are suffixes; E is ending. The distance between paronyms within a group is measured by the number of elementary editing operations in the space of morphemic symbol strings. For example, *+отеч-еств*о* 'homeland' Vs. *отч-еств*о* 'patronym' and *+бед*а* Vs. *+бед*ы* are at the distance 0, *+волос-ат*ый* Vs. *волос-ист*ый*; *вы-нос* Vs. *из-нос*; *эффект-ив-н-ост*ь* Vs *эффект-н-ост*ь* are at the distance 1, *юнош-еск*ий* Vs. *юн*ый*, *гриб-н*ой* Vs. *гриб-к-ов*ый* are at the distance 2. Here the sign

‘+’ initiates a radix; ‘–‘ a prefix or a suffix, ‘*’ an ending. The differences in endings are ignored, since inflexional class is implied by POS and the previous suffix, and specific ending is different for each element of a grammeme.

Our dictionary of morphemic paronyms contains now 1120 paronymy groups with the mean length 5.65. A group element has on an average 1.4 paronyms at the distance 0 or 1. Summarize ‘all-to-all’ links in all groups at any distances, the total link number is up to 55,000, i.e. approximately 49 links within each group. Following is a fragment of the morphemic dictionary:

```
+бег*              +бег-ающ*ий            +бед*а                  +бед-н-еющ*ий
  +бег*а             +бег-л*ый              +бед-н-ост*ь            +бед-н*ый
  +бег-л-ост*ь       +бег-ов*ой             +бед-н-от*а             +бед-ов*ый
  +бег-ств*о         +бег-ущ*ий             +бед-ств-енн-ост*ь      +бед-ств-енн*ый
  +бег-ун*           -при+бег+ающ*ий        +бед-ств-и*е            +бед-ств-ующ*ий
  +бег-ун-ок*        -при+бег+ну-вш*ий      +бед-ств-и*я            -о+бед-н-евш*ый
  +бег-ун*ья         -раз+беж-авш*ий-ся     +бед-ств-ован-и*е       -о+бед-н-енн*ый
  -на+бег*           -с+бег-ающ*ий          +бед*ы
  -при+беж-щ*е       -с+беж-авш*ий          -о+бед-н-ени*е
  -про+бег*          -у+бег-ающ*ий        +бед-н*еть
  -про+беж-к*а       -у+беж-авш*ий          +бед-овать
  -раз+бег*                                 +бед-ств*овать
  -у+беж-ищ*е                               -о+бед-н*еть
```

The search of morphemic errors is cut down by the same ways as for literal errors. If the suspicious word is in the dictionary, only its co-members are taken at the distant 0 or 1 to match. If the textual word is not available in the dictionary, no candidate of morphemic type is searched. We cannot compare our method with others quantitatively, since the latter do not exist. Indeed, the letter distance between morphemic paronyms is usually so high that their direct search in the literal space is absolutely impractical.

Conclusion

It is argued that correction of some semantic errors (namely, malapropisms) is possible by the use of paronyms, i.e. of words similar to each other in letters, sounds or morphs. It is proposed to compile paronymy dictionaries of three types beforehand. Literal paronyms essentially cut the search of correction candidates. Morphemic paronyms permit to quickly correct errors not studied so far and specific for foreigners. Russian dictionaries are already created—for literal and morphemic paronyms. The compiling of sound paronyms is the task for the future.

Bibliography

[Бельчиков & Панюшева, 1994] Бельчиков, Ю. А., М. С. Панюшева. Словарь паронимов современного русского языка. М.: Русский Язык, 1994.

[Bolshakov & Gelbukh, 2001] Bolshakov I. A., A. F. Gelbukh. A Very Large Database of Collocations and Semantic Links. In: Bouzeghoub et al. (eds.) Natural Language Processing and Information Systems. Natural Language Applications to Information Systems. Lecture Notes in Computer Science No. 1959, Springer, 2001, p. 103-114.

[Bolshakov & Gelbukh, 2002] Bolshakov, I. A., A. Gelbukh. Word Combinations as an Important Part of Modern Electronic Dictionaries. Procesamiento del Lenguaje Natural (Spain), No. 29, Sept. 2002, p. 47-54.

[Bolshakov, 2002] Bolshakov, I. A. Detección y Corrección de Malapropismos en Español mediante un Sistema Bietapa para Comprobar Colocaciones. Memorias del XI Congreso Internacional de Computación “Avances en Ciencias de la Computación e Ingeniería de Cómputo” CIC’2002, noviembre 2002, CIC-IPN, México, v. II, p. 303-313.

[Encyclopædia, 1998] The New Encyclopædia Britannica. Micropædia Vol. 7. Encyclopædia Britannica, Inc., 1998.

[Гусев & Саломатина, 2000] Гусев, В. Д., Н. В. Саломатина. Электронный словарь паронимов: версия 1. Научно-Техническая Информация (НТИ), Сер. 2, № 6, 2000, с. 34-41.

[Гусев & Саломатина, 2001] Гусев, В. Д., Н. В. Саломатина. Электронный словарь паронимов: версия 2. Научно-Техническая Информация (НТИ), Сер. 2, № 7, 2001, с. 26-33.

[Hirst & St-Onge, 1998] Hirst, G., D. St-Onge. Lexical Chains as Representation of Context for Detection and Corrections of Malapropisms. In: C. Fellbaum (ed.) WordNet: An Electronic Lexical Database. The MIT Press, 1998, p. 305-332.

[Hirst & Budanitsky, 1998] Hirst, G., A. Budanitsky. Correcting Real-Word Spelling Errors by Restoring Lexical Cohesion. Computational Linguistics (to be published).

[Kashyap & Oomen, 1981] Kashyap, R.L., B.I. Oomen. An effective algorithm for string correction using generalized edit distances. I. Description of the algorithm and its optimality. Information Science, 1981, Vol. 23, No. 2, p. 123-142.

[Mays et al., 1992] Mays, E., F.J. Damerau, R.L. Mercer. Context-based spelling correction. Information Processing and Management. 1992, Vol. 27, No. 5, p. 517-522.

[Vossen, 2000] Vossen, P. (ed.). 2000. EuroWordNet General Document. Vers. 3 final. www.hum.uva.nl/~ewn.

[Wagner & Fisher, 1974] Wagner, R.A., M. J. Fisher. The string-to-string correction problem. J. ACM, Vol. 21, No. 1, 1974, p. 168-173.

Author information

Igor A. Bolshakov – CIC-IPN, Research Professor; Center for Computing Research (CIC), National Polytechnic Institute (IPN), Av. Juan Dios Bátiz s/n esq. Av. Miguel Othon Mendizabal, Unidad Profesional "Adolfo Lopez Mateos", Col. Zacatenco, C.P. 07738, D.F., Mexico; e-mail: igor@cic.ipn.mx

Alexander Gelbukh – CIC-IPN, Research Professor; Center for Computing Research (CIC), National Polytechnic Institute (IPN), Av. Juan Dios Bátiz s/n esq. Av. Miguel Othon Mendizabal, Unidad Profesional "Adolfo Lopez Mateos", Col. Zacatenco, C.P. 07738, D.F., Mexico; e-mail: gelbukh@cic.ipn.mx, see also www.gelbukh.com

TOWARDS COMPUTER-AIDED EDITING OF SCIENTIFIC AND TECHNICAL TEXTS

E. I. Bolshakova

***Abstract**: The paper discusses facilities of computer systems for editing scientific and technical texts, which partially automate functions of human editor and thus help the writer to improve text quality. Two experimental systems LINAR and CONUT developed in 90s to control the quality of Russian scientific and technical texts are briefly described; and general principles for designing more powerful editing systems are pointed out. Features of an editing system being now under development are outlined, primarily the underlying linguistic knowledge base and procedures controlling the text.*

***Keywords:** scientific and technical texts, automatic editing, linguistic knowledge base.*

Introduction

Scientific and technical writing is by no means easy, even for skilled and experienced authors. Usually, the elaboration of a good scientific or technical (sci-tech) text is iterative and time-consuming process, with several persons taking part in it. Besides an author of the document, colleagues, reviewers, and an editor participate in the process, helping the author to improve the text.

Scientific papers and technical documents are essential means of communication between scientists and engineers, therefore the efficacy of the communication depends on the quality of texts. A professional editor of sci-tech texts not only looks for grammar and spelling mistakes, but also accomplishes editing specific for functional style of scientific and technical prose: controlling word usage, revealing drawbacks in logic of reasoning, judging text organization, etc. [10]. The editor explains revealed defects and drawbacks, as well as proposes possible ways of how to overcome them, thereby helping the author to improve the text and to enhance its stylistic uniformity. Almost all sci-tech writers need some aid of professional editor, and without it they lack computer systems automating certain editor functions.

Of course, well-known universal computer text editors and spellers (e.g., MS Word) are widely used for preparing texts. These systems reveal many mistakes, including spelling and simple syntactic mistakes, and their facilities

are permanently extended. But the universality of these systems means that they do not account for specificity of the particular text style and genre, in particular, sci-tech prose with its intensive usage of terms and the other highly standard units. Therefore, additional computer tools are needed for checking scientific and technical texts.

As a whole, sci-tech editing involves a wide spectrum of checks concerning different text levels, so that a deep syntactic, semantic, and logic analysis of the text are required. For this reason, it cannot be fully automated in the nearest future. Nevertheless, several computer systems were built for improving the quality of sci-tech documents, e.g., [1, 5, 7], demonstrating useful features. The systems can do many checks to provide initial editing that earlier has been done by editors or by the writer personally. Most systems are special-purpose editing systems, such as CRESS [5] designed to simplify texts in a narrow problem domain, namely, navy technologies.

Two experimental editing systems named correspondingly LINAR and CONUT were developed in 90s at Moscow State University (MSU) [1, 7]. While the former system was intended to control the quality of technical documents in the narrow subdomain of computer science, the latter was built to support students' practice in writing theses, primarily, to check students' texts with respect to the formal rules of text design, e.g., regularity of abbreviations, bibliography list, references, etc. Having encouraging results in the development of these systems, we aimed at development of a system with advanced facilities, regarding it as a further step towards automation of intellectual functions of sci-tech editor.

The paper shortly describes the systems LINAR and CONUT and summarizes experience of their development, in order to propose designing principles for future editing systems. Then, the research effort going on at the MSU to design more powerful system for checking the quality of sci-tech texts is discussed, including the incorporated linguistic knowledge base and procedures controlling the text. For the sake of clarity, specific features of sci-tech prose are outlined first, along with frequent defects of sci-tech texts.

Sci-Tech Prose: Norms and Defects

Functional style of scientific and technical prose comprises texts of various genres and particular types – research papers and monographs, theses and manuals, reviews and abstracts, technical reports and instructions, patents, etc. This style is admittedly the most distinctive one; its specialty stems from the necessity to express ideas in precise and simultaneously concise manner. The specialty concerns various language levels: lexis and phraseology, syntax, discourse, and composition. Norms and rules, as well as standard language devices and units optimizing sci-tech communication were formed on each level [8, 10].

Sci-tech lexis and phraseology comprises both terms of the particular terminology (e.g., *compiler*, *square root*) and common scientific words and expressions (e.g., *to test the hypothesis, for this reason, summing up*). Whereas specific terms denote concepts, objects, and processes of the particular domain, domain-independent common expressions are used to organize sci-tech text narrative, namely, to express the logic of reasoning, to connect text fragments devoted to different topics, and to structure the text. Among collocations taken from common scientific lexicon, there are clichés, which are relatively stable standard expressions exploited as ready-for-use colloquial formulas (e.g., *to outline directions of further research, the paper reports on*).

As regards the other language levels and corresponding devices and units, we should point out discourse devices. Sci-tech text narrative is organized in accordance with a number of typical discourse-composition frames, including specific ones used in texts of particular genres.

The commonly accepted norms and rules governing sci-tech texts are more of less completely explained in the books devoted to sci-tech writing, e.g. [6, 11]. The writer should follow these rules in order to obtain an accurate, informative, easy readable and understandable text. In particular, on the lexis level, terminological consistency and stability within the given document are required, which implies usage of a standard terminology, unambiguous nominating of concepts within a paper, as well as correct introducing of new terms into the text. We should note, by the way, that newly introduced terms, we call them author's terms, are inevitable in sci-tech prose. Indeed, in scientific papers they denote new concepts and ideas, while in technical documents author's terms designate certain processes and devices. Thereby, author's terms should be properly defined or explained as they are introduced.

To follow all commonly accepted norms and rules is rather difficult task for writers, so sci-tech texts often have various faults. Especially, many students' works (theses and abstracts) are full of defects, since students usually acquire writing skill mainly through their writing practice. Our experience in reading scientific papers in the

domains of computer science and computational linguistics, as well as in reading and editing texts written by students shows that observed text faults vary on their nature. We have compiled and classified a list of frequent defects, which is presented below along with some illustrative examples (given in English and Russian languages) and short explanations of violated rules or requirements.

- Inaccuracy in usage of special terms, both author's and generally accepted ones (the latter are usually referred to as dictionary terms): usage of new terms without their definition or explanation; unjustified usage of multiple term variants (i.g., grammar synonyms *swing smoothing method* and *method for smoothing of swings*, Rus. *сцепление* и *сцепка*). It is worth noting that terminological synonymy is not advisable, only a few term variants is acceptable.
- Stylistic and grammar mistakes in combining words of common scientific lexicon and in standard clichés: wrong collocations (e.g., *to examine a problem* instead of *to study a problem*), mistakes in syntactic agreement and government, omission of clichés elements (e.g., Rus. *обращает внимание сходство* instead of *обращает на себя внимание сходство*).
- Awkward phrases and sentences with multiple complex constituents, such as subordinate clauses, homogeneous parts of the sentence, and nested constructs in parenthesis; as well as several similar syntactic dependencies with the same preposition (e.g., *process of revision of complex fragments of expository texts*).
- Syntactic ambiguity entailing semantic ambiguity: ambiguous anaphoric elements (mainly pronouns), ambiguous grammar structure of a word combination or of a sentences in the whole (e.g., Rus. *недостаток машин* – a lack of machines or a defect of machines?)
- Syntactic heterogeneity of list items (e.g. *1) to consider material 2) to process data 3) analysis of results*); semantic incompatibility of homogeneous parts of the sentence (e.g., *compilation, interpretation, and translator*)
- Drawbacks of discourse-composition structure: weak coherence of the text, lack of logic relations between text fragments (sentences and paragraphs). It is worth noting that such relations are normally expressed by connectors, such as *nevertheless, since, to this end*.
- Violations of commonly accepted rules of sci-tech texts design, such as rules of citation, referring, numeration of text units, abbreviation of words and word combinations, etc.

Clearly, many of these defects and drawbacks are specific to sci-tech prose and are not controlled by universal commercial text editors. Meanwhile, special-purpose editing systems, such as LINAR and CONUT described below check for some presented defects.

Two Systems for Sci-Tech Text Editing

LINAR [7] seems the first system developed for editing Russian sci-tech text. It was intended to control the quality of Russian sci-tech documentation, primarily technical reports and texts of technical tasks on the theme "Architecture of multiprocessor systems".

Besides spelling control, LINAR provides a sufficiently wide set of specific checks. It reveals some style defects (such as presence in the phrase of several words with the same root: e.g., *functional, function*), defects of sentence structure (e.g., violations of neutral order of words), particular semantic defects (e.g., inconsistencies like *compiler, interpreter, and processor*), defects in text composition (such as absence of obligatory parts of document or improper order of parts). These facilities are based on relatively deep morphologic and syntactic parsing of words and phrases; elements of semantic analysis are used as well. Controlling procedures exploit several computer dictionaries, among them a large dictionary of word stems and a semantic dictionary (thesaurus) representing relations between terms of the given problem domain.

As compared with universal text editors, LINAR presents, besides diagnostics indicating revealed text defects and variants to correct them (if any), short explanations of the violated rules. To initiate checking process, the user specifies what checks are to be applied and to what text fragments. This feature of LINAR is connected with its module organization: its program kernel consists of procedures, and each of them is intended to control the particular rule or property (aspect) of text.

The system CONUT [1] was implemented in the late 90s for controlling and editing students' texts (theses and abstracts). It checks texts with respect to formal rules of text design, which comprise:

- ✓ consistency of abbreviations (their introduction and usage);

- ✓ correspondence of bibliography references in the text to the bibliography list;
- ✓ presence of obligatory parts of document (e.g., introductory part and table of contents);
- ✓ correctness of table of contents (its correspondence to headings and pages in the text);
- ✓ regularity of numeration of text units and pictures.

Such rules are considered formal because they do not concern the meaning of the text and its units, but only their organization.

CONUT can also estimate some aspects of style of the text, in particular, its simplicity (readability). A heuristic formula was proposed for this purpose, which accounts for various elements indicating sentence complexity (the number of words, punctuation marks, conjunctions, and pronouns in the sentence).

It is considered important that CONUT not only reveals defects in texts and estimates text style, but also explains the essence of formal rules and style estimation methods being applied. CONUT provides a reference guide accumulating information about formal rules of sci-tech text design. The guide is usable in checking process: when any defect is identified, a corresponding page of the guide (wherein the violated rule is explained) is given to the student. The reference guide is flexibly organized: its text material is represented as a hypertext, enabling both free navigation and learning the material in the recommended (predefined) order.

Comparing this system with LINAR, it should be noted that CONUT does not analyze text properties for which natural language parser is needed, most formal rules can be applied for the check without large dictionaries and syntactical-semantic analysis of phrases. Meanwhile, CONUT provides wider range of formal checks oriented just to revising students' texts and has advanced tutoring function, thus facilitating students' practice in sci-tech writing.

Two described systems demonstrate some significant features, which ought to be considered while developing more powerful and helpful editing systems.

Designing Principles for Sci-Tech Text Editing Systems

Starting at MSU the development of new experimental editing system and having in mind the list of frequent defects described earlier, we intended to concentrate upon checks that are not fully implemented in LINAR and CONUT or are absent in them. At the same time, we take into account following crucial principles derived from our previous experience.

First, future sci-tech text editing systems will inevitably be based on semi (not fully) automatic editing procedures. One reason is obvious: the reliability of all automatic text processing programs is ranging approximately from 70 to 97%. Thus, the result of automatic checks of the text might be wrong, and proposed variants of how to revise text defects (if any) might be improper. So the author of the text should control all results and should ultimately choose ways of text revising. Hence, revising process implies a dialog between the user and the editing system, the latter indicates problems areas to be corrected and proposes variants of revision (as a human editor usually does) while the user makes appropriate decisions.

Second, it is not convenient to straightway apply all possible checks at user's text, since the spectrum of checks and estimations is sufficiently wide even in LINAR and CONUT, and their accomplishment might lead to a time-consuming process with vast diagnostics. It seems more reasonable when the user sequentially chooses desirable checks, initiates checking, and then analyzes obtained results, making necessary decisions. This feature of user interface determines the principle of system organization – the program kernel performing various text checks and estimations should be implemented as a set of procedures, each one controlling the particular rule or norm (for example, correctness of abbreviations) or estimating the particular aspect of the text. This principle was successfully tested within LINAR and CONUT; it enables to easily increment the power of the editing system.

Third, although tutoring function of editing systems is clearly an auxiliary one, it is of no little significance. Comprehensive explanations of particular text defects, and also the explanatory information about formal and informal requirements to sci-tech documents can facilitate writing. Thus, it makes sense to reinforce editing systems with special dictionaries (such as systematic dictionary of common scientific words and expressions) and a reference guide explicating various norms and rules of sci-tech writing – even if some rules can not be checked so far in the particular system. For example, the guide can present explanations and typical examples of how to

properly introduce new terms into scientific and technical texts. For students, such a guide can serve as a tool for systematic learning of sci-tech writing.

Forth and finally, in order to implement checks and estimations specific to sci-tech prose, an editing system should include vast linguistic knowledge base comprising both domain specific and domain independent components reflecting features of the prose.

According to discussed principles, main components of our novel system are the following: module implementing user interface, procedures controlling and estimating text properties, linguistic knowledge base, and reference guide that provides (besides explanation material) browsing data from the knowledge base.

Linguistic Knowledge Base

In order to ensure special-purpose control and estimation of Russian sci-tech texts, several linguistic components are being built into the system:

- Terminological dictionary [3] accumulating units, i.e., single and multi-word terms of commonly accepted terminology of computer science, gathered from several available text dictionaries. The dictionary includes known synonymous variants of terms, in particular, acronyms and other abbreviated forms (e.g. *central processing unit, CPU*). Most terms are nouns and noun combinations, but verbs are included as well. The dictionary represents relations between terminological units, primarily, class-subclass and part-whole relations, thus the dictionary can be regarded as thesaurus.
- List of definition templates [2] describing typical single-sentence definitions of new terms. Such definitions were compiled through manual scanning of sci-tech texts, they contain standard lexical units, such as nouns *term*, *name*, etc., verbs *call, refer, define*, etc. Definition template specifies both immanent lexical components and empty slots (places), their syntactic and semantic properties. An example of definition template is <*Ph*> *we will be call* <*N*>, where *N* denotes an author's term, and *Ph* is a noun phrase explaining its meaning.
- Dictionary of words and word expressions of common scientific lexicon [4]. It comprises both autosemantic and auxiliary words, noun and verb-noun combinations, adverb and participle expressions, compound prepositions and conjunctions. The dictionary unit represents adequate information: syntactic properties of word combinations (interrupted / uninterrupted, stable / free, semantic and syntactic valences, etc.), and semantic class and group of the unit within the proposed semantic classification. The classification comprises 5 main classes: text structuring and composing (e.g., *in addition, next*); expressing logical relations (e.g., *provided that, hence*); indicating sources of information (e.g., *in their opinion*), author's estimates (e.g., *essentially, it is quite likely*); structuring scientific knowledge via common scientific variables (generic nouns), such as *analysis*, *result.* For the latter class, dictionary units describe also syntactic combinability of the noun (e.g., *significant result, to derive result, to question result*).
- Dictionary of standard clichés (stable colloquial expressions) comprising both phrasal formulas (e.g., *the paper describes main features, argument can be made against*) and predicative constructs (e.g., *to outline directions of further research, to take as starting point for*). Some clichés are common for sci-tech prose, the others are specific for particular genres. Clichés are described by templates similar to definition templates: empty slots are indicated, and their syntactic and semantic properties are specified.
- Morphological dictionary of word stems, which covers all words encountered in the other dictionaries (separately or within any multi-word combination). The dictionary unit represents adequate morpho-syntactic information, e.g., part of speech and flexional class (if any), as well as pointers to units of the other dictionaries describing available combinations with the given word (stem). Thus, morphological dictionary connects units of different dictionaries, facilitating their recognition in texts.
- Inventory of prototype discourse frames specifying discourse-composition structures of sci-tech texts. Some frames are domain-specific, e.g., a frame specifying composition of texts that describe particular technical devices. Each slot of prototype frame corresponds to typical subtopic (e.g., functionality of device) and contains pointers to dictionary clichés and common scientific expressions that signal the subtopic in the text.

Text Analysis on Different Levels

We outline a few methods and procedures proposed to implement specific checks of sci-tech texts, which concern different text levels and exploit surface syntactical analysis.

Analyzing Terms and Common Scientific Expressions

For checking regularity of term usage and usage of common scientific expressions, terms and expressions should be recognized in texts. Identification of author's terms presents the major difficulty [2]. Indeed, they are free and often unstable multi-word nominal combination (e.g., *coefficient adjustment learning*) matching several possible syntactic patterns. Moreover, author's terms might be used in texts without any definition or explication, and in order to properly recognize them, local syntactic analysis should be complemented with elements of lexical semantics and term occurrence statistics.

An automatic recognition procedure is proposed [2] based on surface syntactic analysis and dictionary information. The procedure makes use of particular syntactic patterns of terms (e.g., coordinated combination of adjective and noun) and takes into account possible syntactical-semantic variants of new terms (such as *candidate elimination algorithm* and *algorithm for elimination of candidates*). The procedure exploits morphological analyzer converting words to their normalized forms and computes frequencies of term occurrences.

According to this procedure, occurrences of dictionary terms and collocations of common sci-tech lexicon are extracted first. Then, certain author's terms are identified by looking for sentences that match dictionary definition templates and by extracting lexical units from the proper places of the encountered sentences. And finally, the procedure attempts to recognize undefined author's terms: it detects word combinations of the given syntactic patterns, identifies among detected combinations different variants of the same term, and gathers them into groups of related term variants along with computed frequencies.

Units recognized at the last step of the procedure are regarded as term candidates, i.e., as potential new terms. Compiled list of term candidates is to be presented to the author of the text for validating and further revising, for example, selecting the most appropriate term in each group of related variants.

Identifying Syntax Ambiguity and Heterogeneity

We consider simple kinds of syntactic ambiguity, mainly ambiguous pronouns and syntactically ambiguous noun combinations. To check pronouns, local analysis of previous context is applied. For noun combinations we propose rather simple procedure of local syntactical analysis that checks dependencies of words within the noun combination.

To check syntactic homogeneity of items in lists, as well as homogeneous parts of the sentence, an automatic procedure is used, which differentiates noun and verb phrases from phrasal constructs and additionally distinguishes several types of noun and verb phrases.

Analyzing Discourse Structures

For recognition discourse structures of texts we propose a heuristic multi-step procedure, which exploits data from the dictionary of standard clichés and the dictionary of common scientific lexicon.

The recognition procedure first searches in the text for all sentences and composing clauses that match dictionary clichés. Occurrences of common scientific expressions signaling discourse relations are looking for as well. For this purpose, local context techniques similar to those described in [9] are used. When an instance of dictionary cliché or common scientific expression is recognized in the text, the procedure extracts lexical units from proper places of the instance and makes an attempt to fill slots of discourse frames related with this cliché or this expression. In general case, after recognition in the text of all cliché instances and common scientific expressions, slots of several discourse frames might be filled, therefore the frame with the maximum number of filled slots is selected as appropriate one. Information associated with this frame is used to estimate drawbacks of the recognized text structure, in particular, improper order of text parts devoted to particular subtopics.

Estimating Composition

Among parameters indicating quality of text composition, we consider proportionality of units of the same level, for example, proportionality of chapters or proportionality of items in itemized list. The proportionality means that units have comparable sizes, and the size can be computed as the number of words in the unit (another option is the number of units of the level underneath). It seems reasonable to estimate the proportionality of sentences within each paragraph, items within each itemized list, paragraphs within each chapter or text section, and chapters within the whole text.

To evaluate the proportionality, dispersion *D* and mean square deviation σ are calculated:

$$\sigma = \sqrt{D}, \qquad D = \frac{n\sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i\right)^2}{n^2}$$

where x_i is the number of words in *i*-th component of the text, and *n* is the number of components.

Function $exp(-0.1*\sigma)$ is proposed to obtain a measure within the interval *(0,1)* and with maximum values corresponding to texts with better composition.

So the formula for a proportionality value is $E_C = exp(-0.1*\sigma)$. The formula is used systematically to estimate the proportionality of units on different levels: sentences, itemized lists, paragraphs, sections, chapters, and also headings of sections, with x_i denoting the number of words in units being estimated, and *n* denoting the number of them. The estimation value for the whole text can be computed as the arithmetic mean of proportional values already computed for all text units.

Conclusion

While discussing the problem of computer-aided editing of scientific and technical texts we outlined peculiarities of special-purpose systems built for editing Russian text and pointed out the principles applicable for designing editing system with advanced facilities. We also outlined some features of a novel editing system being under development, primarily its linguistic knowledge base and procedures controlling the text. We hope that the system will be useful for a wide community of sci-tech writers, in particular for inexperienced authors, such as postgraduate students.

By now, the linguistic knowledge base of the system is partially implemented, with the terminology dictionary covering terms in certain narrow subfield of computer science and the dictionaries of common scientific lexicon and standard clichés containing more than 700 units. The first versions of the controlling procedures are tested.

Bibliography

1. Bolshakova, E. *Computer Assistance in Writing Technical and Scientific Texts*. Proceedings of 2nd International Symposium "Las Humanidades en la Educación Técnica ante el Siglo XXI", México, 27-29 September, 2000, p. 59-63.
2. Bolshakova, E. *Recognition of Author's Scientific and Technical Terms*. In: Computational Linguistics and Intelligent Text Processing. Second International Conference CICLing 2001. A. Gelbukh (Ed.). Lecture Notes in Computer Science, N 2004, Springer-Verlag, 2001, p. 281-290.
3. Bolshakova, E., Vasilieva N., Yudin D. *Extraction of Dictionary Terminological Word Combinations in Scientific and Technical Texts*. Proceedings of International Workshop on Computational Linguistics and its Applications Dialogue'2001. Russia, 2001, V. 2, p. 48-51 (in Russian).
4. Bolshakova, E. *Desingning Principles for a Computer Dictionary of Commom Scientific Lexicon.* Proceedings of International Workshop on Computational Linguistics and Intellectual Technologies Dialogue'2002. Russia, 2002, V. 1, p.19-23 (in Russian).
5. Glenda, M. *Readability Formulas: Useful or Useless?* IEEE Transactions on Professional Communications. Vol. PC-30, No 1, March, 1987, p.12-15.
6. Emerson, F.B. Technical Writing. Houghton Mufflin, 1987.
7. Malkovsky, M.G., Bolshakova E.I. Intellectual System for Control of Text Quality. In: Intellectual Systems, V. 2, No. 1-4, Moscow, 1997, p. 149 –155 (in Russian).
8. Mitrofanova, O. *Language of Scientific and Technical Literature*. Moscow University Press, 1973 (in Russian).
9. Paice, C., Jones P. *The Identification of Important Concepts in Highly Structured Technical Papers*. Proc. of 16th Annual Int. ACM SIGIR Conference on Research and Development in Information Retrieval, Pittsburg, 1993, p.69-78.
10. Senkevich, M. *Style of Scientific Speech and Literary Editing of Scientific Works*. Moscow, Vysshaia Shkola, 1976 (in Russian).
11. Zobel, J. Writing for Computer Science. Springer, 1997.

Author information

Elena I. Bolshakova – Moscow State Lomonossov University, Faculty of Computational Mathematics and Cybernetic, Algorithmic Language Department, Docent; Leninskie Gory, Moscow State University, VMK, Moscow 119899, Russia; e-mail: bolsh@cs.msu.su

THE TECHNOLOGY OF NEW DOMAINS' ONTOLOGIES DEVELOPMENT

B. Dobrov, N. Loukachevitch, O. Nevzorova

***Abstract**: In the paper we consider the technology of new domain's ontologies development. We discuss main principles of ontology development, automatic methods of terms extraction from the domain text and main ontology relations.*

***Keywords**: Ontology, thesaurus, automatic term extraction from the text, ontology relations.*

Introduction

The present article is dedicated to the technology of creation of the so-called linguistic ontologies, i.e. ontologies, concepts in which are generally based on the semantics of the domain terms. Such ontologies are usually used for the automatic processing of texts in a natural language.

The technology has been developed in the process of creation by the article authors of large and extra large ontologies and thesaurus for various domains and their actual usage in multiple applications of automatic processing of texts.

Among such works are the following:

- thesaurus of the social and political life (28 thousand concepts, 67 thousand terms, 100 thousand concepts relations), then - Social and political thesaurus [Loukachevitch, 2002], which is a search means in the University information system of Russia (www.cir.ru) and is used in such applications of automatic processing of texts as conceptual indexing, automatic rubrication, automatic annotation;
- the Russian language thesaurus RuThes (43 thousand concepts, 100 thousand words and expressions, 166 thousand consepts relations) [Loukachevitch N., 2002];
- thesaurus for the domain «Elections» - is included into the Social and political thesaurus;
- thesaurus for the domain «Economics statistics»;
- ontology for the domain «Software functionality» for the expert decisions making support during complicated program complexes testing;
- Avia-Ontology for the domain, describing the operator (air crew) and airplane board equipment in different flying operations (1200 concepts, 3400 terms) – for the analysis of the completeness of documents, describing the logics of working in different typical situations [Nevzorova, 2001].

Avia-Ontology is currently being developed and will be used as the basic source of examples for the article [Dobrov, 2002].

It should be highlighted that the peculiarity of the proposed technology and the existing experience is namely the activity of a knowledge engineer, who at the beginning of works has a very superficial idea of a conceptual structure of the domain and its terminological body.

1. The formation of textual collection

One of the essential conditions of the successful development of a linguistic ontology is a preliminary creation of an electronic textual collection with reference to a domain. The collection may be of different genres and may include textbooks, scientific articles, technical data, mass media works and so on.

During the development of the thesaurus of the social and political sphere we used the textual collection of the University Information system of Russia, comprised of over 700 thousand documents: official documents, laws, scientific works on social studies, newspaper releases.

During the development of the Avia-Ontology a great effort had to be made in order to form a sufficient collection of electronic documents on the given domain. To a great degree scanning of the printed matter and search of relevant materials in Internet had to be done. As a result an electronic collection with the size over 100Mb was formed.

2. Automatic terms extraction from the domain texts

2.1. Terminological word-combinations assembling on the basis of syntactic information

Having formed an electronic collection of domain texts it is necessary to obtain its "terminological portrait" in the first place, for which the procedures of automatic extraction of terms, essential for the given domain, are employed. For the Russian language terminology the syntactic structure of over 90 percent of different domain terms [Loukachevitch, 2002] refer to one of the constructions from the following list:

- single nouns, adjectives and words unknown to the morphological dictionary – usually abbreviations;
- noun groups (NG) noun + noun in the genitive case;
- NG adjective + noun
- NG adjective + adjective + noun
- NG noun + adjective + noun in the genitive case

Such types of constructions are collected on the basis of a preliminary morphological processing of texts. In the process of such terminological word-combinations separation noun and adjective agreement must be tested, after which the bulk of such word-combinations are syntactically correct groups. Decreasing frequency-ordered lists of such words and word-combinations present an important data for the formation of an idea about the domain.

It should be mentioned that the list obtained is greatly spoilt by non-terminological linguistic expressions.

First of all, any domain texts contain a great number of words of general meaning, e.g. *possiiblity, means, condition, type, point* and others.

The larger is the domain and the textual collection, the bigger problem present multiword constructions of general meaning, such as *task solution, further development, uniform system, beginning of year, present time*. During the Social and political thesaurus development we made use of special lexical filters, set as a specialized vocabulary to exclude from consideration such word-combinations [Loukachevitch, 2002]. Such filters are undoubtedly useful for the creation of extra large ontologies, however, they also aim at the domain and therefore are namely suitable for big developments. So, for instance, the word *argument* may be considered a non-terminological one in the social and political domain, however, *argument* in the domain of software is an important term.

Besides, in a domain there may be longer terms or terms of a different syntactic construction, like those containing prepositions.

It should however be highlighted that the simplicity of the given algorithm of separating words and word-combinations of a domain is an important factor of its use for the analysis of the domain structure.

2.2. Terminological word-combinations separation on the basis of texts structure

In order to separate from texts longer terms and/or terms containing prepositions another method is used.

Many algorithms of multiword terms automatic separation use the supposition that words, comprising a term, frequently occur together. [4, 5]. To find terms of a more complex syntactic structure that described earlier, we use our own variant of algorithms of such type.

If the text author makes use of a certain term as a separate unit of narration, depends on this term in his narration, then exactly in this text term words will occur alongside more frequently than spaced out.

To reveal this during the processing of the text for every word (noun, adjective) the immediate neighbor word and neighbor-words in the text window of a given size are stored. A table of immediate neighbor words and a neighbor table in the text window are created and the frequency of word pairs occurrence is calculated.

Further it is expected that if a pair of words occur as immediate neighbors in over half of cases of their occurrence in the same text window, this proves that this pair in aggregate serves to the author as a reference point, i.e. represents a term or a fragment of a term.

In this case the word pair is glued together to form a common terminological unit and the tables are recalculated as if this unit has been known from the very beginning before the text processing , which gives the possibility to the further development of the term, thus forming units of length 3 and longer. Examples of terms obtained in this way in the aviation domain are the following: *state duty at the aerodrome, enemy plane destruction, duty in the air, throwing into battle, coming into military contact* and so on.

In the domain "Elections" such terms as *member of election committee with right to deliberative vote, executive body of local government, local governor's elections* were obtained.

When being reviewed, the terms obtained are ordered not by the frequency of their occurrence, but by the number of texts in which they occurred. It is supposed that a special attention should be paid to those word-combinations which remained stable in over two texts. As a result of the experiment on the text collection of 50Mb newspaper releases 1346 such word-combinations were obtained. 80% of them were qualified as terms. The comparison was drawn with the big terminological resource of the Social and political thesaurus and there were singled out approximately 30 important terms, not present in it at that moment, such as *volume of output, transitory economics, capital flow.*

The algorithm drawback is that it practically does not extract terms from short texts - the term has to be used in the text at least twice, better three times. However this drawback is insignificant when dealing with extra large collections, such as the collection of UIS of Russia. Thus we suggest to process the whole collection using the abovementioned algorithm to single out terms not present in the Social and Political thesaurus and to further monitor the appearance of new terms.

We consider useful to apply both described algorithms to work with small domains.

3. Using generally valid linguistic resource as a base for the development of applied ontology

Terminology of any domain contains both specific terms, used only in the given domain or in the range of similar domains, and rather commonly known terms. In the given domain the examples of such commonly known terms are *pilot, airplane, pursuit plane, weapon, operations, attack* and many others. This allows not to begin the domain model development from point zero, but to use the knowledge described in more general linguistic and terminological resources.

As such a source we used the Thesurus of the Russian language RuThes [Dobrov, 2002]. The Thesaurus represents a hierarchical chain of concepts, each of which has a number of textual variants (linguistic expression means) and the total of thesaurus other concepts relations. The resource volume nowadays makes up 97 thousand words and word-combinations, confined to 42 thousand concepts. Over 160 thousand relations were manually determined among the concepts. On the transitivity and sequence characteristics over 1200000 relations among concepts were established.

The RuThes Thesuarus contains two big parts. RuThes is comprised of the Social and Political thesaurus, including terminology of economic, political, military, social, scientific and other spheres (64 thousand words and terms). The zone of words and word-combinations of the thesaurus, designating actions, situations, objects, which can occur in any subject area texts, and thus not included in the Social and Political thesaurus, is called general lexicon (33 thousand words and word-combinations).

The presence of a big generally valid linguistic resource makes possible to compare the given resource with the domain texts, to single out the knowledge types described in the thesaurus (concepts, synonyms, relations between concepts), to transfer them to a special working domain as a basis for the creation of a object-oriented ontology. At the same time we use not only those concepts, mentioned in the SO texts directly, but also those thesaurus concepts, found on the conceptual ways among concepts mentioned in the texts, i.e. a sort of a conceptual closing is used.

Lists of gathered words and word-combinations in the domain were compared to the terms of the Social and Political thesaurus. If the comparison of the next word-combination was successful, a corresponding concept from the Social and Political thesaurus together with all the terms that express it in the text, was copied into the domain model. During next step all the relations of the Social and Political thesaurus among the copied concepts were copied.

Besides, the closing of relations was performed: if a concept B is superior compared to concept A, concept C is superior to concept B, while concepts A and C were copied into a domain, then concept B is also copied into the domain model together with its terms variants and relevant relations.

The comparison with the general lexicon zone was controlled manually, as part of such words preserved its general meaning, for instance *necessity, conditions*, etc. and , thus, do not need to be reflected in the domain model, other words like *chandelle, cover, escort* are important concepts of the domain, which therefore have to be

reflected in the model. In the second case the corresponding concepts were also copied together with their terms and relations, just as during the comparison with the Social and Political thesaurus terms.

Undoubtedly, the transferred concepts and relations require a thorough additional test for adjusting the setup to a certain domain. For instance, as a result of transfer among the terms we can come across a synonymic variant, which is improbable in a given domain, for example, the word-combination «rotorcraft» as a synonym of the word «helicopter» is unlikely to be found in the professional technical area.

In the present version of the Avia-Ontology about one-third (1100) terms were transferred from the thesaurus RuThes together with synonymic relations and relations among concepts, which provided a fast development of the Avia-Ontology and a considerable reduction of the new resource development time.

4. The main procedure of the ontology completion

Having exhausted the available linguistic resources, we begin analyzing of the available texts and word and word-combinations lists for the further ontology development.

The series of decision making in this process consists of the following steps:

- on the basis of the available textual material we look for a word or a word-combination, designating an important domain – how to determine the concept «importance», we will consider in the following article paragraphs. Upon entering a new concept we must provide it with an understandable and unambiguous name. If possible it is better to enter the concept names which are unambiguous even outside the current domain. For instance, when entering a concept "airplane wing" it is possible to name it *a wing* – in the given domain this word is unambiguous, but it is better to give it a clearer name, which will not lose its unambiguity with the domain expansion or when including this ontology into a larger ontology - when entering a concept at least one relation of this concept with the other ontology concepts is entered. On the one hand, it is supposed that if it is rather complicated to set a relation for the entered concept, then it is too early to enter such a concept and an additional analysis is required. On the other hand there is no need (and as a rule it is impossible) to describe all the necessary relations of the new concept at once. Practice proves that by entering further new concepts the initial position may become clearer– in order to enter something new, it is often necessary to correct the inaccuracies and distortions of the old. Properly speaking, it is this new that reveals the problems of the existing description;

- and, at last, a concept must be supplied with a list of words and word-combinations, which may help to refer to an entered concept in the text. As such textual entries may perform separate words (nouns, adjectives, verbs), and also noun and verb groups. At the same time a multiword linguistic expression must be used in the texts as an inseparable construction. A textual entry may be polysemantic (may have another meaning), then it must be marked as a polysemantic. Besides, a sequence of normalized forms of all constituents of a multiword expression must be entered (masculine gender, nominative case, singular), which will be used for terms recognition in the texts.

5. Ontology concepts selection

5.1 Separate words-based concepts

In the total of separate words, occuring in the texts of this or that specific domain, it is easy to distinguish two subgroups, which help to make a decision on whether to include or not into the ontology a concept corresponding to their meanings.

One of such groups - evident terms of a given domain, for example, *airplane, flaps,* and corresponding concepts must enter the ontology. Another group – evident words of general meaning, such as *necessity, possibility, creation, etc.,* for which inclusion into the ontology is not necessary.

It is more problematic to make a decision on the other groups.

One of such groups - terms of a given domain that appear on the basis of a general lexicon words like *turn, turn, dumping/stall*. Narrowing or other changes of basic meanings are characteristic of such words. So, in the aviation domain *turns* refer to the airplanes. Besides, to help in distinguishing such terms serves the fact that in the terminological word-combinations lists there is a considerable amount of different word-combinations with the

inclusion of this word, which should also enter the ontology: *chandelle, forced chandelle, chandelle velocity, chandelle radius.*

Another «complex group» of words are words that clearly refer to the vocabulary of a general meaning, but ontology includes a certain number of concepts, based on the terms with this word: *exit – disengagement, withdrawal from an attack, withdrawal from a maneuver*. The questions arises if corresponding generalizing concepts should enter the ontology. Two aspects should be considered. On the one hand, the appearance of such a generalizing concept provides an additional structure to the ontology, which is a positive factor. However, on the other hand, if a word is polysemantic in the framework of the given domain texts, and abstract, then it may cause serious problems in the lexical ambiguity resolution, and this corresponding concept should not enter the ontology.

The last group of words, which requires a special effort to decide if the corresponding concept should be included into the ontology or not, are words, that are on the borderline between domains or those that were relatively accidentally entered into the collection texts, for instance, *aircraft construction* (whether refers to the subject area or not), *adapter, heat insulation* and so on.

5.2. Multiword word-combinations-based concepts

Any domain texts contain a great number of various word-combinations. Word-combinations selection for the inclusion of the corresponding concepts into the ontology presents a serious problem. Present terminological lists, referring to the current domain, usually embrace only a small part of those term-like word-combinations that are met in texts. Experts may not also have a certain opinion on the bulk of such word-combinations. Therefore it is necessary to have a total of principles to help to decide which specific factors are to be taken into account for including multiword word-combinations-based concepts into the ontology.

We should highlight the main principle at this point. If such a concept is included into the ontology, it should happen not so much because the corresponding word-combinations refers or not to the ambiguous category of the given domain terms, as which new information the appearance of this concept in the ontology gives. Thus, a new concept in the ontology is the application point of an additional information which is used by the automatic system in the process of its work.

Such information may be divided into several types.

5.2.1. Existing and important

Any domain has a small number of main points which are extremely important in the given domain. Terms and other linguistic expressions that correspond them are highly frequent in the subject area texts. Such main points (concepts, single objects) must be reflected in the ontology. So, working in the domain «Elections» it is essential to have in the ontology a conceptual unit *CENTRAL ELECTION COMMITTEE OF THE RUSSIAN FEDERATION.*

If concepts entered into the ontology have a fixed and small number of aspectual concepts, then they have to be reflected in the ontology. So, for the domain «Elections» types of elections are reflected, in the social and political thesaurus – types of budget, in the subject area of military aviation - bombing flight regimes *(dive bombing*).

Another important type of information is that two concepts have a common subtype. For instance, concepts *DEFENSIVE MANEUVER* and *AIRPLANE FLAP* have a subtype *DEFENSIVE FLAP*, concepts *PENALTY* and *OFFICIAL REPRIMAND*– subtype *ADMINISTRATIVE PENALTY*.

5.2.2. Word-combination has «interesting» synonyms

A concept can unite the total of various textual expressions with one and the same meaning (to the parts of speech – derivatives (*to take off, take off*) are included into the total of textual entries of one and the same concept). So, the revealing of synonymic expressions or derivatives often leads to the introduction of a new concept for the fixation of the synonymy found. The variety of textual expressions in this case often points at the importance of a corresponding concept.

After the concept is set up, special effort is applied to find other ways of referring to the same concept, i.e. the synonymic range of textual entries is maximally filled. These variants may seem obvious for a person, and their entry may seem tiresome, but as practice proves, during automatic processing of various texts a direct comparison is better that any conclusion. It is often supposed that this or that variant exists, and then its actual

occurrence is checked by Internet. For instance, a new concept is entered on the basis of the term *horizontal flight bombing* and instantly existence of the synonym *horizontal bombing,* which actually exists, is checked. If afterwards a concept for the term *pitch-up bombing* is entered, then, naturally, the existence of the term *pitching bombing* is checked; such term was not found in Internet. Let us give example of synonymic row:

INCREASE OF ENGINE REVOLUTIONS
To step on the gas
To add regime to the engines
To add traction
To increase engines revolutions
To increase traction

5.2.3. Relations that do not follow from the word-combination structure

The principle used to evaluate the necessity of entering a concept into many thesaurus and ontologies is that a multiword term has relations that do not follow from its structure.

Examples of such relations: *accelerated turn– stall, attack evasion – defensive maneuver, superiority in energy–tactically advantageous position.*

To fix this relation it is necessary to introduce the corresponding concepts.

5.2.4. Completion of ontology levels

An important principle of ontology completion is the "closing" principle, which has two subtypes.

In the first place, if a new concept, introduced by any reason has created a new inferior ontology level, then it has to be completed by other essential concepts of the same level. For instance, if a *MISSILE LAUNCH* concept is entered as an inferior one for the concept *USE OF WEAPON,* then it is necessary to enter, for example, the concept *CANNON FIRING*, as the second most important type of using weapons in the given area.

This principle is at the same time limiting: if we decide to enter a new-level concept, we must evaluate the consequences of such a step: how many concepts of the same level we are going to enter; if the number of potential concepts of this level is too big, then the entry limiting principles should be determined at once. So, for instance, in the *Social and political thesaurus* there can be a lot of concepts inferior to the concept *GOODS FOR CHILDREN*. The appearance in the inferior row of concepts *CLOTHES FOR CHILDREN, SHOES FOR CHILDREN, TOYS FOR CHILDREN* is additionally justified by the existence and inclusion into the thesaurus of certain types of these goods, having separate lexemes as textual entries.

On the other hand, an opposite situation may emerge: several concepts sharing common features are found, it is necessary to find a common concept. For instance, on the basis of subject area texts analysis concepts *FLAPS* and *SLATS* are introduced, common features of which is that both are located on the wings and that they serve to control a flight. Extra attention is paid to searching for a generalization, which is found - *WING MECHANIZATION MEANS.*

Another example of generalizing two concepts that were entered: *AFTERBURNING ACTUATION* and *DECREASE OF ENGINE TRACTION* further the entry of the concept *ENGINE REGIME ALTERATION*.

5.2.5 Separate words are polysemantic and word-combination is monosemantic

An important factor which helps to determine the entry of a new concept is the presence of polysemantic words inside a monosemantic multiword word-combination.

So, a polysemantic term *press* is important for the social and political area, and to support the ambiguity resolution process we introduced into the social and political thesaurus such concepts as *OFFICIAL STAMP, CENTRAL PRESS, INTAGLIO.*

When working in the same extensive social and political area we may hesitate if the introduction of concepts *SHORT FILM– FEATURE FILM* is necessary, but as soon as it becomes known in a bilingual environment that short film in English - *short subject*, question of the corresponding concepts entry is approved at once.

We should emphasize that all the abovementioned does not mean that for every word-combination, consisting of polysemantic words, a corresponding concept is created; the described principle only helps to make the decision in such cases when we are almost ready to introduce a concept.

6. Ontology relations

The most important part of the prevailing number of ontologies is the total of relations among the concepts. This set of relations largely depends on the domain and on the task for solving which ontology is meant. We suggest to begin the construction of the ontology on a minimal set of relations and to determine the domain structure according to this set. Such a minimal set of relations does not depend on the type of a domain, on the type of the problem solved, as it is based on the fundamental properties of concepts – in the first place determining for a given concept such concepts on which depend its existence or existence of the given concept examples, i.e. determining the so-called relations of ontological dependence, which are studied in detail in the framework of the philosophical discipline «formal ontology»

The main instruments of essences analysis within Formal Ontology [Smith, 1998] are the following:

- the theory of identity, integrity. The main problems of this type of analysis: what does the fact that two essences are one and the same thing mean, how can an essence change and preserve its identity, what properties are essential for preserving one's identity, etc.
- the theory of part and the whole (mereology, mereotopology). The main problems here are the following: what is что considered as a whole, what makes an essence a whole, what is the connection of parts in the whole, what properties such a connection relation has, how is the whole separated from the «background», what are boundaries and so on.
- the dependence theory [Guarino, 1998].

The main questions of the dependence theory will be examined in detail.

The main question of the dependence theory is if the essence can exist by itself or it supposes the existence of something else:

- whether the existence of essence supposes the existence of something else (rigid dependence), for instance, *boiling* is impossible without the existence of a certain volume of liquid which boils;
- whether existence of examples of a certain class (generic dependence) is supposed, like, the appearance of the concept *garage* is impossible without the existing concept *motor vehicle,* though a certain garage may appear without any reference to a certain motor vehicle;
- whether existence of X at a certain moment of time T supposes the existence of Y at any other moment of time т1 (historical dependence), for instance, straw historically depends on threshing, as straw can not appear without a preliminary threshing process, altogether these works come to an end, while straw continues its existence for a log time.

Thus, we suggest for each created ontology to develop a sort of an "initial" ontology, in which non-toxonomic relations are relations of a conceptual dependence, and then, having determined on the basis of such an ontology a domain structure and a set of relations, necessary for solving the main problem, to specify the relations on the basis of the same set of concepts. In this case conceptual dependence relations are so important for any domain, that there is no need to delete them, it is only necessary to re-name them in the newly introduced relations system.

A specific set of relations , which is used by us now besides taxonomic relations (HIGHER-LOWER relations) is the following:

- PART– WHOLE – is used to describe the traditional parts, participants of situations, properties. Here a conceptual dependence of concept-part on the concept-whole is required;
- unsymmetric association aSC1-aSc2 – is used for the rest of conceptual dependence relations;
- symmetric association is used for concepts, similar by meaning

Thus, two relations in the relations set employed by us are significantly bound with the concept of ontological dependence. In a magnitude relation these two relations occupy approximately half of all relations in our thesaurus and ontologies.

Conclusion

The described technology of constructing ontologies for different domain was employed to create the so-called linguistic ontologies, which are used to solve different problems of the automatic texts processing. However, application of such technologies, connected with the processing of large textual collections is also useful for the creation of ontologies in those problematic domains, which are not directly connected with texts processing

The carried out analysis of the electronic textual collection ensures:

- completeness of concepts covering in reference to the collected corpus;
- objectivity of concepts and terms interpretation, as different texts from the collection are analysed.

"Minimal" relations set

- makes possible to begin the ontology construction at once, as soon as the task is set and the domain is determined;
- provides a conceptual basis for communicating with experts in the given domain;
- provides the initial domain structuring which may be used as a basis for singling out special relations in the domain.

Acknowledgments

The work has been performed thanks to the support of the Russian Fund of Basic Research, grant № 02-07-90279.

Bibliography

[Loukachevitch, 2002] Loukachevitch Natalia V., Dobrov Boris V. Evaluation of Thesaurus on Sociopolitical Life as Information Retrieval Tool // Proceedings of Third International Conference on Language Resources and Evaluation (LREC2002) / M.Gonzalez Rodriguez, C. Paz Suarez Araujo (Eds.) – Vol.1 – 2002, Gran Canaria, Spain – p.115-121.

[Loukachevitch N., 2002] Loukachevitch N.V., Dobrov B.V. Thesaurus of the Russian language for automatic processing of large textual collections// Computer linguistics and intellectual technologies: Works of the International seminar Dialogue'2002 / edited by. a.S.Narinyani – м.: Nauka – 2002. – Vol.2 - p.338-346. In Russian.

[Nevzorova, 2001] Nevzorova O.A., Fedunov B E. System of analysis of technical texts "Lota": main conceptions and project decisions. // RAS publishing house. Theory and management systems – 2001. – № 3.– pp. 138-149. In Russian.

[Dobrov, 2002] Dobrov B.V., Loukachevitch N.V., Nevzorova O.A. Computer-aided construction of the applied ontology: technological aspects // International conference IEEE Artificial intelligence systems (IEEE AIS'02) Gelendzhik-Divnomorskoe, 5-10 September 2002 года – Text processing and cognitive technologies: Collection (ed. 7) / edited by V.D.Solovyev – Kazan: Otechestvo 2002. – pp.103-109. In Russian.

[Loukachevitch, 1996] Loukachevitch N.V. Computer-aided formation of information searching thesaurus of social and political life in Russia// NTI. ser.2. - 1995. - N 3. - pp.21-24. In Russian.

[Smith, 1998]Smith B. Basic tools of formal ontology. In "Formal Ontology in Information Systems", N. Guarino, ed.

[Guarino, 1998] Guarino N. Some Ontological Principles for Designing Upper Level Lexical Resources. In "Proceedings of First International Conference on Language Resources and Evaluation".

Author information

Boris Dobrov – Research Computing Centre MSU, Russia, Moscow, Vorobjovy Gory; e-mail: dobroff@mail.cir.ru
Natalia Loukachevitch – Research Computing Centre MSU, Russia, Moscow, Vorobjovy Gory; e-mail: louk@mail.cir.ru
Olga Nevzorova – Chebotarev Institute of Mathematics and Mechanics, Russia, Kazan, ul. Kremlevskaja, 18; e-mail: Olga.Nevzorova@ksu.ru.

ONBOARD OPERATIVE ADVISING EXPERT SYSTEMS AND INFERENCE TECHNIQUE IN THEIR KNOWLEDGE BASES

B. E. Fedunov

Abstract: *On the basis of an analysis of the object domain (typical situations, semantic networks of problem subsituations for each typical situation), the inference technique used in onboard operative advising expert systems is investigated. The inference techniques based on the system of rules "if..., then ...,else ...,"on Saaty's algorithms, of multicriteria choosing of alternatives, and on algorithms using the knowledge matrix are presented.*

Keywords: *expert system, inference.*

Introduction

Onboard operative advising expert systems of typical situations (OOAES TS) of anthropocentric object functioning are intended to solve problems of the second global level of control [1, 2]. These are the socalled tactical problems; i.e., the problems determining rational ways for the attainment of a current aim of the operation, which is operatively appointed by the crew of the anthropocentric object. For any typical situation (TS) functioning, a special OOAES should be created. The structure of the knowledge base of the OOAES is based on a formal model of the object domain [2] in which the general problem of the operation of the anthropocentric object is represented via the semantic network of problem subsituations (PrS/S).

We briefly dwell on the description of the destination and the form of the inference technique in the knowledge base of an OOAES (Fig. 1; in the figure, we use the terminology corresponding to an anthropocentric of aircraft type).

Using the current information from onboard measuring devices, standard algorithms in onboard computer, signals from the information-control field (ICF) of the crew compartment, a situation vector SV(TS-PrS/S) is formed in the knowledge base of the OOAES. This vector describes the- state of the outboard and onboard environment for assigning (or identifying) current PrS/S. We call the technique of such an assignment the inference technique on the set of PrS/S. It is constructed on the basis of results of cooperating with experts who are specialists in the object domain considered. These mechanisms are implemented in the OOAES in the form of the rules "if ..., then ..., else...". Their completeness and consistency is achieved by finalizing the OOAES on systems of imitational modeling (SIM) [1] together with experts.The inference technique used for (determining) a rational solution to the current PrS/S is represented by three types of mechanisms

It seems that these three types of inference techniques do not exhaust all of the possible types inherent in the object domain considered. Of course, their choice for the top-priority investigation is stipulated by the available practice of designing the knowledge bases of the first versions of known OOAES. Below, we dwell only on the inference technique of OOAES TS proposed for use while resolving a problem subsituation.

1. Inference Technique Based on Optimization Models

The inference technique of the first type, i.e., the mechanism based on product rules, is most often used in OOAES. In a system of rules (constructed in a certain way), a situation vector SV(PrS/S-solution) that describes the current state of the problem in qualitative coordinates (the left-hand side of the product rule) is associated with the most rational (optimal) way for its resolution (the right-hand side of the product rule).

Let us briefly dwell on the methods for constructing the product rules, which are most frequently used in practice when designing OOAES knowledge bases: interviewing experts; constructing the rules on the basis of results of the investigation of optimization models.

Constructing the rules on the basis of a preliminary investigation of the problem on its mathematical model is similar to the procedure for constructing the rules on the basis of discussions with experts. In the latter, the model of the problem and the results of its investigation are in the mind of an expert, and the designer of the OOAES

should interview some experts and then formalize the knowledge obtained in the form of a totality of rules. The effectiveness and the difficulties of this method for constructing a system of OOAES rules (an inference procedure) have been discussed in [3].

Without neglecting the use of this method, we, nevertheless, prefer the second one, namely, the method for constructing the rules on the basis of optimization models. Moreover, we admit the possibility of the joint use of both these methods.

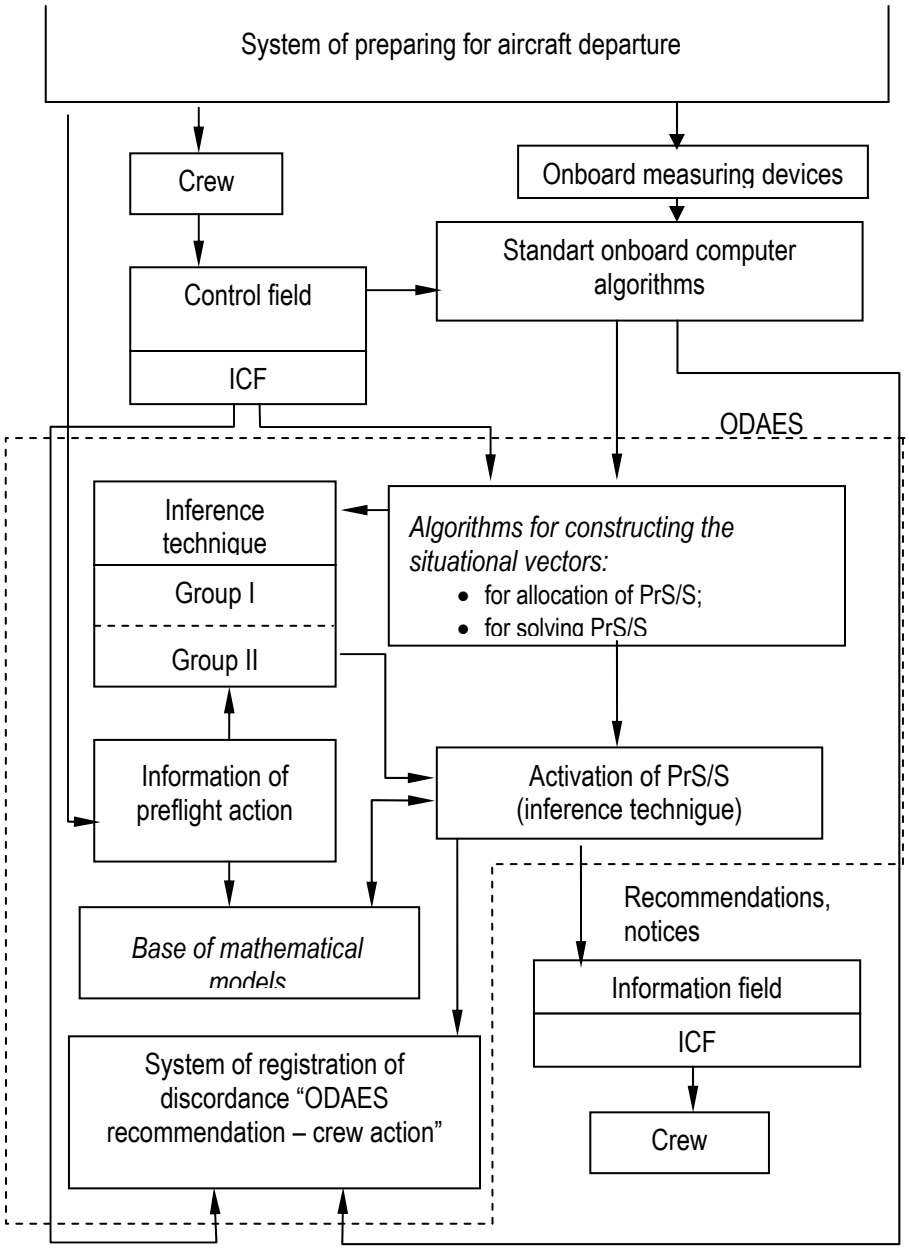


Fig. 1

2. Inference Technique Based on Algorithms for Multicriteria Choise

For a number of anthropocentric objects (for instance, piloted aircraft), problem subsituations aredistinguished, whose complexity does not allow one to formulate adequate optimization ma thematical problems, but for which, at the stage of preparation for the forthcoming operation session of the anthropocentric object, their crews produce the following:

a set of alternative methods for resolving the PrS/S alternatives $\{A_i\} = (A_b..., A_{i,}.... A_n)$ for resolving the PrS/S;

set of criteria $(Kj\} = (K,,..., Kj,.... K_t)$ for estimating the result of applying each of the alternatives (prefnce criteria).

As a rule, any particular PrS/S realized in the operation requires a certain adaptation of each alternative A_i. and, possibly, operative reestimation of the relative importance of the criteria *{Kj).*

We shall make the operative multicriteria choice of the most preferable alternative using the method of pairwise comparisons proposed by Saaty [8]. Below, we shall briefly dwell on its presentation.

2.1. Inference Technique Taking into Account the Current Preferences of the Crew.

Let a problem and some alternatives for its resolution be given. Suppose that, in each alternative, we are interested in some of its properties, which we will use for comparing the alternatives while choosing the most preferable one. We shall call these properties the comparison criteria. Suppose that we have several criteria which we use for comparing the alternatives. Also sup: that there is an expert (or experts)who has a suffitly definite opinion about the problem and the alternatives for solving this problem, which allows him to pairwise compare the alternatives according to each criterion.

The method of multicriteria choice of an alternative is a systematic procedure for hierarchical ordering of the elements of the problem. This method allows one to arrange the alternative according to their preferences with respect to a totality of specified comparison criteria. In order to constructively use this method in the inference technique, we present its justification based on studies by Saaty.

2.1.1. A method of pairwise comparison of alternatives with one preference criterion.

Let us distinguish one of the comparison criteria and pairwise compare the alternatives with respect to this criterion.

To formalize the procedure for choosing a preferable alternative, first, we consider the simplest example, namely, the problem of choosing the heaviest object (the comparison criterion) in a given set of objects A_1,..., A_i ..., A_n (alternatives) whose absolute weights (physical) are known. The weight of the object A_1, is w_1, the weight of the object A_2 is w_2, the weight of the object A_n is w_n. Since the object weights are known, to range them by weight, it is sufficient to arrange their weights in ascending order (to sort *n* numbers) and to choose the object with the greatest weight. Let us see in what ways one can choose the heaviest object.

We shall pairwise compare the weights of the objects A_i; i = 1 - n, recording the comparison results in the form of a table (matrix) (2.1), where $a_{ij} = \left(\frac{\omega_i}{\omega_j} \right) = \omega_{ij}$

One can see that such a table is a matrix $A(a_{i,j})$ whose entries $a_{ij} = w_i/w_j$ have the following properties: 1) all entries of the matrix are positive; 2) the principal diagonal of the matrix is filled with units; 3) the matrix is inversely symmetric with respect to the principal diagonal, i.e., a,j = 1/ $a_{i,j}$,4) the matrix has the transitivity property:$a_{i,k}$ $a_{k.j}$ = $a_{i,j}$, i.e., the product of the matrix entry $a_{i,k}$ located at the intersection of the *i*th row and *k*th column by the entry a_{kj} located at the intersection of the *k*th row and *jth* column is the entry $a_{i,j}$ located at the intersection of the ith row and jth column.

A matrix obtained in this way will be called the "ideal" matrix of pairwise comparisons.

For an ideal matrix of pairwise comparisons, the eigenvector (for convenience, normalized by unit) corresponding to its maximal eigenvalue has coordinates corresponding to the priorities of the considered objects compared by weight.

The formulated properties of an ideal matrix of pairwise comparisons can also be used for experimental matrices.

Let a researcher have a sufficient amount of qualitative information about some instances $A_{1,}$..., A_n compared by a certain criterion. Suppose that the matrix of pairwise comparisons of order *n* (experimental matrix) is composed for these instances. Naturally, it differs from the ideal matrix (calculated for this case). To estimate this difference, Saaty proposes the following procedure [8].

Calculate the consistency index (CI) of the experimental matrix CI = $\frac{\lambda_{\max} - n}{n-1}$, where λ_{max} is the maximal eigenvalue of the experimental matrix of pairwise comparisons and и is the order of this matrix.

One can immediately see that, if the experimental matrix is ideal, then CI = 0.

In [8], an estimate (the random consistency index) is introduced for an arbitrary square matrix of order *n,* which is positive, inversely symmetric, and has a unit principal diagonal. In the same paper, a table of the random consistency indices (RCI) of such matrices is given in the table from [8].

Finally, Saaty proposes to calculate the consistency ratio (CR): CR = $\frac{CI}{RCI}$. For 0 ≤ CR ≤ 0.10-0.15, he proposes to consider the experimental matrix as close to the ideal one and to use all useful properties of the latter for the experimental matrix. The main point of these properties is that, by the eigenvector of the experimental matrix corresponding to its maximal eigenvalue λ_{max}, one can judge the priorities of the instances compared according to the criterion considered.

2.2. An Approximate Method for Calculating the Maximal Eigenvalue and the Corresponding Eigenvector of the Matrix of Pairwise Comparisons.

It is known that the determination of the eigenvalues and the corresponding eigenvectors for matrices of a high order n is a fairly complicated problem. We propose an approximate method for determining them. This method is based on the properties of the ideal matrix of pairwise comparisons.

Let us use these properties of the ideal matrix of pairwise comparisons for arranging the arbitrary alternatives (instances) $A_1, ...,A_i, ...,A_n$ according to a certain criterion. Suppose that the matrix of pairwise comparisons of these alternatives is composed, and suppose that, checking its consistency, we find that this matrix is reasonably consistent; i.e., 0 ≤ CR < 0.10-0.15 for this matrix.

Then, the following Rule 1 is valid.

Rule 1. For a reasonably consistent experimental matrix of pairwise comparisons of the alternatives $A_1,...,A_i,...,A_n$ with respect to the criterion *Kj,* the vector of priorities is determined in the following way:

a) one composes the products of elements of every row of the matrix of pairwise comparisons and takes the *n*-th root from any of them, where *n* is the order of the matrix of pairwise comparisons

Kj	A_1 ... A_i ... A_n	
A_1	a_{11} ... a_{1i} ... a_{in}	$a_1 = \sqrt[n]{a_{11}...a_{1i}...a_{1n}}$
:	- - - - - - - - - - - - -	- - - - - - - - - - - - -
A_i	a_{i1} ... a_{ii} ... a_{in}	$a_i = \sqrt[n]{a_{i1}...a_{ii}...a_{in}}$
:	- - - - - - - - - - - - -	- - - - - - - - - - - - -
A_n	a_{n1} ... a_{ni} ... a_{nn}	$a_n = \sqrt[n]{a_{n1}...a_{ni}...a_{nn}}$

(2.1)

$$b_i = \sum_j a_{ij})$$

After normalizing the obtained vector ($a_1, ...,a_i, ...,a_n$) by unit, we obtain the required vector of priorities

$$s = \left\{ s_1 = \frac{a_1}{\sum_i a_i}; ..., s_i = \frac{a_i}{\sum_i a_i}; ..., s_n = \frac{a_n}{\sum_i a_i} \right\}$$

of the alternatives $A_1, ...,A_i, ...,A_n$ with respect to the criterion *Kj.*

Rule 2. The vector of priorities *s* can be used if the experimental matrix of pairwise comparisons is reasonably consistent.

On the basis of the property of an ideal matrix of pairwise comparisons, we estimate the maximal eigenvalue λ_{max} of the experimental matrix in the following way:

(a) find the sums of the entries in each column of the experimental matrix $b_i = a_{1i} + \ldots + a_{ji} + \ldots + a_{ni} = \sum_j a_{ji}$; i=1–n. As a result, we obtain a vector As a result, we obtain a vector b = (b_1, …, b_i, …, b_n);

(b) Multiply the vector *b* by s coordinate-wise and take the sum of the products obtained, setting

$$b_1s_1 + \ldots + b_is_i + \ldots + b_ns_n = \lambda max::;$$

(c) Calculate the consistency index $CI = \frac{\lambda_{\max} - n}{n-1}$ and the consistency ration CR = CI/RCI, where RCI is the random consistency index, which is taken from [8] for the appropriate "n".

(d) If $0 \le CR \le 0{,}10 - 0{,}15$ then the obtained experimental matrix of pairwise comparisons is reasonably consistent, and one can use the obtained vector of priorities "s".

2.3. Constructing the Matrices of Pairwise Comparisons (Experimental Matrices) in Practical Problems.

We have considered the case where the "exact" weight ratios are presented in the matrix of pairwise comparisons.

However, in practical problems, it is often impossible to exactly measure the results of pairwise comparisons. One of the methods for quantitatively estimating the ratio of alternatives (for the ideal matrix, this is the ratio of the object weights) is the use of a numeral scale.

The method used most often is the Saaty 9-mark scale. In the general form, it consists of the following [8].

Using a property (criterion) of alternatives being compared, one pairwise compares them: A_i is compared with A_j .

1.If A_i has no advantage (according to this criterion) over A_j, then the ratio (the analogue consiered is the weight ratio) is estimated by the number $\left(\frac{\varpi_i}{\varpi_j}\right) = 1$

……/

5. If A_i has an *absolute* advantage over A_j , then the ratio is estimated by the number $\left(\frac{\varpi_i}{\varpi_j}\right) = 9$

If one uses the Saaty scale, then the matrix of pairwise comparisons is inversely symmetric.

It is convenient to represent the results of pairwise comparisons in the form of matrix (2.1). In the upper left corner of the matrix, we write the name of the criterion, according to which the objects are pairwise compared. In the example considered above, it is the weight. The rows and columns of the matrix correspond to the names of alternatives. The vertical ordering of the names (the first column of the matrix) and the horizontal one (the first row of the matrix) are the same. Any cell of the matrix contains the result of the pairwise comparison of the alternative in the row with the alternative in the column. This result is estimated according to the Saaty scale.

2.4. Multicriteria Choice of an Alternative.

Let there be several alternatives A_1,..., A_i,.... A_n for the solution to a problem. These alternatives should be ordered according to criteria K_t,..., Kj,..., K_s.

For any criterion K_j, we estimate the weights of the alternatives A_1,..., A_i,.... A_n

$$S\,(Kj) = \{S_1\,(Kj), \ldots, Si\,(Kj), \ldots, Sn\,(Kj)\}$$

Using the method of pairwise comparisons and estimating the results by the Saaty scale, we determine the weights of the criterion significances $S = \{S_{lt} ...,Sj,...,S_s\}$ for the researcher.

Then, for any A_i, it is natural that its weight according to a criterion $S_i(K_j)$ is taken into account in the resulting weight for all criteria with the coefficient equal to the weight of this criterion's significance. The total weight (priority, rating) of the *i*-th object is determined by the formula

$$R_i = S_i(K_1)\,S_1 + \ldots + S_i(K_j)\,S_j + \ldots + S_i(K_s)\,S_s = \sum_{j=1}^{S} S_i(K_j)S_j$$

Finally, the alternatives $A_1,\ldots, A_i,\ldots A_n$ in the problem of multicriteria choice are arranged in accordance with the total weights. The alternative with the greatest total weight is the most preferable according to the whole set of comparison criteria.

2.5. The Structure of the Inference Technique in OOAES Constructed on the Basis of the Algorithm of Multicriteria Choice.

The knowledge base of OOAES contains a mathematical model for generating alternative versions for resolving problem subsituations of admissible types which are fed into OOAES at the stage of preparation for the operation session of the anthropocentric object (for piloted aircraft, when preparing for departure). The MM contains algorithms for determining the criterion values $K_j \in \{K_j\}$ for any alternative generated.

Current information characterizing the problem subsituation and admissible types of alternatives for resolving this PrS/S is supplied at the input of the MM. On the basis of admissible types of alternatives and the existing conditions for the occurrence of the PrS/S, in the MM, a complete set of alternatives $\{A_i\}$ of admissible types is generated, and, for any alternative $A_i \in \{A_i\}$, the numerical value of each criterion $K_j \in \{K_j\}$ is calculated. Moreover, an operative correction (by the crew or onboard computer) of the values of some coordinates of the vector SV(PrS/S-solution) characterizing the PrS/S is possible.

Thus, any alternative (from the set generated by the MM) is characterized by a vector whose coordinates are the numerical values of the criteria Kj.

On the basis of these vectors, the matrices of pairwise comparisons of the alternatives are constructed for each criterion.

We separately dwell on the matrix of pairwise comparisons of criteria. When constructing this matrix, one should maximally take into account the crew preferences formed by analyzing the existing current (for the operation session) situation. Taking into account that the crew's possibility of inputting this information is unlikely, one should maximally use the transitivity property of the matrix of pairwise comparisons when constructing this matrix.

After this, the vector of total weights of alternatives is calculated by the algorithm presented in Subsection 2.4. An example of implementing the inference technique described is given in [9].

3. Inference Technique Based on Precedents.

Such inference methods are used in problem subsituations, whose complexity does not allow one to con structively formalize them, but for which there is some experience (precedents) of their successful resolution.

One of difficulties of this approach is the correct choice of the coordinates $(x_1,\ldots,x_i,\ldots, x_n)$ of the situational vector SV(PrS/S-solution), both in their number and in the form of representation of each coordinate. The completeness of the description of the situational vector and the connection of a particular vector with a particular precedent is established by long-term cooperation with experts, who are actual bearers of this knowledge.

As a rule, the coordinates of the situational vector are linguistic variables.

3.1. Linguistic Variable as a Coordinate of the Situational Vector.

A linguistic variable is defined by Zadeh in [10] as a variable whose values belong to a specified set of terms or expressions of a natural language. The latter were also called terms.

To work with linguistic variables, one should represent each term via an appropriate fuzzy set [11]. The latter, in turn, is represented via a universal set (universe) and the membership function of the elements of the universal set to the considered fuzzy set.

The membership function takes the values in the interval [0, 1]. It quantitatively estimates the grade of membership of an element in a fuzzy set.

Note that both the universal sets and the membership functions on the set are specified on the basis of investigation results (together with experts) of the corresponding object domain.

For a large number of terms, their membership functions are usually specified in a unified form. Most often, this is a piecewise linear function.

3.2. Knowledge Matrices by Precedents.

Let a state of a problem subsituation be described by a situational vector with coordinates ($x_{1,...,}$ x_i, .., x_n) and each coordinate x_i be a linguistic variable with a set of terms $A_i = \{a_i^1,...,a_i^j,...,a_i^{K_i}\}$. For certain realizations of the situational vector, where each linguistic variable takes one of its possible values (a concrete term), there is a precedent of successful resolution of this PrS/S.

Suppose that a set d_j, j = 1, ..., p, of precedents is accumulated and each precedent is associated with a set of particular situational vectors, for which this precedent has been selected.

Let us construct the matrix of this correspondence (this matrix have the form of Table 1). We select the rows of the matrix corresponding to a precedent (the block of the precedent). Any row of the matrix is a concrete situational vector for which the corresponding precedent has been successfully realized in the past.

We enumerate the rows of the block of precedent d_j, with two indices: the first index is the number of the precedent (here, it is the number of the block), and the second index is the serial number of the situational vector in this block.

This matrix determines a system of logical propositions of the form *"if..., then..., else..."* For instance, the row j_1 of the matrix encodes the following proposition:

if $x_1 = a_1^{j1}$ and $x_2 = a_2^{j1}$ and ... and $x_i = a_i^{j1}$ and ... and $x_n = a_n^{j1}$, *then* d_j, (3.1)

else a similar proposition for the next row, etc.

The obtained system of logical propositions ordered in this way is called a fuzzy knowledge matrix or, simply, a knowledge matrix.

3.3. Algorithm for Calculating the Membership Function of Precedent d_t.

First of all, we present an algorithm [12] for determining the membership function μ_{dj} (x_1, .. x_i, .. x_n) of the precedent d_j interpreted as a fuzzy set on a universal set $Ud = Ux_1 \times ... \times Ux_i \times ... \times Ux_n$, where Ux_i is a universal set on which the terms of the linguist! variable x_i are defined, and U_d is the Cartesian product of the universal sets Ux_i .

Any logical proposition of the type (3.1) or, equivalently, any row of the knowledge matrix is a fuzzy relation of the corresponding fuzzy sets. For instance, for (3.1), this is $a_1^{j1} \times a_2^{j2} \times ... \times a_n^{jn}$.

Table 1.

Nos.	Coordinates of the situational vector					min	max	d
	x_1		x_i		x_n			
:	:	:	:	:	:	:	:	:
j_1	$(a_1^{j1})^*$	...	$(a_i^{j1})^*$	...	$(a_n^{j1})^*$	$\min_i (a_i^{j1})^*$		
:	:	:	:	:	:	... ...		
j_s	$(a_1^{js})^*$	...	$(a_i^{js})^*$	...	$(a_n^{js})^*$	$\min_i (a_i^{is})^*$	$\max_{js} \min_i (a_i^{js})^*$	μ_{d_j}
:	:	:	:	:	:	... ...		
j_{K_j}	$(a_1^{jK_j})^*$	...	$(a_i^{jK_j})^*$	...	$(a_n^{jK_j})^*$	$\min_i (a_n^{jK_j})^*$		
:	:	...	:	...	:	...	...	:

In accordance with [10, 11], the membership function of a fuzzy set generated by this fuzzy relation is $\mu_{a_1^{j1}}(x_1)$ $\wedge \ldots \wedge \mu_{a_i^{j1}}(x_i) \wedge \ldots \wedge \mu_{a_n^{j1}}(x_n)$, where "$\wedge$" we denote the "min" operation.

Analyzing the whole block of logical propositions related with precedent d_j (the block of the corresponding rows of the knowledge matrix), note that they form the union of the corresponding fuzzy sets generated while considering the rows of the selected block. In accordance with [10, 11], the membership function of this union, which is identified with the membership function of the precedent *dj,* is

$$\mu_{d_j}(x_1,\ldots,x_i,\ldots,x_n) = (\mu_{a_1^{j1}}(x_1) \wedge \ldots \wedge \mu_{a_i^{j1}}(x_i) \wedge \ldots \wedge \mu_{a_n^{j1}}(x_n)) \vee (\mu_{a_1^{jK_j}}(x_1) \wedge \ldots \wedge \mu_{a_i^{jK_j}}(x_i) \wedge \ldots \wedge \mu_{a_n^{jK_j}}(x_n))$$

where by "$\vee$" we denote the "max" operation.

Formally, this algorithm for determining the membership function of the precedent *dj* can be written in the following form:

(a) fix an arbitrary point $(x_1^*,\ldots,x_i^*,\ldots,x_n^*) \in U_{x_1} \times \ldots \times U_{x_i} \times \ldots \times U_{x_n}$;

(b) for any block of the knowledge matrix corresponding to d_j determine $\mu_{dj}(x_1, .. x_i, .. x_n)$ at this point according to the scheme of Table 1.

Note that, for any fixed point (x_1^*, ..., x_i^*, ..., x_n^*), the block of the matrix presented in Table 3 is numerical, because each term a_i^{js} from this block is replaced with the value of its membership function $(a_i^{js})^*$ calculated at the corresponding x_i^*. The operation $\min\limits_i a_i^{js}$ is performed with the numbers located in rows "i," 1≤ i≤ n, and the minimal number in the corresponding row is placed in the column "min."

operation $\max\limits_{jS} \min\limits_i a_i^{js}$ selects the greatest of the row minima obtained for $1 \le j_s \le K_J$. This number is the value of the membership function $\mu_{dj}(x_1, ..., x_i, ..., x_n)$ at the fixed point (x_1^*, ..., x_i^*, ..., x_n^*). Performing this calculation for every point of the universal set, we obtain the membership functions that interest us.

3.4. Algorithm for Choosing a Precedent when Observing a Situational Vector with Quantitative Coordinates.

When observing a situational vector [12] with quantitative coordinates (all coordinates of the vector are measured by numerical scales), in order to select the most preferable precedent, it is not necessary to completely determine the membership functions $\mu_{dj}(x_1, .. x_i, .. x_n)$ on the whole set of points of the universal set. It is sufficient to calculate their values only for fixed numerical values of the coordinates of the vector, which is obtained by us as a result of the observation. For this purpose, we should use the algorithm from Subsection 3.3 once, taking the coordinates of the observed situational vector as (x_1^*, ..., x_i^*, ..., x_n^*).

As a result, for any precedent *dj,* we obtain a number $d_j(x_1^*, ..., x_i^*, ..., x_n^*)$, which is the grade of membership of d_j to the point (x_1^*, ..., x_i^*, ..., x_n^*).

Starting from this interpretation, the most preferable precedent for resolving the observed PrS/S is the precedent d_j^* such that

$$d_j^*(x_1^*,\ldots,x_i^*,\ldots,x_n^*) = \max_{1 \le j \le p} d_j(x_1^*,\ldots,x_i^*,\ldots,x_n^*).$$

Conclusion

Three types of inference techniques are presented. They are constructive for resolving problem subsituations of typical situations of anthropocentric object functioning (for instance, flights of piloted aircraft).

The first type of inference technique (on product rules) is based on results of the mathematical investigation of the PrS/S and, to a certain degree, on the knowledge of experts in the object domain under consideration. This technique is widely used by designers of the knowledge bases of the first Russian and foreign OOAES.

The second type of inference technique is based on the use of the algorithm for multicriteria choice of an alternative. The technique is directed to the substantial use of the results of the preparation for the operation

session of the anthropocentric object (prior information) and crew preferences formed while operatively analyzing the situation existing in the current operation session. The considered example of the use of this technique allows one to hope that it can be successfully employed in practice when designing the real knowledge bases of OOAES.

The third type of inference technique has been studied only theoretically and has not been tested on practical examples. It is directed to be used in the inference technique of precedents and is based on the algorithms for choosing a solution on the basis of the knowledge matrix. These algorithms are successfully applied in diagnostic problems.

The choice of the type of inference technique for constructing in OOAES a recommendation for resolving a problem subsituation of a concrete type depends on its complexity and on the possibility of adequately formalizing it mathematically.

References

1. Fedunov, B.E., Problems of the Development of Onboard Operative and Advising Expert Systems for Anthropocentric Objects, *Izv. Ross. Akad. Nauk, Tear. Sist. Upr.,* 1996,№ 5.
2. Fedunov, B.E., Constructive Semantics of Anthropocentric Systems for Development and Analysis of Specifications for Onboard Intelligent Systems, *Izv. Ross. Akad. Nauk, Tear. Sist. Upr.,* 1998, ;№. 5.
1. Gavrilova, T.A. and Chervinskaya, K.R., *hvlechenie istrukturirovanie znanii dlya ekspertnykh sistem* (Extraction and Structurization of Knowledge for Expert Systems), Moscow: Radio i Svyaz', 1992.
2. Pospelov, D.A., *Situatsionnoe upravlenie: teoriya i pmktika* (Situational Control: Theory and Practice), Moscow: Nauka, 1986.
3. Fedunov, B.E., The Optimization Models for Taking the Decision in the Algorithmic and Indicational Support System Designing: Systems, Analysis, Modeling Simulation, *J. Mathematical Modeling and Simulation in Systems Analysis,* 1995, vols. 18, 19.
4. Egorova, E.T. and Fedunov, B.E., Minimax Problem of Placement and Tracing Points in a Circle and Potential Capabilities of Measurement Devices, *Jzv. Ross. Akad. Nauk, Tear. Sist. Upr.,* 1996, №. 3.
5. 7. Demkin, M.A. and Fedunov, B.E., A Fragment of the Knowledge Base OOAES TS Ensuring Evasion of the OOAES Carrier from a Pursuer, *Iskusstvennyi intellekt* (Artificial Intelligence), NAN Ukrainy, 2001, № 3.
6. Saaty, T., *The Analytic Hierarchy Process. Planning,Priority Setting, Resource Allocation,* New York: McGraw-Hill, 1980.
7. Musarev, L.M. and Fedunov, B.E., The Structure of Target Allocation Algorithms on Board the Commander of an Aircraft Group, *Izv. Ross.Akad. Nauk, Teor. Sist. Upr.,*2001, № 6.
8. Zadeh, L., *The Concept of a Linguistic Variable and Its Application to Approximate Reasonong,* New York:Elsevier, 1973.
9. Kaufmann, A., *Introduction a la theorie de sous-ensem bles flous,* Paris: Masson, 1977.
10. Rotshtein, A.P., Intelligent Technologies of Verification, Vinnitsa: Universum, 1999.

Author information

B. E. Fedunov: Stale Scientific Research Institute for Aviation Systems, ul. Viktorenko 7, Moscow, 125319 Russia; e-mail: boris_fed@gosniias.msk.ru

THE INTELLIGENT SYSTEM OF THE HEARING INVESTIGATION

Filatova N.N., Strelnikov I.N., Grigorieva O.M., Bodrin A.V., Kalugniy M.V.

***Abstract**: This paper describes a prototype of the intelligent system of the hearing investigation developed by the Tver State Technical University. The problem of automatic diagnostics, considered as the recognition problem of object not completely determined on set of the diseases classes descriptions, is discussed. The management strategy of the hearing investigation is proposed.*

***Keywords**: hearing investigation, recognition of object.*

1. Introduction

On modern stage an integration problems of patient condition investigation strategies and diseases diagnostics are most actual in the field of medical informatics. Realization of such strategy allow raise accuracy of the diagnostics and shorten time and examination cost. In report the realization way of similar strategy are considered as example of the creation of intelligent integral system, controlling the investigation process and diagnostics of the hearing organs. Majority of strategies use the hearing indirect estimations resulting from patient sensations. In this connection investigation strategies adaptation must be provided to patient in this area of medicine. Necessary to take into account its intellect possibilities and ability of the correct self-awareness. This particularity and the using of qualitative factors for the hearing estimation intensify the subjective nature of the hearing organs investigation results. The latter sufficiently complicates the task of the disease diagnostics because practically only subjective and qualitative information is attracted for its solution.

2. The Composition of system and information models.

The intelligent system of survey and the hearing diagnostics includes five subsystems: the information, the expert- diagnostic, the investigations, the planning and the training. In report main attention is spared of the investigations subsystem and subsystem of diagnostics.

The information subsystem is meant for automated record-keeping of the patient's history as well as organization of interconnection as of all software blocks.

The expert diagnostic subsystem is created on the basis of the concept of intelligent expert systems and is meant for automated formation of diagnostic conclusion hypotheses by means of patient's condition estimation. The main function of the subsystem are: results analysis of separate tests and their collection; analysis and recognition of the audiograms type; patient model analysis and search in the knowledge-base of relevant rules; formation of the diagnosis hypothesis; output of preliminary conclusion as of the hearing investigation results; formation of the conclusion about the disease development course based on the investigation results for the whole observation period; connection with database of diseases histories.

The audiometric investigations subsystem is meant for automated testing of the patient by software realization of methods of tone and speech audiometry.

The training subsystem allows automate the procedures of explanation to patients of strategies of the hearing investigation.

The subsystem of treatment planning is meant for information support of remedial measures, automatic generation and successive correction of the investigation plan and treatments of patient.

3. Management of investigations and stating the diagnosis

The strategy and management algorithms of interconnected processes of investigation and the hearing diagnostics are created on the base of gradual development process analysis of physician’s knowledge (the

systems) about patient. Considering its temporary distribution, problem of automatic diagnostics can be referred to the recognition problem of object not completely determined on set of diseases classes descriptions.

Let, $S_i^* \subseteq S_i \quad , S_i \in V_{PN}$, where S_i - is a cortege of diagnostic signs, assigning description of patient on overbroad investigation, S_i^* - is a cortege of diagnostic signs, assigning patient description as a result of not completed investigation, PN_i - is a set of symptoms classes descriptions (in private event a class can define the disease).

Moreover, if $(\exists j) \quad S_i^* \bigcap S_j \neq 0 \quad \text{and} \quad S_i \bigcap S_j \neq 0$, that obviously on given step of investigations S_i^* there are symptoms, which typical of not single disease.

To recognize of object, not completely determined, necessary to find classes most close to S_i^*. The methods and algorithms of decision of such problems are defined by the manners of assignation and determining of measures of resemblance and differences in space of signs values.

Let measure of resemblance between objects S_i and S_j will be nonnegative material function $C(S_i, S_j)$, which has limit, increases with growth of objects vicinity and possesses the following characteristics [Andreischikov,1998]:

$$\left\{ \begin{array}{c} 0 \le C(S_i, S_j) \le 1 \text{ when } \quad i \neq j \\ C(S_i, S_j) = 1, \quad \text{full resemblance} \\ C(S_i, S_j) = \quad C(S_j, S_i) \end{array} \right\} \tag{1}$$

Let give the similar determination of the difference measure $D(S_i, S_j)$:

$$\left. \begin{array}{c} D(S_i, S_j) \ge 0, \quad i, j \in \Im \\ D(S_i, S_j) = 0 \qquad full\ resemblance \\ D(S_j, S_i) = D(S_i, S_j) \\ D(S_i, S_j) \le D(S_i, S_k) + D(S_k, S_j) \end{array} \right\} \tag{2}$$

Form of concrete function C (S_i, S_j) is know to depend on scales of signs measurements and purposes created categorizations [Larischev, 1998; Zagoruko, 1999]. The resemblance characteristic on the Chekansky-Serensen's formula is measured by the simple equation:

$$C(S_i, S_j) = \frac{2m(S_i \bigcap S_j)}{m(S_i) + m(S_j)} \tag{3}$$

where: $m(S_i)$ - is the number of binary signs, entered in object description, $m(S_i \bigcap S_j)$ - is the number of the general binary signs, presented in descriptions S_i, and S_j, $m(S_i \bigcup S_j)$ - is the total number of binary signs in descriptions S_i, and S_j.

The conclusions from correlation (3), got on artificial samples of objects, well coincides with qualitative interpreting of resemblance notion. Approximately such result is got on the base of Kulchinsky's formula:

$$C(S_i, S_j) = \frac{1}{2} \cdot m(S_i \cap S_j) \cdot \left[\frac{1}{m(S_i)} + \frac{1}{m(S_j)} \right] \tag{4}$$

Coincided signs are taken into account only once in formula of Gakkar:

$$C(S_i, S_j) = \frac{m(S_i \bigcap S_j)}{m(S_i \bigcup S_j)} \tag{5}$$

This measure, calculated by modified formula, is provided in Polovinkin's work [Polovinkin 1998]:

$$C(S_i, S_j) = 1 - \left[\frac{m(S_i \cup S_j) - m(S_i \cap S_j)}{m(S_i \cup S_j)} \right] \quad (6)$$

Estimation of resemblance (5 or 6) in contrast with (3) is less sanguine.

The Gakkar's adjustments are preserved in Sokal-Sinet's formula, as well as attempt to intensify importance underbar (discriminating) signs is undertaken:

$$C(S_i, S_j) = \frac{m(S_i \cap S_j)}{2m(S_i) + 2m(S_j) + 3m(S_i \cap S_j)} \quad (7)$$

The quantitative estimation of resemblance, formed by (7), is else lower, then on Gakkar's formula.

The estimation of function, received on Andreev's formula, several is uprated:

$$C(S_i, S_j) = \frac{4m(S_i \cap S_j)}{m(S_i) + m(S_j) + 2m(S_i \cap S_j)} \quad (8)$$

To estimate of differences between objects we use the metrics of type:

$$D(S_i, S_j) = m(S_i) + m(S_j) - 2m(S_i \cap S_j) \quad (9)$$

The formulas (1- 9) install the measure of resemblance between objects described by conjunctive functions. In the more general event the vicinity measure determination of object towards some class is described by conjunctive-disjunctive function of type:

$$PON_i = PON_i^k \;\&\; PON_i^D \quad (10)$$

Conditional distance between single object and class is valued by hierarchy of function δ_i:

$$\delta_i = \sum_j, \sum_j = \sum_{j-1} + r_i, \qquad j = \overline{1, m}$$

where: under j=1 $\sum_j = 0$, m – is the number of attributive features in object model:

$$S_i = \mathop{\&}_{j=1}^{m} P_j^0$$

$$(\forall j) \quad r_1 = \begin{cases} 1, & P_j^0 \cap PON_i \neq 0, \quad P_j^0 \in PON_i \\ 0, & P_j^o \cap PON_i = 0 \end{cases} \quad (11)$$

Thus, δ_1 characterizes amount of binary signs (values of multivariate signs) from model of object S_i, which entered into description of class PON_i.

Restriction (10) imposing on structure of class description, reinforcement of function δ_i is provided:

$$(\forall j)\, r_2 = \begin{cases} 1, & P_j^0 \cap PON_i^k \neq 0, \quad P_j^0 \in PON_i \\ 0, & in\ rest\ events \end{cases} \quad (12)$$

Consequently, δ_2 characterizes amount of binary signs (values of multivariate signs) from model of object S_i , which fall into description of function PON_i^k only, that is to say these signs are general signs for all objects of class PON_i.

The relative factor of use frequency of any feature P_j^0 in class description is defined by means of:

$$\delta_3 = \frac{\delta_1}{m}$$

Since, δ_2 characterizes the coincidences number in descriptions of object and function PON^k , that δ_4 will define respective relative factor: $\delta_4 = \delta_2 / n_1$.

If $PON_r^k = \&_{i=1}^{m} P_{ij}$ *and* $PON_r^k \subset PON_r$ *and* $S_r = \&_{i,j} P_{ij}$, that

$$(\forall j)\, r_5 = \begin{Bmatrix} 1, & (\exists\, k) \quad P_{ik} \in PON_r, \quad P_{ij} \neq P_{ik}, \quad P_{ik} \in \overline{P}_i \\ & \left(\overline{P}_i \quad set\ of\ multi\,\mathrm{var}\,iate\ sign\ value\right) \\ & 0, \quad in\ rest\ events \end{Bmatrix} \tag{13}$$

Consequently, δ_5 will characterize number of multivariate signs from model description of object S_i, values of which fall into function PON $_r$.

The Analysis (3.28 - 3.30) shows that given features allow to value:

- δ_1 - is the absolute vicinity of object S_i toward class PON_i on number of coincided features;
- δ_2 - is the local vicinity of object S_i toward class PON_i in the field of strict generality, (characterizes number of signs P_j , entered in the strict generality in description of analysed class);
- δ_3 - is relative vicinity of object S_i toward class, (takes into account coincidence of features and length of description of analysed class);
- δ_4 - is the relative local vicinity of object S_i toward analysed class PON_i;
- δ_5 - is the typical vicinity of object S_r toward class PON_r on count of alike signs (in multivariate event $f(P_{ij})$ on background of possible variations their value).

For example, we have certain factor δ_{ij}, characterizing vicinity measure not completely determined situations S toward some class of objects V_j. Let, a rule: " if δ_{ij} = 1, that object belongs to the class $S_i^* \in V_j$; if δ_{ij} = 0 , that situation has nothing in common with class V_j ", is equitable. If δ_{ij} < 1 , that object S_i^* has some general signs with class V_j . Logical function PN_i , which takes the different true values on set $\{S_k\}$, is know to exist and to be determined obviously. Assume that classes PN_r and PN_t are most close for S_i^* in size δ_{ij}. These classes is know to be not cross, and at least one sign value, falling into description as PN_r so and PN_t, exists.

Situation vicinity degree S_i^* to each class can be different in general event. Assume that $\delta_{ir} > \delta_{it}$, i.e. S_i^* closer to class Vr. Then inverse correlation $d_{ir} < d_{it}$ for co-equivalent feature $d_{ir} = 1 - \delta_{ir}$, defining differences between object and class, is equitable.

The Features, being general features for class PN_r and situation S_i^*, defines the subset $I = \overline{PN_r} \cap \overline{S_i^*}$, $\overline{PN_r} \setminus I = D_i$, where $\overline{PN_r}$, $\overline{S_i^*}$ - is the set of signs values, being arguments of functions PN_r and S_i^* accordingly. The set D unites values of signs, falling into class determination PN_r , but not being present in S_i^* . The separation of features, distinguishing situation, not completely determined, S_i^* from class PN_r , allow build the management strategy of investigation (and diagnostic) on base of minimization of vicinity function δ_{ij}.

Proposed strategy of investigation management includes the following stages:

- Creation of i description of patient condition in the manner of situation, not completely determined and given in signs space.
- Separation of classes subset, closed to under investigated object S_i^*; degree of vicinity is the adjusted parameter (δ_{ij})

- Analysis of classes specifiers for the reason revealing the set of object signs, not determined within the framework of model of i approximation (S_i^*).
- The forming of managing influence:
  - the choice and activation of investigation strategy for determining certain signs
  - the forming of information messages with lists of the not yet defined features and recommendations on the plan of investigation for the physician.
- The correcting of the patient description: $S_i^{**} = S_i^* \bigcup P_m^0$.
- The checking of vicinity functions: if δ_{ij} = 1 ($S_i^{**} = S_i$), that purpose of management is reached, otherwise cycle is repeated with the preliminary actualization of all subsets.

Conclusion

Designed algorithm of management allows integrate the procedures of investigation and diagnostics, shortens the number of features defined in the course of investigation, brings about reduction of temporary expenses.

The first version of intelligent system of investigation and diagnostics of the hearing passes the test in polyclinic of regional clinical hospital in Tver.

Bibliography

[Andreischikov, 1998] Андрейчиков А.В. Андрейчикова О.Н. Компьютерная поддержка изобретательства (методы, системы, примеры применения).- М.: Машиностроение. 1998.-476с.

[Larischev, 1998] Ларичев М.И. Интеллектуальные системы диагностики.// Сб.труд.Межд. конф. КИИ-98.

[Zagoruko, 1999] Загоруйко Н.Г. Прикладные методы анализа данных и знаний. Новосибирск: Изд. Ин-та матем. 1999.-270с.

[Polovinkin,1988] Половинкин А.И. Основы инженерного творчества: Учеб.пособие для студентов вузов. –М.: Машиностроение/ 1988.-368с.

Author information

Filatova N.N., Strelnikov I.N., Grigorieva O.M., Bodrin A.V., Kalugniy M.V. - The Tver State Technical University; the post address: 170023 Tver, Lenin's prospectus, 25; e-mail: nfilatova99@mail.ru.

SELECTION OF THEMATIC NL-KNOWLEDGE FROM THE INTERNET

V.Gladun, A.Tkachev, V.Velichko, N.Vashchenko

***Abstract**: The paper deals with methods of choice in the INTERNET of natural-language textual fragments that are relevant to a given theme. Relevancy is estimated on the basis of semantic analysis of sentences. Recognition of syntactic and semantic connections between words of the text is carried out by the analysis of combinations of inflections and prepositions, without use of categories and rules of traditional grammar. Choice in the INTERNET of the thematic information is organized cyclically with automatic forming of the new key at every cycle when addressing to the INTERNET.*

***Keywords**: semantic analysis, information search, INTERNET.*

1. The purposes and base ideas

Among various variants of practical use of storehouses for the textual information the necessity to find the information having thematic unity prevails. These are needs of a scientist, a journalist, a politician, an official, a writer, a student. Usually the theme arises as one or several concepts, some initial situation having a number of blank valences and situational roles which serve as reference points for search of the new relevant information. The new information gives rise to new directions of search. This complex, sometimes psychologically painful, creative process requires the automated support. Thematic search needs laborious work with the texts stored in libraries, archives, the INTERNET, textual databases. Difficulty of this work consists, in particular, in necessity to select not the whole texts, but relevant to the theme fragments of texts. The contents of many texts is an interlacing of a number of themes. Thus, a problem arises of search inside textual documents of fragments, relevant to the given theme.

In the paper methods, software and results of selection of the thematic textual information are considered. The researches submitted in the paper continue the works published in [1-3].

The solving of the problem unites the following actions:

1) selection of texts or fragments of texts relevant to an investigated theme;
2) selection from the relevant information of the most important, first of all, such which defines and connects the most essential terminology of a theme;
3) representation of the chosen information in the user-friendly form.

Implementation of the specified actions is based on the following ideas:

1) to focus a technique of selection of the thematic textual information on the INTERNET, as on the most full storehouse of the textual data;
2) to combine search by key words with the semantic analysis of NL-texts;
3) to use semantic criteria for selection of the most important thematic information;
4) to organize automatic cyclic process of key words formation to investigate the theme as complete as possible.

2. A technique

The initial stage of the thematic information selection consists in search of textual documents in the INTERNET using the given key. Existing methods of information search in the INTERNET give out a lot of unnecessary for a user “garbage” information which filtration takes too much time. The way out consists in use of the semantic criteria providing selection of the most essential characteristics of concepts concerning which the information is gathered.

The offered method is based on the assumption, that the most important user information is contained in *kernel constructions* of sentences. The term “kernel constructions” is used in transformation grammar for designation of simple base judgment by which transformation the sentence as a whole is formed. In our case the kernel construction consists of a subject, a predicate and a link.

The method represents cyclically repeating sequence of the following operations:

1. Selection of the given quantity (parameter) of texts using a key. A set of used search systems is unlimited. Now the program can use the following search systems: Yandex, Rambler, Meta-Ukraine, Aport, Google.
2. Selection in the found texts of the sentences containing a given key.
3. Selection in set of the sentences, that were chosen in item 2, the sentences containing kernel constructions. For item 3 performing the natural-language semantic analyzer is used.
4. Formation *of n-step expansions of the kernel* of the selected sentences. *n*-step expansion of the kernel is a part of the sentence containing its kernel, and also the words connected in a tree of dependencies with elements of the kernel by paths which length does not exceed *n. n* is a user-given parameter.

The item 4 is performed on the basis of the semantic analysis of the sentence.

5. Selection in the set of the sentences chosen in item 3, such sentences in which *n*-step expansions of the kernel contain the given key.
6. Formation of a new key on the basis of the analysis of semantic representations of before selected sentences. Transfer to the item 1.

The initial key word is given by a user. New keys on the subsequent cycles of the algorithm are chosen among *terms* that are significant words used only within the limits of investigated domains. The terms are marked in the dictionary.

When choosing a new key, the degree of its relevance to the given theme is taken into account. The relevance is defined on the basis of results of semantic analysis of sentences. At the following cycle of the algorithm the term, that was not used earlier and has the greatest relevancy coefficient, is chosen as a key.

After a choice of a new key actions 1 - 6 are repeated.

3. The semantic analysis

The basic operation of the semantic analysis of natural-language texts is recognition of the syntactic and semantic relations connecting words of the text. Recognition of relations is carried out on the basis of their descriptions (models). Such models are necessarily present at all methods of the analysis though it is not always obvious. In the majority of the analysis methods the process of recognition of relations is preceded with translation of initial natural-language representation of relations to be recognized in the language of categories of traditional grammar (gender, case, time, etc.). Rules of recognition of syntactic and semantic relations operate with grammatical descriptions of words. Binding to grammatical descriptions of elements of the text results in the following imperfections: heterogeneity of ways of processing separate words and word combinations; bulkiness of processing; complexity of adaptation to changes of lexicon and a user's domain; laboriousness of the research. Meanwhile, transition to grammatical descriptions is not an obligatory condition for performance of the semantic analysis of natural-language texts. The information necessary for recognition of syntactic and semantic relations is contained directly in the text. As a proof to that, there are "human" processes of the analysis of the natural-language texts, which are not connected with grammatical categories and rules. Therefore, it is competent another approach based on use of conformity between relations and means of their expression in natural-language texts. Recognition of syntactic and semantic connections between words is carried out by the analysis of combinations of inflections and prepositions, without using categories and rules of traditional grammar. By virtue of its basic features, such approach allows to exclude the imperfections named above.

Models of relations in which elements of natural-language texts are used for recognition of syntactic and semantic relations, we shall refer to as *lexical models of relations*. The algorithm of the semantic analysis of natural-language sentences on the basis of lexical models of relations is described in [1-3].

4. Implementation and results

The structure of the program complex realizing processes of thematic knowledge formation consists of the programs which are carrying out the following actions:

1. Selection in the INTERNET of the textual fragments containing a given key.
2. Formation of semantic representations of sentences (the linguistic processor).
3. Selection of sentences, relevant to a theme, on the basis of the analysis of semantic representations of sentences.
4. Choice of a new key.

At the present time lexical data and knowledge bases of the complex are created for Russian language.

As a result of working a program complex the text is formed which consists of separate sentences that are relevant to a theme designated by an initial key which is given by a user. For each sentence, the address of corresponding document is indicated. The set of sentences selected from one document allows to generate a conception about its thematic relevance as a whole. The high level of relevance of the document may induce a user to choose this document for detailed studying. The set of all selected sentences throws light on an investigated theme as a whole. The degree of completeness of the selected information on a theme depends on efficiency of the used search machine and quantity of the texts chosen in the INTERNET. Experience of the complex exploitation shows that the set of sentences selected by a program on the basis of the thematic analysis well correlates with result of "manual" selection of "useful" sentences by an end user. The complex provides the high degree of elimination of the information that is unnecessary for a user.

Conclusion

Above described method of thematic selection of information can be used for the information search not only in the INTERNET, but in any textual databases. We also consider it as the instrument for creation of ontology's. The merit of the method is effective filtration of the information on the basis of criteria of relevancy to the given theme that is obtained at the cost of semantic analysis of sentences and a cyclic process of automatic selection of a new key at every cycle. The method allows comparatively simple adaptation to changes of a text language.

The literature

1. Gladun V.P. Processes of formation of new knowledge. - Sofia: СД "Педагогь". 1994. - 192p. (in Russian).
2. Gladun V.P. Planning of decisions. Kiev: Наукова думка, 1987.-168p. (in Russian).
3. Gladun V.P. Natural language in purposeful systems.//DIALOG-2000. Applied problems. 2000, p.99-102. (in Russian).

Author information

Victor Gladun - V.M.Glushkov Institute of cybernetics of NAS of Ukraine, Prospekt akad. Glushkova 40, 03680 Kiev, Ukraine; e-mail: glad@aduis.kiev.ua

Alexander Tkachev - V.M.Glushkov Institute of cybernetics of NAS of Ukraine, Prospekt akad. Glushkova 40, 03680 Kiev, Ukraine; e-mail: glad@aduis.kiev.ua

Vitaly Velichko - V.M.Glushkov Institute of cybernetics of NAS of Ukraine, Prospekt akad. Glushkova 40, 03680 Kiev, Ukraine; e-mail: glad@aduis.kiev.ua

Neonila Vashchenko - V.M.Glushkov Institute of cybernetics of NAS of Ukraine, Prospekt akad. Glushkova 40, 03680 Kiev, Ukraine; e-mail: glad@aduis.kiev.ua

THE STRUCTURE OF INFORMATION DIALOGUES: A CASE STUDY

M. Koit

Abstract: *In the paper we consider the structure of information dialogues. Our study is based on Estonian dialogue corpus which contains two kinds of dialogues – transcriptions of spoken conversations, and dialogues collected with the Wizard of Oz method. We are using two ways for describing the structure of dialogues – a typology of dialogue acts, and a system of communicative strategies. We depart from the notion of communicative strategy introduced by Kristiina Jokinen in her Constructive Dialogue Model. The analysis of our empirical material shows that people are using similar communicative strategies in telephone conversations and computer interactions. In the same time, the structure of human-human conversation is much more complicated.*

Keywords:*. computer intellectualization.*

Introduction

Estonian dialogue corpus consists of two kinds of dialogues. Firstly, 255 spoken dialogues are recorded and transliterated by the transcriptional system of conversational analysis (Jefferson 1979). 150 of the dialogues are telephone conversations where a person calls an office (railway station, bus terminal, travel agency, etc.) aiming to get some information. The remained 100 are face-to-face conversations. Secondly, we have collected 20

dialogues by the Wizard of Oz (WOZ) method. All the WOZ dialogues are information requests. The participants of our WOZ experiments were allowed to ask questions about bus schedule in Estonia and ship or plain traffic between Estonia and Finland. Therefore, we have a reasonable number of information dialogues in our corpus.

Building our corpus, we have two goals. The first goal is studying of spoken human-human conversation, and the second is modelling of human-computer interaction. Our further aim is to build an experimental dialogue system which could act as a rational agent and provide the needed information to the user. The dialogue system will integrate several language technology modules built up for Estonian so far (morphological and syntactic analysis, text-to-speech synthesis etc.). To work out a dialogue manager, we are studying the structure of our information dialogues.

There are several ways to describe the dialogue structure. From one side, we can use a system of dialogue acts and represent dialogue as a sequence of such acts. From the other side, communicative strategies for achieving certain communicative goals can be found in dialogues, and dialogue can be represented as implementation of the strategies. Both of these developments are methods for expressing and achieving the coherence of dialogues.

Typology of Dialogue Acts

There exist several typologies of dialogue acts. The first well known typology was worked out by J. Sinclaire and M. Coulthard on the ground of the study of real dialogues (Sinclair, Coulthard 1975). The system of dialogue acts was further developed by A.-B. Stenström (Stenström 1994). Several researchers are considering practical problems of dialogue acts determination during the last decade – corpus linguists, discourse and conversation analysts, language technologists (Hakulinen 1989; Allwood et al. 2000; Stolcke et al. 2000; Jokinen et al. 2001).

Choosing a dialogue act mark-up system we have had two goals: to study spoken human-human conversation and to model human-computer interaction. We started with analysis of existing dialogue act systems and typologies (Klein, Soria 1998, Francis, Hunston 1992, Stenström 1994, Dybkjær 2000). It proved difficult to take over a ready-made typology because most of them are domain-oriented (eg. furnishing an apartment, guessing a journey on the map, determining a meeting, etc.) Therefore, we decided to work out our own typology. We departed from the Stenström system which is based on conversation analysis.

There are 140 dialogue acts in our system divided into 8 groups:
1) rituals – greeting, introducing, etc.;
2) acts for re-structuring of conversation, with help of which the speaker starts a new topic or changes the type of conversation;
3) acts for exchanging of turn-takings, with help of which the speaker is asked to continue, or the existence of contact is checked;
4) repairing acts, with help of which partners are solving communication problems;
5) directive acts for giving and receiving of commands, requests, etc.;
6) questions and answers – pairs of acts, with help of which one partner asks a question and another answers it;
7) acts for taking up of attitudes, with help of which one partner represents an attitude (belief, evaluation, charge) and another responds it;
8) the last group contains the remaining acts (additional information, argument, conclusion, promise, acknowledgement, signal of new information, etc.).

The acts from all the groups, except of the last, can form adjacency pairs. For that reason, they are divided into 2 sub-groups: the first and second parts. The first parts are used to give commands, ask questions, etc. The second parts express reactions to commands, answers to questions. Acts from the 8th group can supplement both the first and second parts.

A simplified formal grammar determining our dialogue acts system is as follows (cf. Koit 2001). The terminals (dialogue act names) are written in capitals.

```
interaction::= (transaction)+
transaction::= (exchange)+
exchange::= organisational-exchange | conversational-exchange
organisational-exchange::= ritual | repair | CONTINUER
ritual ::= CALL  RESPONDING-THE-CALL | GREETING  RESPONDING-THE-GREETING
```

```
| THANKING RESPONDING-THE-THANKING | LEAVE-TAKING RESPONDING-THE-LEAVE-
TAKING
  repair::= hearer-initiated-repair | self-repair
  hearer-initiated-repair::= INITIATING-REPAIR CARRING-OUT-REPAIR |
INITIATING-REPAIR CARRING-OUT-REPAIR EVALUATION
  initiating-of-repair ::= NON-UNDERSTANDING | RE-QUESTION | SPECIFYING-
CONDITONS-OF-THE-ANSWER
  self-repair::= REFORMULATION
  conversational-exchange::= directive-exchange | question-exchange
  directive-exchange::= directive's-pre-member directive's-re-member
  directive's-pre-member::= ORDER | REQUEST | PROPOSAL | WISH | CALL-UP |
OFFER | REQUEST-TO-WAIT
  directive's-re-member::= FULFILMENT | REFUSAL | AGREEMENT | POSTPONING-
THE-ANSWER | FULFILMENT-WITH-RESERVATIONS | YOU-ARE-WELCOME
  question-exchange::= question's-pre-member question's-re-member
  question's-pre-member::= CLOSED-YES/NO-QUESTION | OPEN-YES/NO-QUESTION |
WH-QUESTION | SPECIFYING-THE-CONDITIONS-OF-ANSWER
  question's-re-member::= AGREEMENT-(YES)| AGREEMENT-(NO) | NON-AGREEMENT |
open-answer | POSTPONING-THE-ANSWER | ANSWER-AS-AN-ALTERNATIVE
  open-answer::= GIVING-INFORMATION | INDICATING-THE-ABSENCE-OF-INFORMATION
```

Our typology does not allow to annotate dialogues on several levels as it is possible, for example, in DAMSL (Allen et al., 1997). However, some levels can be differentiated indirectly. Communicative status is indicated by the dialogue act REFUSAL which marks a non-interpretable or unfinished utterance. Information level is expressed by the conversational exchanges (as opposite to organisational ones). The role of forward-seeking functions is played by the first parts (pre-members) and the role of backward-seeking ones by the second parts (re-members) of adjacency pairs. Our scheme is more detailed as DAMSL. For example, the group of rituals consists of 34 acts (there are only 2 acts in DAMSL – opening and closing). Such detailedness is very useful for study of human-human conversation even though it makes the annotation process more difficult. If we had only one goal – training a question-answering system – then we could to be satisfied with a more superficial typology of acts. But our primary goal is to study human-human conversation.

Our studies are currently centered on information seeking dialogues. We are using our system for annotating our corpus. Supposedly, the typology can be reduced in process of the work.

Dialogue Acts in Information Dialogues

For this paper, we annotated 10 spoken (telephone) and 10 WOZ dialogues from our corpus.

It is possible to outline the structure of information-seeking dialogue as consisting of four parts with different functions (Figure 1).

The four parts are
- a ritual beginning (greeting, introducing etc.);
- a ritual ending (thanking, farewell);
- requesting and giving information (answering questions, giving telephone numbers, etc.);
- solving communication problems (misunderstanding, inaudibility, unreliability of information) in cooperation of partners. This part often follows after the first question and forms an inserted sequence within the first adjacency pair, also it can be repeated within the following adjacency pairs.

Ritual parts can be missed in conversations. It is usual in WOZ dialogues that the user (A) does not greet the computer (B), he starts interaction with request. A's information request is expressed by directive's or question's pre-member (usually, open yes/no question, wh-question or wish). Pre-messages can be added to request (for example, 'I have a question'). B's answer is expressed as directive's or question's re-member, usually as open answer: giving information. B often asks adjustable questions to specify the conditions of answer.

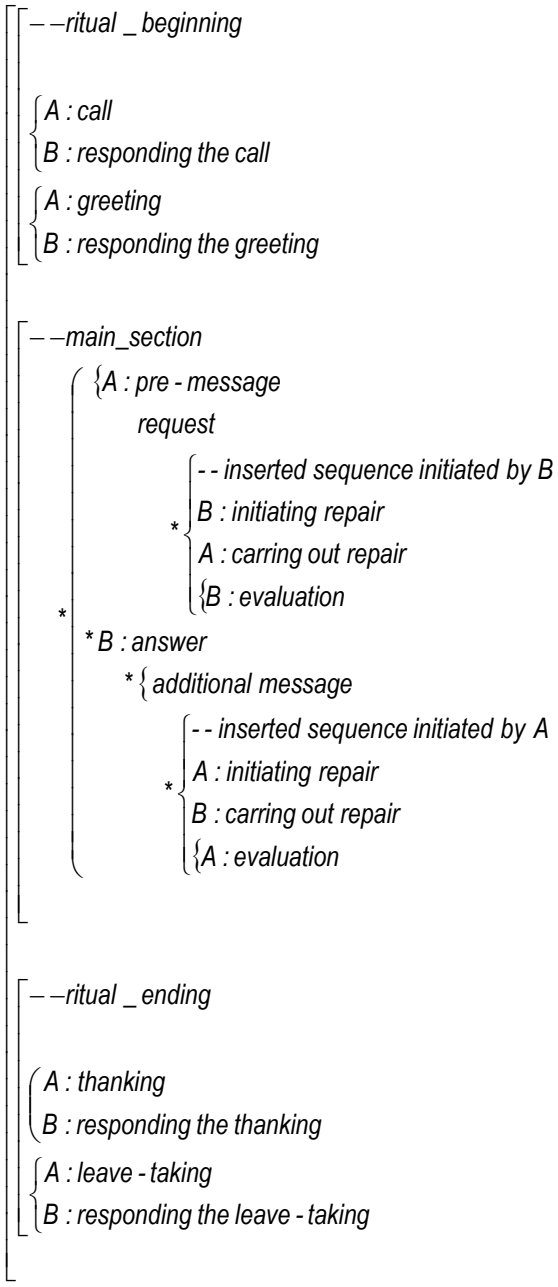


Figure 1. The structure of information dialogue.
Notations: (– adjacency pair, [– connects the whole dialogue or its section, { – dialogue act, adjacency pair or triad which is optional, * – dialogue act, adjacency pair or triad which can be repeated, -- – start of comment.

Let us consider two examples from our corpus (cf. Examples 1 and 2). The first dialogue is a telephone conversation and the second one is a WOZ dialogue. In the last case, the user put in his questions from the keyboard, and got answers from the wizard on the screen. The ritual beginning and ending parts are put out in the examples.

Example 1. A – client, B – a travel clerk. (Translated from Estonian.)

No	*Utterance*	*Dialogue act*
1	A: I'm interested in trips to Scandinavian states.	WISH
2	B: Yes?	YOU ARE WELCOME

3	More precisely?	SPECIFYING THE CONDITIONS OF ANSWER, POSTPONING THE ANSWER
4	A: Which variants do you have	WH QUESTION, POSTPONING THE ANSWER
5	to Sweden, Norway?	ADDITIONAL INFORMATION: SPECIFICATION
6	B: mmm… You can bay tickets by us.	ANSWER AS AN ALTERNATIVE
7	A: So.	ACKNOWLEDGEMENT, SIGNAL OF NEW INFORMATION
8	B: Plane and ship tickets.	ANSWER AS AN ALTERNATIVE, ADDITIONAL INFORMATION: SPECIFICATION
9	Unfortunately, we don't offer a whole travel packet.	OPEN ANSWER: ANOTHER
10	A: So.	ACKNOWLEDGEMENT, SIGNAL OF NEW INFORMATION
11	B: I mean a group trip.	ADDITIONAL INFORMATION: SPECIFICATION
12	A: mmm	CONTINUER

Various adjacency pairs appear in our dialogues: in the first example, A's wish (1st utterance) is closed by B's open answer (9), A's open question (4) is closed by B's answer as alternative (6), etc. In the second example, A's open question (1st utterance) is closed by B's open answer (8-11). Any adjacency pair can contain other pairs within. In the second example, the pair open question – open answer contains a pair adjustable question – open answer: giving information (utterances 4 and 5). An utterance can hold more than one function, as utterance 3 in the first example (postponing the answer that closes the utterance 1, and adjustable question that opens a new adjacency pair).

Some of dialogue acts do not occur in WOZ dialogues, for example, checking contact, continuer, taking up of attitudes. In our experiments, clients seldom asked questions which could be answered shortly, i.e. giving information about one or two trips. For that reason the computer asked very many adjustable questions followed by an open answer: giving information. In some degree, the questions were necessary for the wizard to extend the time for information retrieval.

Example 2. A – user, B – WOZ. Translated from Estonian.

No	*Utterance*	*Dialogue act*
1	A: When will the bus depart from N. to Tallinn,	OPEN QUESTION
2	in order that to arrive in Helsinki by ship at 12 o'clock?	ADDITIONAL INFORMATION
3	B: Wait please!	REQUEST TO WAIT
4	Are you interested in a certain week-day?	SPECIFYING THE CONDITONS OF ANSWER
5	A: I am interested in workdays	OPEN ANSWER: GIVING INFORMATION
6	B: Do you agree to spend a night in Tallinn?	SPECIFYING THE CONDITONS OF ANSWER
7	A: Yes.	AGREEMENT (YES)
8	B: The bus departs at 9.10 p.m. from X.	OPEN ANSWER: GIVING INFORMATION
9	Arrives at 10 p.m. in Tallinn.	OPEN ANSWER: GIVING INFORMATION
10	The ship departs at 8 a.m. from Tallinn, arrival at 10.30 a.m. in Helsinki,	OPEN ANSWER: GIVING INFORMATION
11	or departs at 9.30 a.m., arrives at 12.	OPEN ANSWER: GIVING INFORMATION
12	Do you need an earlier bus?	CLOSED YES/NO QUESTION
13	A: No.	NON-AGREEMENT
14	Thank you.	THANKING

The wizard asked such adjustable questions as 'Do you prefer a certain time-interval?', 'Do you prefer a certain weekday?', 'Are you interested in certain arrival time?' and so on.

The wizard often needed relatively long time to compile answers. In order that the user did not suspect of an error of the program or slow Internet-connection, the wizard used the dialogue act request to wait in the form 'Wait please!'.

Communicative Strategies in Information Dialogues

Let us depart from the notion of communicative strategy, considered in (Jokinen 1996a,b) as a part of the Constructive Dialogue Model (CDM). The departure point of the CDM is in general communicative principles which constrain cooperative and coherent communication. Dialogue participants are engaged in a cooperative task whereby a model of the joint purpose is constructed. Contributions are planned as reactions to the changing context. Communicative strategy is used by a participant to build up the next turn as a reaction to partner's previous one. Thus, communicative strategies express the coherence of the dialogue similarly as adjacency pairs of dialogue acts. Four context factors are used in CDM to determine communicative strategies:

1) expectations – is the turn expected or not;

2) the central conception – does the partner's turn keep the topic or not (related or unrelated);

3) goals – are the speaker's goals fulfilled or not;

4) initiatives – has the speaker initiative or not.

The first two parameters are hearer-related and the last two speaker-related.

All the context factors have binary values (1 or 0) in CDM which gives 2^4=16 communicative strategies. Every strategy can be represented by a vector of factors with coordinate values 1 or 0, for example, finish/start (vector 1111, i.e. expected-related-fulfilled-speaker), new request (0010, i.e. non-expected-unrelated-fulfilled-partner), subquestion (0101), follow-up old (1100), object (0001), etc.

By means of communicative strategies changing initiatives, achieving goals, changing topics, digressing from normal talk can be traced in dialogue structure.

$$
\left[\; * \left[\; \begin{array}{l} \text{A : finish/start} \\ * \left\{ \begin{array}{l} \text{- - inserted sequence initiated by B} \\ \text{B : subquestion} \quad | \ \text{object} \\ \text{A : follow - up old} \ | \ \text{continue} \quad | \text{backto} \end{array} \right. \\ \text{B : } \{ \text{notrelated} \\ \quad \text{follow - up old} \quad | \text{continue} \\ \quad \{ \text{somethingselse} \\ * \left\{ \begin{array}{l} \text{- - inserted sequence initiated by A} \\ \text{A : subquestion} \\ \text{B : follow - up old} \end{array} \right. \\ \{ \text{specify new} \ | \ \text{new dialogue} \end{array} \right. \\ \{ \text{A : continue} \end{array} \right.
$$

Figure 2. The structure of information dialogue: communicative strategies.
Notations: [– connects the whole dialogue or its section; { – an optional strategy or a sequence of strategies, * – strategy or a sequence of strategies which can be repeated, | – variants of strategies; -- – start of comment.

We annotated dialogue strategies in 10 spoken and 10 WOZ dialogues. The same information-seeking dialogues were analysed as for dialogue acts. The more frequent strategies were follow-up-old (represented by the vector 1100, i.e. expected-related-unfulfilled-partner), finish/start (1111), subquestion (0101) in spoken as well as in WOZ dialogues. Wizard often implemented the strategy unrelated (0000) ('Wait please!' in our example) which is unusual in spoken dialogues. From the other side, there are more changes of topic in telephone conversations as the WOZ dialogues. The user more strictly keeps the topic when interacting with the computer. Likewise, the initiative more often goes from one participant to the other in telephone conversations. Wizard tried to keep initiative and control interaction. The general structure of information dialogue is represented on Figure 2. The ritual beginning and ending parts are omitted.

Let us go back to the examples and use now communicative strategies for expressing the structure of dialogue (Examples 3-4).

Example 3 (cf. Example 1). A – client, B – a travel clerk.

No	Utterance	Vector of factors	Strategy
1	A: I'm interested in trips to Scandinavian states.	1111	finish/start
2	B: Yes?	1100	follow-up-old
3	More precisely?	0101	subquestion
4	A: Which variants do you have		
5	to Sweden, Norway?	1101	backto
6	B: mmm… You can by tickets by us.	0101	continue
7	A: So.	1100	follow-up-old
8	B: Plane and ship tickets.	1100	follow-up-old
9	Unfortunately, we don't sell a whole travel packet.	1100	follow-up-old
10	A: So.	1100	follow-up-old
11	B: I mean a group trip.	1100	follow-up-old
12	A: mmm	0001	object

Example 4 (cf. Example 2). A – user, B –WOZ.

No	Utterance	Vector of factors	Strategy
1	A: When will the bus depart from N. to Tallinn,		
2	in order that to arrive in Helsinki by ship at 12 o'clock?	1111	finish/start
3	B: Wait please!	0000	unrelated
4	Are you interested in a certain week-day?	0101	subquestion
5	A: I am interested in workdays	1100	follow-up-old
6	B: Do you agree to spend a night in Tallinn?	0101	subquestion
7	A: Yes.	1100	follow-up-old
8	B: The bus departs at 9.10 p.m. from X.		
9	Arrives at 10 p.m. in Tallinn.		
10	The ship departs at 8 a.m. from Tallinn, arrivals at 10.30 a.m. in Helsinki,		
11	or departs at 9.30 a.m., arrives at 12.	1100	follow-up-old
12	Do you need an earlier bus from N.?	0111	new dialogue
13	A: No.		
14	Thank you.	1110	follow-up-new

Discussion and Conclusion

When constructing the next utterance, a participant must act cooperatively and follow certain conversational norms. The reason is that dialogue can be considered as a negotiation process where each participant is responsible for continuation of communication. When we are speaking in terms of communicative acts, it means that there are certain acts that typically can follow an act, and if a speaker does not choose one act from this set then it can be treated as a violation of the norm. From the other side, when we are speaking in terms of communicative strategies then context factors determine the next strategy, and similarly, they guarantee the coherence of interaction.

A many-to-one mapping can be determined from the set **D** of dialogue acts to the set **S** of communicative strategies. The strategies where the speaker has initiative correspond to the first parts of adjacency pairs, and any act sets up a new goal. For example, wish and open question represent the finish/start strategy, specifying the conditions of answer – the subquestion strategy, opposing – the continue strategy, etc. Therefore, when interacting with a user, the dialogue system which uses information both of dialogue acts and communicative strategies, can more adequately respond to the user.

In our previous work, we have considered argumentation dialogues and determined communicative strategy as an algorithm for achieving a certain communicative goal (Koit, Oim 2000a,b). We also determined communicative tactics as algorithms for building the next utterances. Tactics of enticement, persuasion and threatening were considered. Thus our communicative tactics correspond to communicative strategies in (Jokinen 1996a,b). So far, we were interested in such conversations where participants could have antagonistic goals. The information-seeking communication, in opposite, is cooperative. Starting conversation, one of participants, A, has a communicative goal 'A get information P'. The communicative goal of the (cooperative) partner B is this same. This type of dialogues clearly will be the area where in the next few years already systems will be required that would be practically reliable, but at the same time could follow the rules of natural human communication.

Our further work will be concentrated on a formal model which integrates both a dialogue grammar and communicative strategies with our previous (a kind of BDI) model, and implementation of the model in information-seeking interactions.

Bibliography

[Allen et al., 1997] Allen, James, Mark Core. Draft of DAMSL: Dialog Act Markup in Several Layers; http://www.cs.rochester.edu/research/cisd/resources/damsl/RevisedManual/RevisedManual.html, 1997, used 10.02.2003

[Allwood et al., 2001] James Allwood, Ahlsen E., Björnberg M., Nivre J. *Social activity and communication act-related coding* In "Cothenburg Papers in Theoretical Linguistics 85. Dialog Coding – Function and Grammar. Göteborg Coding Schemas". Ed. Jens Allwood. Goteburg, 2001 , pp. 1–28.

[Dybkjær, 2000] Dybkjær, Laila. MATE Deliverable D6.2. Final Report; http://mate.nis.sdu.dk/about/deliverables.html, 2000, used 10.02. 2003.

[Francis, Hunston, 1992] Francis, Gill, Susan Hunston. Analysing everyday conversation. Advances in Spoken Discourse Analysis. Ed. By Malcolm Coulthard. Routledge, London and New York, 1992.

[Hakulinen, 1989] Auli Hakulinen *Keskustelun luonnehtimisesta konteksti- ja funktionaalisten tekijöiden nojalla* In Kieli, No 4. Suomalaisen keskustelun keinoja 1. Ed. by Auli Hakulinen. Helsinkin yliopiston suomen kielen laitos. Helsinki, 1989, pp. 41–72.

[Jokinen et al., 2001] Kristiina Jokinen, Hurtig T., Hynnä K., Kanto K., Kaipanen M., Kermanen A. *Self-organising dialogue management* In Proceedings of the NLPRS Workshop Neural Networks and Natural Language Processing, Tokyo, Japan, 2001.

[Jokinen, 1996a] Kristiina Jokinen *Cooperative Response Planning in CDM: Reasoning about Communicative Strategies.* In "TWLT11. Dialogue Management in Natural Language Systems", S. LuperFoy, A. Nijholt & G. Veldhuijzen van Zanten, ed. Enschede: Universiteit Twente, 1996, pp. 159-168.

[Jokinen, 1996b] Kristiina Jokinen Rational Agency. In "Rational Agency: Concepts, Theories, Models, and Applications", M. Fehling, ed.. Proc. of the AAAI Fall Symposium. MIT, Boston, 1996, pp. 89-93.

[Klein, Soria 1998] Klein, M., Soria, C. 1998. MATE Deliverable D1.1. Supported Coding Schemes. Dialogue Acts. – http://www.dfki.de/mate/d11/chap4.html, used 10.02.2003.

[Koit, 2001] Mare Koit *Annotating dialogue acts in Estonian Dialogue Corpus* In Proc. of the Euroconference Recent Advances in Natural Language Processing (RANLP'01). Ed. G. Angelova, K. Bontcheva, R. Mitkov, Nicolas Nicolov, Nikolai Nikolov. Tzigov Chark, Bulgaria, 2001, pp. 273-275.

[Koit, Oim, 2000a] Mare Koit and Haldur Õim *Dialogue management in the agreement negotiation process: a model that involves natural reasoning* In The 1st SIGdial Workshop on Discourse and Dialogue. Ed. L. Dybkjaer, K. Hasida, D. Traum. HongKong, Association for Computational Linguistics (ACL), 2000, pp. 102-111.

[Koit, Oim, 2000b] Mare Koit and Haldur Õim *Developing a model of natural dialogue.* In "From spoken dialogue to full natural interactive dialogue-theory, Empirical analysis and evaluation. LREC2000 Workshop proceedings", L. Dubkjær, ed. Athen, 2000, pp. 18-21.

[Sinclair, Coulthard, 1975] J.M. Sinclair, Coulthard, R.M. *Towards of Analysis of Discourse: The English used by Teachers and Pupils.* London: Oxford UP, 1975.

[Stolcke et al., 2000] Andreas Stolcke, Coccaro N., Bates R., Taylor P., Van Ess-Dykema C., Ries K., Shriberg E., Jurafsky D., Martin R., Meteer M. *Dialogue act modeling for automatic tagging and recognition of conversational speech* In Computational Linguistics, 26:3, 2000, pp. 339–373.

[Stenström, 1994] Anna-Brita Stenström *An Introduction to Spoken Interaction.* London and New York: Longman, 1994.

Author information

Mare Koit – University of Tartu, J. Liivi 2, 50409 Tartu, Estonia; e-mail: koit@ut.ee

О ЛЕКСИКО-СТАТИСТИЧЕСКОМ АНАЛИЗЕ НАУЧНО-ТЕХНИЧЕСКИХ ТЕКСТОВ

Н. М. Мищенко, Н.Н. Щеголева

Abstract: *The objectives and tools for information extraction from the special texts resulting in frequency lists of word stems, lexemes or combination of words used in these texts are presented. Such lists can help the user to elicit the terminology from the text, to determine the themes of texts or to select texts according to the field of knowledge defined by the terms given. The corresponding tools, named FEST, can process any professional text in any inflected language using Latin or Cyrillic alphabet. During performance FEST uses three dictionaries: of endings, of auxiliary words and of term stems. The last one can be empty or non-empty depending on the task raised to FEST. All dictionaries are generated by FEST on the base of formal specifications of endings or lexemes. Controlled by user FEST adds new stems to term dictionary recognizing automatically the grammatical categories of stems of unknown words with high frequency of their entries in the text. The strategy and the result of FEST application to the present paper in Russian are proposed.*

Keywords: *text analysis, text theme, endings dictionary, auxiliary words dictionary, dictionary of terms, list of homonyms, frequency list of word stems, frequency list of terms, frequency list of combination of words.*

Ключевые слова*: анализ текстов, словарь окончаний, словарь служебной лексики, словарь терминов, список омонимов, тематика текстов, частотный список основ словоформ, частотный список терминов, частотный список словосочетаний.*

Введение

Лексико-статистический анализ текстов выполняется во многих приложениях. В литературоведении частотные списки лексики используются для составления энциклопедических словарей писателей и поэтов. Сравнительный анализ частотных списков лексики художественных произведений разных авторов позволяет обнаружить особенности стилей и употребляемой лексики. Полученную информацию можно использовать для определения авторства текстов.

Наиболее интенсивное использование частотных списков наблюдается при обработке специальных текстов, которые являются самой распространенной формой представления информации в Интернете. Благодаря Интернету все народы мира превращаются в единое многоязычное информационное общество с высокоразвитым информационным обслуживанием [Markov, 2000], которое является основной характеристикой этого общества. Интернет — стимулятор исследований в области компьютерной лингвистики. Важными результатами развития инфраструктуры информационного обслуживания пользователей Интернета являются лингвистические системы перевода с одного языка на другой, поисковые системы, электронные словари различного типа и пр.

Выделяют три аспекта информационного обслуживания: собственно информация, инфраструктура и технические средства, из которых будем рассматривать первый аспект — информацию.

Согласно [InfoTerm, 2000] сегодня информация представляет собой четвертый фактор производства после труда, капитала и времени. Таким образом, информация становится важным “сырьем” почти во всех отраслях современной экономики. Чтобы использовать информацию оптимально, необходимо обеспечить *ясность* используемых в ней концепций и терминов. Это означает, что содержание терминов должно интерпретироваться однозначно и быть представлено адекватно на всех языках многоязычного информационного общества.

Терминология как совокупность терминов, представляющих ту или иную область знаний на уровне концепций, играет ключевую роль в формировании, обработке и передаче информации, а также в обучении.

Организация международного сотрудничества в области терминологий началась в Вене (Австрия), где была создан Технический комитет по терминологии при Международной Ассоциации стандартов в 1936 году. В настоящее время действует Международная Ассоциация по терминологии с центром в Вене. Цель

Ассоциации – поддержка и организация сотрудничества между членами Ассоциации, подготовка специальных публикаций, разработка и распространение программного обеспечения, участие в терминологических проектах. В 2000 году членом Ассоциации стала Украина.

С 1987 года регулярно проводятся международные конференции по компьютеризированной терминологиии и инженерии знаний. В Украине на протяжении последних двенадцати лет было проведено семь терминологических конференций.

Терминология может представлять информацию по-разному, мы же рассматриваем только ее текстовое представление. Главным направлением в развитии и использовании терминологии являются создание, сбор, унификация, стандартизация и трансляция терминов. В каждом из перечисленных направлений исследований главная роль принадлежит специалистам соответствующих областей знаний. При этом лексико-статистический анализ специальных текстов можно рассматривать как вспомогательный процесс в исследовании терминологии.

Цели и средства лексико-статистического анализа

Целью проводимого нами лексико-статистического анализа специальных текстов является поиск терминов, уточнение их значений и связей между ними для построения терминологических словарей и тезаурусов, определение частоты употребления в текстах терминов, находящихся в словаре, распознавание тематики текстов, а также использование терминологических словарей для поиска текстов, принадлежащих тематике, заданной словарем и пр.

В качестве наиболее доступного средства для предварительного исследования терминологии специальных текстов в настоящей работе рассматриваются частотные списки лексики текстов. Использование частотных списков для исследования терминологии научно-технических текстов базируется на высокой частоте употребления терминов (ключевых слов) в специальных текстах, выше которой является лишь частота употребления служебной лексики. Поэтому частотные списки, упорядоченные по убыванию частоты вхождений лексем в текст, позволяют человеку анализировать предельно сжатую информацию о терминах, используемых в тексте, в виде начальных фрагментов частотных списков вместо всего текста.

Формирование частотных списков осуществляет программная система FEST на основе результатов морфологического анализа научно-технических текстов. Система FEST применима к флективным языкам, таким как русский и украинский, а также к аналитическим, таким как болгарский и английский.

В процессе анализа специальных текстов системой FEST используются три словаря: словарь окончаний, словарь служебной лексики и словарь терминологической лексики. Все три словаря генерируются самой системой FEST по формальным описаниям (спецификациям) окончаний и лексики [Мищенко, 2000], которые выполняются человеком – носителем языка исследуемых текстов, с использованием учебников или грамматических словарей (таких как Грамматический словарь русского языка А.А. Зализняка).

Отсутствие словаря общеупотребительной лексики среди используемых системой FEST словарей позволяет сравнительно легко настраивать систему на новый язык специальных текстов.

Система FEST строит три типа частотных списков: частотный список основ не найденных в словаре словоформ, частотный список терминов и частотный список словосочетаний (элементарных синтаксических единиц), входящих в анализируемый текст.

В процессе функционирования системой FEST выполняются такие функции:

а) генерация словоформ по спецификации лексики с целью проверки правильности спецификаций;

б) генерация словаря окончаний по спецификации кортежей окончаний;

в) генерация словарей лексики по спецификации лексем;

г) морфологический анализ текстов;

д) распознавание словосочетаний в тексте;

е) формирование частотных списков;

ж) определение словарной формы словоформ, найденных в тексте и представленных в частотном списке их общей основой и последовательностью окончаний.

Процедура, выполняющая определение словарной формы словоформ, может быть применена к словоформам произвольного текста для построения словаря лексики текста.

Рассмотрим подробнее используемые системой FEST словари и формируемые ею частотные списки.

Описание словарей

Для формирования частотных списков система FEST использует разные конфигурации из трех словарей: словаря окончаний, словаря служебной лексики и словаря терминов.

Словарь окончаний используется при морфологическом анализе для формировании частотных списков всех типов. Он генерируется системой FEST для каждого языка один раз по списку именованных кортежей окончаний, составленных пользователем.

Каждый кортеж окончаний определяет класс лексем, принимающих окончания кортежа. Результат обработки последовательности кортежей окончаний – это список структур данных для морфологического анализа и список омонимов при каждом окончании. Строка списка омонимов некоторого окончания содержит имя кортежа, в который входят окончание и падеж(и) для принимающих это окончание именных частей речи или лицо для глаголов. Поскольку списки омонимов окончаний являются основными данными для алгоритма построения спецификации лексики, то целесообразно рассмотреть определение и представление омонимов более подробно.

Омонимия окончаний – это графическое совпадение окончаний, имеющих разные морфологические роли (значения). Рассмотрим кортеж окончаний, принимаемых лексемой русского языка *словарь*:

смь_1: -ь -я -ю -ь -ем -е -и -ей -ям -и -ями -ях;

Здесь смь_1 – мнемоническое имя кортежа, которое означает, что окончания кортежа, расположенные в порядке следования падежей, принимаются именами существительными мужского рода, словарная форма которых оканчивается на -ь, а число 1 используется для того, чтобы отличить данный кортеж от другого кортежа окончаний имен существительных с таким же окончанием (-ь) в словарной форме. В данном кортеже окончание -ь омонимично, оно имеет два значения: именительный и винительный падежи единственного числа. Окончание -и также имеет два значения: именительный и винительный падежи множественного числа.

Следовательно, в списке омонимов окончания -ь появится строка “смь_1, им. п. ед. ч., вин. п. мн. ч.”. Это же окончание встречается и в других кортежах, поэтому список омонимов будет пополнен новыми строками. И так для всех окончаний языка. Характерным для списков омонимов является то, что имена кортежей в одном списке омонимов не повторяются, но все окончания одного и того же кортежа в своих списках имеют имя этого кортежа. Список омонимов составляется и тогда, когда окончание не имеет омонимов. В этом случае список содержит одну строку с именем кортежа и одним значением окончания.

Словарь служебной лексики содержит наиболее часто встречающиеся части речи: союзы, предлоги, частицы, местоимения. Важной информацией в спецификации предлогов является указание на управление следующими за предлогом словами.

Использование словаря служебной лексики при морфологическом анализе обеспечивает расположение в начальной части частотных списков основ неизвестных значимых словоформ, среди которых могут быть термины.

Словари терминов, генерируются системой FEST по составленной человеком спецификации известных заранее терминов (например, взятых из бумажных словарей), а также динамически, то есть, по спецификации, составленной с участием системы FEST по частотному списку основ неизвестных словоформ – результату морфологического анализа входного текста без словаря терминов или при неполном таком словаре.

Построению частотного списка того или иного типа соответствует своя конфигурация используемых словарей. Тип частотного списка, в свою очередь, определяется конкретной задачей, которая решается с помощью списка. Мы рассматриваем такие задачи:

1) распознавание тематики текстов, например, для классификации массива текстов по тематике.

2) определение принадлежности текста тематике, заданной с помощью словаря терминов. Такая задача решается при выборке текстов определенной тематики из большого массива текстов разной тематики.
3) внесения частотных списков терминов и словосочетений в терминологические словари, где они характеризуют частоту употребления терминов, помещенных в словарь.

Описание частотных списков

Описание частотных списков сопровождается примерами списков, построенных на основе морфологического анализа данного текста доклада. Анализ знакомого текста позволяет сравнить наши знания об использованных терминах с полученными формально.

Частотный список основ словоформ, расположенных по убыванию частоты вхождений в текст, составляется с использованием только двух словарей: словаря окончаний и словаря служебной лексики. Список содержит не найденные в словаре основы, которые в дальнейшем будем называть неизвестными. Каждая основа в списке сопровождается последовательностью окончаний всех входящих в текст словоформ с данной основой. В качестве примера предлагаем первые пять строк (из 24-х, содержащих основы с высокой частотой употребления) построенного системой FEST частотного списка основ словоформ, в котором вручную будут исправлены ошибки, обусловленные отсутствием основ значимых лексем во время морфологического анализа.

1) 96 3.37% 5 спис (-ов, -и, -е, -0, -а, -у, -ах, -ом, -ам, -ами);
2) 68 2.49% 1 текст (-ов, -ах, -ам, -а, -0, -е, -ом, -ы);
3) 67 2.45% 9 словар (-я, -ей, -и, -е, -ем, -ь, -ями, -ю, -ях);
4) 61 2.16% 5 частотн (-ых, -ые, -ый, -ом, -ого, -ому, -ым, -ыми);
5) 49 1.74% 9 термин (-ов, -ах, -ы, -ами).

В каждой строке частотного списка первое число – это порядковый номер строки, второе – число вхождений словоформ с соответствующей основой в текст, третье – процент вхождений по отношению ко всем словоформам текста, четвертое – строка текста, где впервые встретилась словоформа с данной основой и первым окончанием в скобках.

Заметим, что в частотном списке основ вместе с терминами может оказаться служебная или общеупотребительная лексика, что вполне естественно, если она входит с терминами в словосочетания, являющиеся ключевыми для данного текста. Так, судя по приведенному выше частотному списку, общеупотребительнаяе лексема *список* в данной работе является словом-термином.

Частотный список терминов, также расположенных по убыванию частоты, содержит известные, то есть, найденные в словаре лексемы. В частотном списке они представлены словарной формой: имена существительные в именительном падеже единственного числа, имена прилагательные в том же падеже и числе мужского рода и т.д. В каждой лексеме списка аккумулированы все используемые в тексте ее словоформы, число вхождений которых приводится в строке частотного списка с данной лексемой. Приводим пример фрагмента частотного списка терминов данной статьи, построенного с использованием словаря терминов, который был сгенерирован по представленному выше частотному списку основ неизвестных словоформ.

1) 96 3.42% 5 список
2) 68 2.44% 1 текст
3) 67 2.44% 9 словарь
4) 61 2.16% 5 частотный
5) 49 1.76% 9 термин

Числа в строках означают то же самое, что и в частотном списке основ неизвестных словоформ.

Частотный список словосочетаний, расположенный по убыванию длин словосочетаний, а в пределах одинаковой длины – по убыванию частоты вхождений в текст. По предположению словосочетания из наиболее часто встречающихся в специальных текстах словоформ являются ключевыми словами (терминами). Окончательное решение о признании словосочетаний терминами принадлежит человеку.

Заметим, что при построении частотных списков словосочетаний словарь служебной лексики для флективных языков (таких как русский, украинский) может не использоваться, так как с участием служебных слов, в основном местоимений, порождается очень много словосочетаний. В то время как для аналитических языков, типа английского, построение частотного списка словосочетаний желательно выполнять с участием словаря служебных слов.

Часть полученных словосочетаний из словаря терминов данной статьи приведены в начале статьи как ключевые слова.

Определение тематики текста

Определение тематики специального текста выполняется системой FEST с участием человека за несколько шагов. Рассмотрим действия этой системы и человека на каждом шаге (результаты действий системы поданы в качестве примеров в предыдущем разделе).

Шаг 1. **Формирование частотного списка основ словоформ**, не найденных в словаре во время морфологического анализа, который выполнялся только со словарем окончаний и словарем служебных слов. В начальном фрагменте частотного списка основ словоформ должны находиться основы терминов, так как значимые слова отсутствуют в словаре служебных слов. Условимся в дальнейшем использовать словосочетание “в частотном списке” вместо “в начальном фрагменте частотного списка”, поскольку в поисках терминов будем анализировать содержимое только начальных фрагментов.

Полученный частотный список основ анализируется человеком. В результате анализа может быть принято одно из нескольких альтернативных решений:

а) частотный список позволяет сделать вывод о тематике текста в той мере, которая удовлетворяет человека. На этом работа над текстом прекращается. Возможно построение словаря обнаруженных терминов (шаг 2);

б) частотный список не содержит достаточно информации для понимания тематики текста. В этом случае человек может придти к выводу о слабо выраженной специализации текста и прекратить работу;

в) возможен, однако, вариант, когда человек захочет получить более точную информацию о тексте. В этом случае он может заказать построение частотного списка словосочетаний с участием слов из частотного списка. Для этого следует прежде всего сформировать формальное описание (спецификацию) лексем, основы которых находятся в частотном списке основ, и которые по предположению являются основами терминов. Для того чтобы процесс формирования спецификации лексики был более эффективным, человек должен предварительно исправить ошибки в частотном списке, связанные, прежде всего, с ложной омонимией окончаний. Он также должен решить, сколько основ следует внести в словарь.

Шаг 2. **Формирование спецификаций основ** с участием системы FEST. Объязательным элементом спецификации каждой лексемы является имя кортежа окончаний, которые принимают словоформы, производные от лексемы. Это имя определяется системой FEST с помощью окончаний при основе в частотном списке и списков омонимов каждого окончания в словаре окончаний.

Из того, что окончания при неизвестной основе принадлежат одному кортежу, следует, что списки омонимов всех этих окончаний должны содержать строку с именем этого кортежа. Отыскать общее имя кортежа для всех окончаний одной и той же основы не представляет труда, если словоформы с этой основой встречаются настолько часто, чтобы в частотном списке появились все окончания соответствующего кортежа. Из предположения, что для словоформ, являющихся терминами, это требование выполняется, следует применимость предлагаемого способа определения имени кортежа.

Если окончаний для однозначного определения имени кортежа недостаточно, то система находит несколько общих имен кортежей и строит несколько спецификаций, а человек удаляет неправильные. Помощь человека нужна и в том случае, когда в процессе словоизменения в основе происходит чередование согласных, выпадание гласных и т.п.

Итак, неизвестные основы из частотного списка основ словоформ, которые приведены в предыдущем разделе в виде примера, будут специфицированы следующим образом (спецификации сопровождаются комментариями):

опис => * : см0_1 "ок" (еи, евн) "к" ; /* см0_1 – имя кортежа окончаний, которое означает: основа принадлежит имени существительному мужского рода с нулевым окончанием. Лексема принимает суффикс "ок" в именительном и винительном падежах единственного числа (для предметов), в остальных падежах – суффикс "к" */

текст => * : см0_3; /* имя кортежа означает то же, что и в предыдущей спецификации, а индекс 3 указывает на другой кортеж окончаний */

словар => * : смь_3; /* имя кортежа означает: имя существительное мужского рода с окончанием –ь.*/

частотн => * : пмый/пжая/псое; /*здесь указаны имена трех кортежей имен прилагательных, соответственно, мужского рода на -ый, женского рода на –ая и среднего рода на -ое */

окончани => * : ссе; /* имя кортежа означает: имя существительное среднего рода на -е*/.

Шаг 3. **Генерация словоформ** по сформированной спецификации. В качестве примера подаем последовательность словоформ, сгенерированных системой FEST по спецификации основы слова *список* с указанием падежа и числа каждой словоформы: список (им.ед., вн.ед.), списка (рд.ед.), списку (дт.ед.), списком (тв.ед.), списке (пр.ед.), списки (им.мн., вн.мн.), списков (рд.мн.), спискам (дт.мн.), списками (тв.мн.), списках (пр.мн.).

Человек проверяет словоформы и удаляет неправильные спецификации, если такие есть.

Шаг 4. **Генерация словаря терминов** по построенной спецификации неизвестных основ.

Шаг 5. Генерация частотного списка словосочетаний из словоформ, основы которых находятся в заданном словаре терминов. Словарь служебных слов флективного языка не используется. Человек анализирует полученный частотный список словосочетаний и принимает окончательное решение о тематике текста.

В заключение раздела приведем некоторые статистические данные о настоящем тексте, полученные с помощью системы FEST: в тексте имеется 2800 словоформ, из которых служебная лексика составляет 30% словоформ (116 лексем), специальная 27% словоформ (24 лексемы). Остальные 43% словоформ составляют общеупотребительную лексику (603 лексемы) .

Определение принадлежности текста заданной тематике

Построения на шаге 5 предыдущего раздела по сути могут дать ответ на вопрос, принадлежит ли данный текст области знаний, заданной с помощью словаря терминов. Другая возможность – построение частотного списка терминов, когда система FEST выполняет морфологический анализ только со словарем заданных терминов.

Для окончательного ответа на заданный вопрос необходимо сравнение частотных списков терминов, построенных по заданному словарю терминов, с частотным списком основ незвестных словоформ этого же текста, построенных без словаря терминов. Если частота основ неизвестных словоформ ниже частоты заданных терминов, то текст можно отнести к искомым, в противном случае – нет.

Для того, чтобы снабдить словарь терминов частотными списками вхождений этих терминов или словосочетаний в специальные тексты, необходимо построить такие частотные списки на большом текстовом материале из данной области знаний с использованием данного словаря терминов.

Выводы

Предложена программная система FEST для лексико-статистического анализа специальных текстов на флективных или аналитических языках как простое и эффективное средства для первичной обработки специальных текстов с целью определения их тематики, поиска терминов и создания словарей терминов. Отсутствие необходимости в словарях общеупотребительной лексики способствует быстрой адаптации системы FEST, которую следует рассматривать как прототип для создания текстовых процессоров, учитывающих требования конкретных пользователей. На основе системы создана производственная программная система для фирмы, занимающейся классификацией текстов по юриспруденции на русском и украинском языках.

Библиография

[Markov, 2000] Kr. Markov, Kr. Ivanova, I. Mitov. The Information Society // Abstracts of the Fifth International Conference ITA 2000 (Sept., 1-15, 2000, Varna, Bulgaria), FOI-COMMERCE, Sofia, 2000, p.10.

[InfoTerm, 2000] www.infoterm.or.at/terminology.html

[Мищенко 2000] Н.М. Мищенко. О генерации языковых процессоров на основе формальной спецификации лексики обрабатываемых текстов /Труды Межд. Сем. "ДИАЛОГ'2000. Компьютерная лингвистика и ее приложения" (31 мая – 5 июня 2000, РФ, Протвино) в 2-х т. Т.2. – Прикладные проблемы. – Протвино. – 2000. – С. 271-278.

Информация об авторах

Надежда Михайловна Мищенко – Киев, Украина, e-mail: nady@dolphin.icyb.kiev.ua

Наталья Николаевна Щеголева – Институт кибернетики имени В.М. Глушкова Национальной Академии Наук Украины, Киев, 03187, просп. Академика Глушкова, 40, Украина, e-mail: nat@d105.icyb.kiev.ua

FEATURE EXTRACTION FOR CLASSIFICATION IN THE DATA MINING PROCESS

M. Pechenizkiy, S. Puuronen, A. Tsymbal

***Abstract**: Dimensionality reduction is a very important step in the data mining process. In this paper, we consider feature extraction for classification tasks as a technique to overcome problems occurring because of "the curse of dimensionality". Three different eigenvector-based feature extraction approaches are discussed and three different kinds of applications with respect to classification tasks are considered. The summary of obtained results concerning the accuracy of classification schemes is presented with the conclusion about the search for the most appropriate feature extraction method. The problem how to discover knowledge needed to integrate the feature extraction and classification processes is stated. A decision support system to aid in the integration of the feature extraction and classification processes is proposed. The goals and requirements set for the decision support system and its basic structure are defined. The means of knowledge acquisition needed to build up the proposed system are considered.*

***Keywords**: Feature Extraction, Classification, Data Mining.*

Introduction

Data mining applies data analysis and discovery algorithms to perform automatic extraction of information from vast amounts of data. This process bridges many technical areas, including databases, human-computer interaction, statistical analysis, and machine learning.

A typical data-mining task is to predict an unknown value of some attribute of a new instance when the values of the other attributes of the new instance are known and a collection of instances with known values of all the attributes is given. In many applications, data, which is the subject of analysis and processing in data mining, is multidimensional, and presented by a number of features. The so-called "curse of dimensionality" pertinent to many learning algorithms, denotes the drastic raise of computational complexity and classification error with data having high amount of dimensions [Bellman, 1961]. Hence, the dimensionality of the feature space is often reduced before classification is undertaken.

Feature extraction (FE) is one of the dimensionality reduction techniques. FE extracts a subset of new features from the original feature set by means of some functional mapping keeping as much information in the data as possible [Fukunaga, 1990]. Conventional Principal Component Analysis (PCA) is one of the most commonly used

feature extraction techniques. PCA extracts the axes on which the data shows the highest variability [Jolliffe, 1986]. There exist many variations of the PCA that use local and/or non-linear processing to improve dimensionality reduction [Oza, 1999], though they generally do not use class information.

In our research, beside the PCA, we discuss also two eigenvector-based approaches that use the within- and between-class covariance matrices and thus do take into account the class information. We analyse them with respect to the general task of classification, to the learning algorithm being used and to dynamic integration of classifiers (DIC).

During the last years data mining has evolved from less sophisticated first-generation techniques to today's cutting-edge ones. Currently there is a growing need for next-generation data mining systems to manage knowledge discovery applications [Fayyad, 1996a]. These systems should be able to discover knowledge by combining several available data exploration techniques, and provide a fully automatic environment, or an application envelope, surrounding this highly sophisticated data mining engine [Fayyad, 1996b].

In this paper we consider a decision support system (DSS) approach that is based on the methodology used in expert systems (ES). The approach combines feature extraction techniques with different classification schemes. The main goal of such a system is to automate as far as possible the selection of the most suitable feature extraction approach for a certain classification task on a given data set according to a set of defined criteria.

In the next sections we consider the feature extraction process for classification and present the summary of achieved results. Then we consider a decision support system that integrates the feature extraction and classification processes, describing its goals, requirements, structure and the ways of knowledge acquisition. As a summary the obtained results are discussed and the focus of the further research is described.

PCA-based Feature Extraction

Generally, feature extraction for classification can be seen as a search among all possible transformations of the feature set for the best one, which preserves class separability as much as possible in the space with the lowest possible dimensionality [Fukunaga, 1990]. In other words we are interested in finding a projection **w**:

$$\mathbf{y} = \mathbf{w}^T \mathbf{x} \qquad (1)$$

where **y** is a $p' \times 1$ transformed data point (presented using p' features), **w** is a $p \times p'$ transformation matrix, and **x** is a $p \times 1$ original data point (presented using p features).

In [Oza, 1999] it was shown that the conventional PCA transforms the original set of features into a smaller subset of linear combinations that account for the most of the variance of the original data set. Although it is the most popular feature extraction technique, it has a serious drawback, namely the conventional PCA gives high weights to features with higher variabilities irrespective of whether they are useful for classification or not. This may give rise to the situation where the chosen principal component corresponds to the attribute with the highest variability but having no discriminating power.

A usual approach to overcome the above problem is to use some class separability criterion [Aivazyan, 1989], e.g. the criteria defined in Fisher linear discriminant analysis and based on the family of functions of scatter matrices:

$$J(\mathbf{w}) = \frac{\mathbf{w}^T \mathbf{S}_B \mathbf{w}}{\mathbf{w}^T \mathbf{S}_W \mathbf{w}} \qquad (2)$$

where $\mathbf{S}_B$ is the between-class covariance matrix that shows the scatter of the expected vectors around the mixture mean, and $\mathbf{S}_W$ is the within-class covariance, that shows the scatter of samples around their respective class expected vectors.

A number of other criteria were proposed in [Fukunaga, 1990]. Both parametric and nonparametric approaches optimize the criterion (2) by using the *simultaneous diagonalization algorithm* [Fukunaga, 1990].

In [Tsymbal, 2002] we analyzed the task of eigenvector-based feature extraction for classification in general; a 3NN classifier was used as an example. The experiments were conducted on 21 data sets from the UCI machine learning repository. The experimental results supported our expectations. Classification without feature extraction produced clearly the worst results. This shows the so-called “curse of dimensionality” with the experimented data sets and the necessity to apply feature extraction with them. The conventional PCA was the worst feature

extraction technique on average. The nonparametric technique was only slightly better than the parametric one on average. However, this can be explained by the selection of the data sets, which are relatively easy to learn and do not include significant nonnormal class distributions. Besides, better parameter tuning can be used to achieve better results with the nonparametric technique. The nonparametric technique performed much better on categorical data for this selection of the data sets.

Still, it is necessary to note that each feature extraction technique was significantly worse than all the other techniques at least on a single data set. Thus it was shown that among the tested ones there does not exist any "the overall best" feature extraction method for classification with regard to all given data sets, and the problem of selection of the best suited feature extraction algorithm with its optimal parameters for classification was stated.

Feature Extraction for a Classifier and Dynamic Integration of Classifiers

The other interesting research question is to look for the best combination of a feature extraction method and a classifier among the available methods for a data set. We considered three PCA-based feature extraction methods with a number of different classifiers. A series of experiments were conducted on the same 21 data sets from the UCI machine learning repository. The results showed that there does not exist "feature extractor – classifier" pair that would be the best one for any given data set.

The other problem of search for the best suited feature extraction algorithm and its parameters for a certain classifier with regard to the given data set was stated.

Recent research has proved the benefits of the use of ensembles of classifiers for classification problems [Merz 1996]. The challenge of integration is to decide which classifier to select or how to combine classifications produced by several classifiers.

The integration of an ensemble of classifiers has been shown to yield higher accuracy than the most accurate base classifier alone in different real-world tasks. The two main approaches to the integration are: first, the *combination approach,* where the base classifiers produce their classifications and the final result is composed using those classifications and second, the *selection approach,* where one of the classifiers is selected and the final result is the result produced by it.

We consider the use of feature extraction for coping with the curse of dimensionality in the dynamic integration of classifiers [Tsymbal 2003]. The FEDIC (Feature Extraction for Dynamic Integration of Classifiers) algorithm was proposed which combines the dynamic selection and dynamic voting classifier integration techniques with the conventional PCA and two supervised eigenvector-based approaches that use the within- and between-class covariance matrices.

Our main hypothesis has been that with the data sets for which feature extraction improves classification accuracy employing a base classifier (such as *kNN* or Naïve Bayes), it will also improve classification accuracy when employing a dynamic integration approach. Conversely, we expected that with data sets for which feature extraction decreases or has no effect on classification accuracy with the base classifier, it will also decrease or will have no effect on classification accuracy employing a dynamic integration approach. This hypothesis was supported by the results obtained during the experiments conducted on a number of data sets from the UCI machine learning repository.

Decision Support System for the Best-suited Technique and Its Parameters Selection

Summarising the results of the up to this date research in the area we can state that there is no feature extraction technique that would be the best for any data set given with respect to the task of classification. Thus the problem of adaptive selection of the most suitable feature extraction technique for a data set needs further research work. We do not have canonical knowledge, perfect mathematical models or any relevant tool to select the best-suited technique. Thus, we are dealing with so-called empirical domain area having a volume of accumulated empirical facts, some trends and some dependencies found. And the theoretical summarization of these facts, trends and dependencies is the question of future research.

These prerequisites lead us on to consider the possibility of decision support system developing based on the methodology of expert system design in order to help to manage the data mining process with regard to the selection of the best-suited combination for a classification task. The main goal of such a system is to recommend

the best-suited feature extraction method and a classifier for a given data set according to a set of rules related to a given problem. Achieving this goal produces a great benefit in the sense that it would be possible to come from the *wrapper* type approach to the *filter* paradigm. In the wrapper type approach the interaction between the feature selection process and the construction of the classification model is assumed and the parameter tuning for every stage and for every method is needed. In the filter paradigm evaluation process is independent from the learning algorithm and the methods, and their parameters' selection process is performed according to a certain set of criteria before the algorithm starts. However, an additional goal of the prediction of model's output performance needs also further consideration.

The coverage of the responsibilities of the decision support system in the data mining process is depicted on fig. 1 (left). It can be seen that as soon as a training data set comes to the data mining system and the preliminary data preparation and data cleaning processes are finished, the Decision Support System takes responsibility to manage the processes of feature extraction and classification, namely to select the best-suited methods and the best parameters for those methods. And only after the model is built and validated, it comes to final evaluation on a test set.

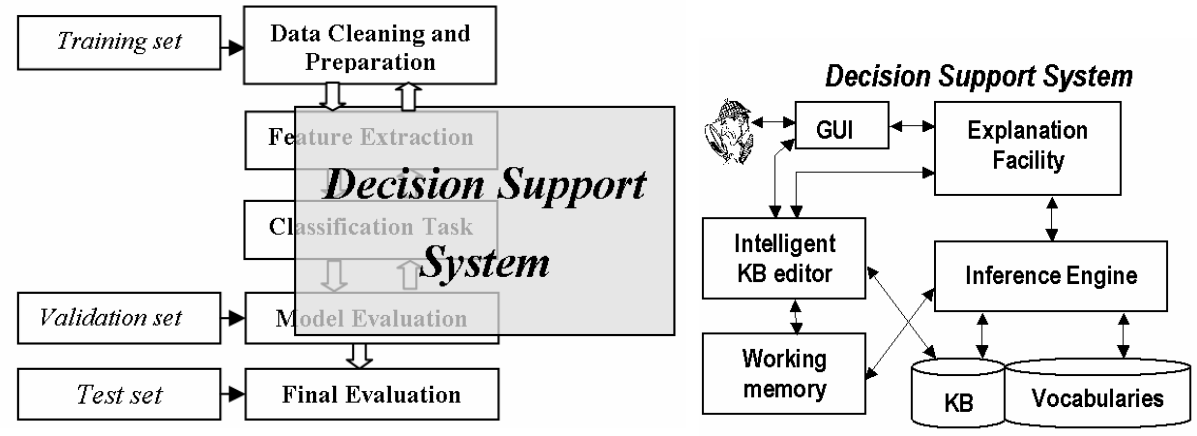


Figure 1 – Integration of the decision support system (right) into the data mining process (left).

The basic structure of the DSS is presented on fig.1 (right). The "heart" of this system is the *Knowledge Base* (KB) that contains a set of facts about the domain area and a set of rules in a symbolic form describing the logical references between the "symptoms" that are a concrete classification problem and recommendations about the best-suited model for a given problem. These facts and rules can be a basis for the new ones. Generally, the knowledge base is a dynamic part of the system that can be supplemented and updated through the knowledge acquisition and knowledge refinement processes. The content of the knowledge base is updated when time elapse, and, in some cases, even during solving the task.

Vocabularies contain the lists of terms that include feature extraction methods and their input parameters, classifiers and their input and output parameters, and three types of data set characteristics: simple measures such as the number of instances, the number of attributes, and the number of classes; statistical measures such as the departure from normality, correlation within attributes, the proportion of total variation explained by the first *k* canonical discriminants; and information-theoretic measures such as noisiness of attributes, the number of irrelevant attributes, and the mutual information of class and attribute.

Inference Engine is the "brain" of the system. It can be considered as a rule interpreter. It is a logical programming component of the system that realises a reasoning mechanism based on the Knowledge Base and the Working Memory information. This subsystem is the backbone of the consultation process since it produces the information how the system comes to a conclusion. The inference engine is able to search for the missing knowledge either by asking from a researcher or by conducting additional experiments if there is not enough information to come to a confident conclusion. This subsystem should contain at least three main components: an interpreter that executes some agenda: a set of rules from the Knowledge Base; a scheduler that controls the

execution of the agenda; a consistency enforcer that maintains the reasoning process used to obtaine the decisions.

The Working Memory contains all the data that is essential for the current problem including input data, results of inference, and intermediate results.

The Intelligent KB Editor is a tool that aims to provide intelligent methodologies for refinement of knowledge accumulated in the knowledge base and insertion of new knowledge got from the knowledge acquisition process. Particularly, the Editor should include the patterns of knowledge representation language and provide a tool for a knowledge engineer to develop the knowledge base in a dialogue mode. Beside these common functions related to the Editor, its tasks include experimentation routing management that will be considered in the next section.

The Explanation Facility is the subsystem that is aimed to trace problem solving steps to the obtaining results. Intermediate conclusions and the consequence of applied rules are stored in the tree of conclusion. Successful application of a rule corresponds to moving to another node of the tree during reaching a goal statement. The explanation system should be able to answer on the questions as: how was a certain conclusion inferred? why was additional information requested? how was the output model's performance estimated?

The Graphical User Interface (GUI) provides an interactive environment that links the KB, Intelligent KB Editor and knowledge engineer as well as the Explanation Facility module with the users of the system.

Filling the knowledge base is among the most challenging task related to the development of the DSS and it will be in the heart of our research focus. Potential contribution might be found discovering a number of criteria from the experiments conducted on artificially generated data sets with pre-defined characteristics examining the dependencies between the characteristics of a data set in general and the characteristics of every local partition of the instance space in particular, and the type and parameters of the feature extraction approach best suited for the data set will help to define a set of criteria that can be applied for the generation of rules needed for the decision-making system.

The Knowledge Acquisition Process

The Knowledge Base is a dynamic part of the system that can be supplemented and refreshed through The Intelligent KB Editor. We should notice that there are two potential sources of knowledge to be discovered for the proposed system. These are the analysis of theory background that lies behind the feature extraction and classification methods, and field experiments.

In the first case, knowledge is formulated by an expert in the area of the specific feature extraction methods and classification schemes, and then represented as a set of rules by a knowledge engineer in the terms of a knowledge representation language that is supported by the system. We argue that it is possible and reasonable to categorise the facts and rules that are present in the Knowledge Base. Categorisation can be done according to the way the knowledge has been obtained – has it been got from the analysis of experimental results of from the domain theory, was it put automatically by the Intelligent KB Editor or by a knowledge engineer (who could be a data miner as well). Another categorisation criterion is the level of confidence of a rule. The expert can be sure in a certain fact or may just think or to hypothesize about another fact. In a similar way, a rule that has been just generated from the analysis of results by experimenting on artificially generated data sets but has been never verified on real-worlds data sets and a rule that has been verified on a number of real-world problems. These two rules definitely should not have the same level of confidence.

In addition to the "trust" criteria due to the categorisation of the rules it is possible to adapt the system to a concrete researcher needs and preferences by giving higher weights to the rules that actually are the ones of the user.

And, in the second case, a data miner can discover knowledge during the analysis of results obtained from the experiments as separate facts, trends and dependencies. In the same manner, discovered knowledge is represented as a set of rules by a knowledge engineer using of the knowledge representation language. Alternatively, the knowledge acquisition process can be automatic, i.e. the knowledge discovery process would be accomplished without any interference with a human expert. This may happen using the possibility of deriving new rules and updating the old ones based on the analysis of results obtained during the self-run experimenting.

In both the last cases we have a problem of learning how the Intelligent KB Editor should try to build up a classification or a regression model on meta-data resulted from experiments. In this context the input parameters for a classification model are specific data set characteristics and a classification model's outputs that include accuracy, sensitivity, specificity, time complexity, etc. The combination of a feature extraction method's and a classification model's names with their parameter values represents a class label. When building a regression model – meta-data-set attributes are data set characteristics, the feature extraction method's and the classification model's names, and one of the model output characteristics is the attribute which value (continuous) has to be predicted.

Then, in terms of attribute-value (feature-value) notation, each instance can be represented in the following way:

$$\mathbf{x} = [v(x_{DS_1}), \ldots, v(x_{DS_l}), v(x_{MO_1}), \ldots, v(x_{MOm_1})],$$

where $v(x_{DSi})$ denotes the value of attribute x_{DSi} that represents one of the data set characteristics, and $v(x_{MOi})$ denotes the value of attribute x_{MOi} that represents one of the model output characteristics, and $l + m = p$ is the number of attributes that constitute the meta-data-set.

And placing an instance into one of a finite set of possible categories can be depicted as

$$C(\mathbf{x}) \in range(\mathbf{y}),$$

where $range(\mathbf{y})$ denotes the set of possible values for the categorical output attribute, class value *y*. In our case class value is assigned to every distinct combination of a feature extraction method and a classifier with their parameters values.

The results obtained by us up to the present stage of research show a high level of complexity in dependencies between the data set characteristics and the best-suited scheme for the data mining process. In order to further develop our understanding it is necessary to proceed the research with the following iterations:

- Generation of artificial data sets with known characteristics (simple, statistical and information-theoretic measures);
- Design of experiments on the generated artificial data sets;
- Derivation of dependencies and definition of the criteria from the obtained results;
- Development of a knowledge base defining a set of rules on the set of obtained criteria;
- Proof of the constructed theory with a set of experiments on real-world data sets.

Thus, three basic research methods are used in the research: the theoretical approach, the constructive approach, and the experimental approach. These approaches are closely related and are applied in parallel. The theoretical backgrounds are exploited during the constructive work and the constructions are used for experimentation. The results of constructive and experimental work are used to refine the theory.

An example of such a procedure can be presented as:

- Generation of artificial data sets with the number of attributes from 2 to 100, with the number of instances from 150 to 5000, with the number of classes from 2 to 10, with the average correlation between the attributes from 10% to 90%, with the average noisiness of attributes from 10% to 50%, with the percent of irrelevant attributes from the total number of attributes from 10% to 50%.
- Design of the experiments on generated artificial data sets and analysing accuracy and efficiency of the classification model built on different learning algorithms and using different feature extraction methods. Tuning of the input parameters for each combination is required.
- Analysis of the dependencies and trends between output accuracies and efficiencies, feature extraction methods and classifiers, their input parameters, and pre-defined data set characteristics.
- Definition of a set of rules that reflect found dependencies and trends.
- Execution of a number of experiments on UCI data sets using DSS for the best-suited feature extraction method and classifier selection.
- Addition of the invented rules that were successfully validated during the tests on the benchmark data sets to the knowledge base.

Conducting a number of experiments on artificial data sets with pre-defined characteristics, according to the example shown above, we will get an input space **x**, i.e. as the one presented in Table 1.

Table 1 – An example of the hypothetic meta-data-set.

Data set characteristics						Model output characteristics				ModelID (class label)
Simple		Statistical		Inf. Theoretic		Accuracy		Complexity		
attributes	classes	corr.	normality	noise	entropy	accuracy	diversity	training time	test time	
$v(x_{DS_1}^{(1)})$			…		$v(x_{DS_i}^{(1)})$	$v(x_{MO_1}^{(1)})$	…		$v(x_{MOm}^{(1)})$	C(**x**1)
…					…	…			…	…
$v(x_{DS_1}^{(n)})$			…		$v(x_{DS_i}^{(n)})$	$v(x_{MO_1}^{(n)})$	…		$v(x_{MO_{m1}}^{(n)})$	C(**x**n)

We consider a decision tree learning algorithm as a mean of automatic rule extraction for the knowledge base. Decision tree learning is one of the most widely used inductive learning methods [Quinlan, 1993]. A decision tree is represented as a set of nodes and arcs. Each node contains a feature (an attribute) and each arc leaving the node is labelled with a particular value (or range of values) for that feature. Together, a node and the arcs leaving it represent a decision about the path an example follows when being classified by the tree. Given a set of training examples, a decision tree is induced in a "top-down" fashion by repeatedly dividing up the examples according to their values for a particular feature. This is known as a "divide and conquer" or "recursive partitioning" approach to learning. Initially all the examples are in one partition and each feature is evaluated for its ability to improve the "purity" of the classes in the partitions it produces. The splitting process continues recursively until all of the leaf nodes are of one class.

At the Figure 2 an example of the part of the abstract model built by decision tree on the meta-training set is presented. By means of analysing the tree branches it is possible to generate "if-then" rules for the knowledge base. A rule reflects certain relationship between meta-data-set characteristics and a combination of a feature extraction method and a classification model.

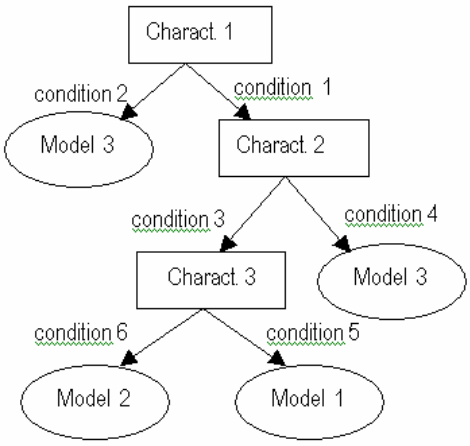


Figure 2 – The example of the part of abstract model built by decision tree.

Conclusion

Feature extraction is one of the dimensionality reduction techniques that are often used to struggle against the problems caused by the "curse of dimensionality". In this paper we considered three eigenvector-based feature extraction approaches that were applied for different classification problems. We presented the summary of results that shows a high level of complexity in dependencies between the data set characteristics and the best-suited scheme for the data mining process. There is no feature extraction method that would be the most suitable for all classification tasks. Due to the fact that there is no well-grounded strong theory that would help to build up an automated system for such feature extraction method selection, a decision support system that would accumulate separate facts, trends and dependencies between the data characteristics and output parameters of classification schemes performed in the spaces of extracted features was proposed.

We considered the goals of such a system, the basic ideas that define its structure and methodology of knowledge acquisition and validation. The Knowledge Base is the basis for the intellectuality of the expert system. That is why we recognised the problem of discovering rules from the experiments of an artificiality generated data set with known predefined simple, statistical and information-theoretic measures, and validation of those rules on benchmark data sets as a prior research focus in this area.

It should be noticed that generally the proposed approach has a serious limitation. Namely the drawbacks can be expressed in the terms of fragmentariness and incoherence (disconnectedness) of the components of knowledge to be produced. And we definitely do not claim about completeness of our decision support system. Otherwise, certain constrains and assumptions to the domain area were considered, and limited sets of feature extraction methods, classifiers and data set characteristics were considered in order to guarantee the desired level of confidence in the system when solving a bounded set of problems.

Acknowledgements

This research is partly supported by the COMAS Graduate School of the University of Jyväskylä, Finland. We would like to thank the UCI ML repository of databases, domain theories and data generators for the data sets, and the MLC++ library for the source code used in our studies.

Bibliography

[Aivazyan, 1989] Aivazyan, S.A.: Applied Statistics: Classification and Dimension Reduction. Finance and Statistics, Moscow, 1989.

[Aladjem ,1994] Aladjem, M. Multiclass discriminant mappings. Signal Processing, 35:1-18, 1994.

[Bellman, 1961] Bellman, R., Adaptive Control Processes: A Guided Tour, Princeton University Press, 1961.

[Blake, 1998] Blake, C.L., Merz, C.J. UCI Repository of Machine Learning Databases [http://www.ics.uci.edu/~mlearn/MLRepository.html]. Dept. of Information and Computer Science, University of California, Irvine CA, 1998.

[Fayyad, 1996a] Fayyad U.M., Piatetsky-Shapiro G., Smyth P., and Uthurusamy. Advances in Knowledge Diskovery and Data Mining. *AAAI Press/MIT Press*, 1996

[Fayyad, 1996b] Fayyad U.M. Data Mining and Knowledge Discovery: Making Sense Out of Data, *IEEE Expert*, Vol. 11, No. 5, Oct., 1996, pp. 20-25

[Fukunaga, 1991] Fukunaga, K. Introduction to Statistical Pattern Recognition. Academic Press, London, 1991.

[Hall, 2000] Hall, M.A. Correlation-based feature selection of discrete and numeric class machine learning. In Proc. Int. Conf. On Machine Learning (ICML-2000), San Francisco, CA. Morgan Kaufmann, San Francisco, CA, 359-366, 2000.

[Jackson, 1999] Jackson P. Introduction to Expert Systems, 3rd Edn. Harlow, England: Addison Wesley Longman, 1999.

[Jolliffe, 1986] Jolliffe, I.T. Principal Component Analysis. Springer, New York, NY. 1986.

[Kohavi, 1996] Kohavi, R., Sommerfield, D., Dougherty, J. Data mining using MLC++: a machine learning library in C++. Tools with Artificial Intelligence, IEEE CS Press, 234-245, 1996.

[Krzanowski, 1994] Krzanowski, W.J., & Marriott, F.H.C. Multivariate analysis part 2: Classification, Covariance structures and repeated measurements. London: Edward Arnold, 1994

[Liu, 1998] Liu H. Feature Extraction, Construction and Selection: A Data Mining Perspective, ISBN 0-7923-8196-3, Kluwer Academic Publishers, 1998

[Merz, 1996] Merz, C.: Dynamical Selection of Learning Algorithms. In: D.Fisher, H.-J.Lenz (eds.), Learning from Data, Artificial Intelligence and Statistics, Springer-Verlag, NY, 1996.

[Oza, 1999] Oza, N.C., Tumer, K. Dimensionality Reduction Through Classifier Ensembles. Technical Report NASA-ARC-IC-1999-124, Computational Sciences Division, NASA Ames Research Center, Moffett Field, CA, 1999.

[Quinlan, 1993] Quinlan, J.R. 1993. C4.5 Programs for Machine Learning. San Mateo CA: Morgan Kaufmann.

[Tsymbal, 2003] Tsymbal A., Pechenizkiy M., Puuronen S., Patterson D. Feature Extraction for Dynamic Integration of Classifiers (to be submitted), 2003.

[Tsymbal, 2002] Tsymbal A., Puuronen S., Pechenizkiy M., Baumgarten M., Patterson D. Eigenvector-based feature extraction for classification, In: *Proc. 15th Int. FLAIRS Conference on Artificial Intelligence*, Pensacola, FL, USA, AAAI Press, 354-358, 2002.

[Tsymbal, 2001] Tsymbal A., Puuronen S., Skrypnyk I. Ensemble feature selection with dynamic integration of classifiers, In: Int. ICSC Congress on Computational Intelligence Methods and Applications CIMA'2001, Bangor, Wales, U.K, 2001.

[Turban, 2001] Turban E. and Aronson J.E. Decision Support Systems and Intelligent Systems, Prentice-Hall, 2001.

[William, 1984] William D.R., Goldstein M. Multivariate Analysis. Methods and Applications. *ISBN 0-471-08317-8,* John Wiley & Sons. 1984, 587 p.

Author information

Mykola Pechenizkiy - Department of Computer Science and Information Systems, University of Jyväskylä, P.O.Box: 35, Jyväskylä 40351, Finland; e-mail: mpechen@cs.jyu.fi

Seppo Puuronen – Department of Computer Science and Information Systems, University of Jyväskylä, P.O.Box: 35, Jyväskylä 40351, Finland; e-mail: sepi@cs.jyu.fi

Alexey Tsymbal - Department of Computer Science, Trinity College Dublin, College Green, Dublin 2, Ireland; e-mail: Alexey.Tsymbal@cs.tcd.ie

DIFFERENTIAL BALANCED TREES AND (0,1) MATRICES[2]

H. Sahakyan, L. Aslanyan

Abstract*: Links and similarities between the combinatorial optimization problems and the hierarchical search algorithms are discussed. One is the combinatorial greedy algorithm of step-by-step construction of the column-constraint (0,1) matrices with the different rows. The second is the base search construction of databases, - the class of the well known weight-balanced binary trees. Noted, that in some approximation each of the above problems might be interpreted in terms of the second problem. The constraints in matrices imply the novel concept of a differential balance in hierarchical trees. The obtained results extend the knowledge for balanced trees and prove that the known greedy algorithm for matrices is applicable in the world of balanced trees providing optimization on trees in layers.*

Keywords: *search, balanced trees, (0,1)-matrices, greedy algorithm.*

1. Introduction

In this paper a new class of weight-balanced trees [K, 1973] is introduced and investigated. In some sense these are extensions of the concept of the bounded balanced trees. Bounded balanced trees were analysed in various

[2] The research was supported by INTAS 00-397 and 00-626 Projects.

publications, e.g. [R, 1977], being the main data structure of search in dynamic databases. In Section 2 below the height estimate for bounded-balanced trees is considered and an estimate for the weight-balanced trees with the newly introduced differential balances and constraints is obtained.
The theory of weight-balanced trees is very rich. Practically this is also the base model of hierarchical search and decision support. In search, several restrictions in terms of balances are applied in a dynamic environment with insertion of new and deletion of obsolete search elements. The balances in nodes are under the change during this process. In a dis-balanced node rotations are used to correct the situation. Several queries, related to these models are traditional. Which is the tree height in a given balance and in a given set of search elements? Which is the average path length in a search tree? A particular new postulation is the following. Is it possible to construct a tree or to construct all the trees that may appear in a search model with the given constraints? This is a particular interest of the current paper.
The stated problem will be studied in several extensions, which are also a typical element of search models. E.g. - some specific classes of balanced trees, called trees of bounded heights, introduced in [A, 1989], [A, 1999].
The concept of bounded-balance is extended in Section 3, defining layer-constraint balanced trees. The idea of layer-constraints is then developed in Section 4, considering a practical extension of the concept of weight-balanced trees - defining summary balances for tree layers. This structure is related to mentioned combinatorial problem – constructing the constraint based (0,1)-matrices with different rows. In [S, 1986], [S, 1995] a greedy algorithm is constructed for solving the mentioned combinatorial problem and it is proven optimal in local steps. The algorithm for solving this problem is reducible to the constructing of weight-balanced trees by the given summary differential balances in layers. Similarly, in the world of balanced trees this proves a heuristic optimization on trees in layers.

2. Bounded-balanced Trees

Let T_m be a non-empty extended binary tree [R, 1977] with m leaves, and T_l and T_r are the left and right root-subtrees of T_m. We denote by l and r the numbers of leaves of T_l and T_r (called weights) and assume that $l > 0$ and $r > 0$. Then $m = l + r$.

Definition [R, 1977]. The fraction l / m is called the balance (left, fractional) of T_m in root vertex, being denoted by $\beta(T_m)$. $\beta(T_m)$ expresses the ratio weight of the left root-subtree and it obeys the condition $0 < \beta(T_m) < 1$.

Definition [R, 1977]. For a given α, $0 \le \alpha \le 1/2$, T_m is called an α-balanced tree (or a tree from $WB[\alpha]$) if

1) $\alpha \le \beta(T_m) \le 1 - \alpha$,
2) left and right subtrees of T_m belong to $WB[\alpha]$.

We assume by definition that the empty binary tree belongs to $WB[\alpha]$.

$WB[0]$ is the set of all binary trees and $WB[1/2]$ is the set of all perfectly balanced (with the equal left and right subtrees weights in each node) binary trees, and this is possible, when the number of leaves has the form 2^k $(k \ge 1)$.

The maximum possible height $h_\alpha(m)$ of trees from $WB[\alpha]$ is estimated in [R, 1977] – by the consideration of the most asymmetric trees of $WB[\alpha]$:

$$h_\alpha(m) \le \frac{\log m}{\log(1/(1-\alpha))} \qquad (1)$$

It is also important to treat the question: given the binary trees with m leaves and with heights, restricted by a given number n, then - how "unbalanced" may be the trees, - which is the allowable minimum value for α? The answer (in a form of a sufficiency condition) is given by the lemma below, using the monotonity of (1).

Lemma 1. If $\alpha \geq 1 - \frac{1}{\sqrt[n]{m}}$, then $h_\alpha(m) \leq n$.

The $\alpha \geq 1 - \frac{1}{\sqrt[n]{m}}$ implies $\frac{1}{1-\alpha} \geq 2^{\frac{\log m}{n}}$, and then $\log(1/(1-\alpha)) \geq \frac{\log m}{n}$, and $\frac{\log m}{\log(1/(1-\alpha))} \leq n$. For those α, $h_\alpha(m) \leq \frac{\log m}{\log(1/(1-\alpha))} \leq n$ by (1).

Now let us turn to the concept of balances in terms of differences of weights between subtrees.

Definition. The difference $r - l$ is called the differential balance (right) of T_m in the root vertex, denoted by $\delta(T_m)$. It obeys the following condition: $1 - m < \delta(T_m) < m - 1$.

Definition. For a given d, $0 \leq d < m-1$, T_m is called differential-balanced tree with balance d (or a tree from $WDB[d]$) if

1) $-d \leq \delta(T_m) \leq d$,
2) left and right subtrees of T_m belong to $WDB[d]$.

The two balance schemes are tightly related. Let us formulate the base relations between the fractional and differential balances.

Let T_m be an extended binary tree with m leaves, and v_i is a vertex (not leaf) of T_m. We denote by m_i the weight of subtree rooted at v_i, and by l_i and r_i - the weights of its left and right subtrees, correspondingly. Starting at this point we will assume also, that l_i-s are not greater than r_i-s.

If T_m is an α-balanced tree then for each vertex v_i we have $\frac{l_i}{m_i} \geq \alpha$ by the definition. Let's estimate the weight differences between right and left subtrees for each v_i:

$r_i - l_i = m_i - 2l_i = m_i(1 - 2\frac{l_i}{m_i}) \leq m_i(1-2\alpha)$. Hence $r_i - l_i \leq \max_i m_i(1-2\alpha) = m(1-2\alpha)$.

Conclusion is that an α-balanced tree T_m is a differential balanced tree with $d = m(1-2\alpha)$.

Now let T_m is a differential balanced tree with balance d. For each vertex v_i, $r_i - l_i \leq d$ by the definition. Let's estimate the fraction $\frac{l_i}{m_i}$; $\frac{l_i}{m_i} = \frac{m_i - (r_i - l_i)}{2m_i} = \frac{1}{2}(1 - \frac{r_i - l_i}{m_i}) \geq \frac{1}{2}(1 - \frac{d}{m_i}) \geq \frac{1}{2}(1 - \frac{d}{m})$.

Thus, a differential balanced tree T_m with balance d, is an α-balanced tree, with $\alpha = \frac{1}{2}(1 - \frac{d}{m})$. A more correct estimate is:

$$\alpha = \frac{1}{2}(1 - \max_i \frac{\min\{d, m_i - 2\}}{m_i}).$$

Next we consider the estimate of height of trees from $WDB[d]$, constructing the most asymmetric trees in this class. On each layer the subtrees with greatest weights have been partitioned into the subtrees with maximization of weights differences. Then the height estimate is the length of these "maximum weighted" branches.

On the first layer we get subtrees of weights $\frac{m+d}{2}$ and $\frac{m-d}{2}$. We will follow only the branch of weight $\frac{m+d}{2}$. On the next layer we will get a subtree of weight $\frac{\frac{m+d}{2}+d}{2}=\frac{m+3d}{4}$. In continuation, let k is the minimal index, where the maximal subtree weight becomes less than d. At that point the maximum weight doesn't exceed $\frac{m+(2^k-1)d}{2^k}$.

$$\frac{m+(2^k-1)d}{2^k}=\frac{m-d}{2^k}+d\text{, therefore }\frac{m-d}{2^k}<1\text{, and }k>\log(m-d).$$

Resuming, we receive, that after at most $\log(m-d)+1$ steps the weight of maximal subtree is less than d. If $d\le 1$, then the tree construction is complete, and we get a tree with the height estimate $\log(m-d)+1$. Otherwise we continue the process, with the arbitrary partition of subtrees. At most $d-1$ steps will be required. We receive the following final estimation – the heights of trees from $WDB[d]$ are restricted by $\log(m-d)+d$.

Now we treat the question about the constraints on balances when given that the heights are restricted. Let us consider the binary trees with m leaves, and heights, restricted by the given number n. The counterpart of Lemma 1 is the following proposition:

Lemma 2. If the differential balance d obeys: $\log(m-d)+d\le n$, then the height of tree with m leaves is restricted by n.

A practical note. The concept of differential balancing is reasonable to apply on trees as far as the weights of subtrees are greater than d, therefore - on layers of at most $\log(m-d)+1$ far from the root.

3. Layer-constrained Weight-balanced Trees

At this point the concept of differential balances is introduced and the general comparison with the base scheme – the weight-balanced trees is outlined. The particular properties of differential balances are that these are flexible on tree layers. The balance constraints may vary from layer to layer and/or the constraints might be given in terms of summary balances. In some cases it is important to apply these structures in the traditional case of the weight-balanced trees. These issues are considered below.

Definition. For a given α_i, $0\le\alpha_i\le 1/2$, we say that T_m is α_i-balanced on layer i, if for each subtree T_{i_j} - rooted at layer i, $\alpha_i\le\beta(T_{i_j})\le 1-\alpha_i$.

Definition. Given numbers $\alpha_0,\cdots,\alpha_k$, where $0\le\alpha_i\le 1/2$, $i=0,\cdots,k$. We say that T_m is a tree from class $WB[\alpha_0,\cdots,\alpha_k]$, if T_m is α_i-balanced on layer i.

The leaves may be composite in $WB[\alpha_0,\cdots,\alpha_k]$ (when k-sequences are not enough to differentiate the nodes, the composite nodes may remain consisting of sets of virtual leaves). On the other hand, part of the balance values (a last portion) may be redundant. Consideration of the most asymmetric trees and paths in

$WB[\alpha_0,\cdots,\alpha_k]$ gives the following estimation: the weights of subtrees (virtual at this point) of k -th layer are restricted in size by $(1-\alpha_0)(1-\alpha_1)\cdots(1-\alpha_k)m$. If there exists h, $h \le k$, such that $(1-\alpha_0)(1-\alpha_1)\cdots(1-\alpha_h)m \le 2$, then the height of the tree is restricted by h.

Now we consider layer constrained weight-balanced trees in sense of differential balances.

Definition. For a given d_i, $0 \le d_i < m-1$, we say that T_m has d_i differential balance on layer i, if for each subtree T_{i_j} - rooted at layer i, $-d_i \le \delta(T_{i_j}) \le d_i$.

Definition. Given numbers $d_0,\cdots,d_k$, where $0 \le d_i < m-1$, $i=0,\cdots,k$. We say that T_m is a tree from class $WDB[d_0,\cdots,d_k]$, if T_m has differential balances d_i on layers i.

Similarly with the class $WB[\alpha_0,\cdots,\alpha_k]$, the leaves may be composite, or some last balance values may be redundant for trees of $WDB[d_0,\cdots,d_k]$. Consider the most asymmetric trees of the class $WDB[d_0,\cdots,d_k]$. Using reasoning, similar to the used above, we get that the weights of subtrees on the k -th layer of trees are restricted by $\dfrac{m+d_0+2d_1+\cdots+2^k d_k}{2^{k+1}}$.

If there exists h, $h \le k$, such that $\dfrac{m+d_0+2d_1+\cdots+2^h d_h}{2^{h+1}} \le 1$, then the overall height of tree is restricted by h. It is easy to see that $d_0,\cdots,d_k$ must obey in this case very specific restrictions, which limits the selection and the meaning of differential balances.

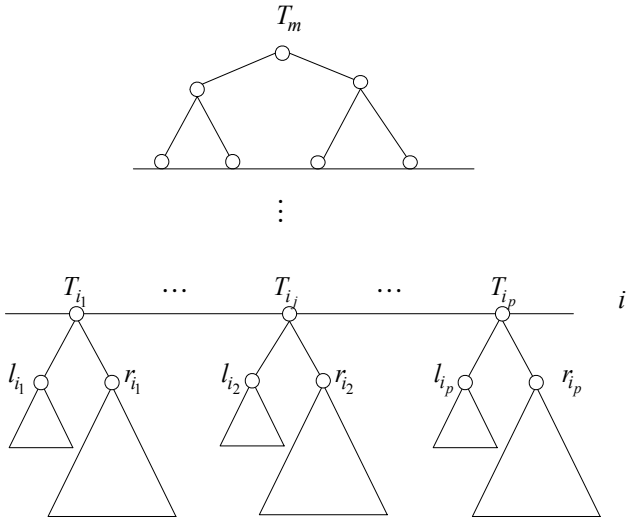
4. Summary Differential Balanced Tress and (0,1)-matrices

Definition. Given numbers $D_0,\cdots,D_n$ where $0 \le D_i < m-1$, $i=0,\cdots,n$. We say that T_m has D_i summary differential balance on i -th layer, if $R_i - L_i \le D_i$, where R_i is the sum of weights of the all right subtrees rooted at the layer i and L_i is the same sum for the left subtrees.

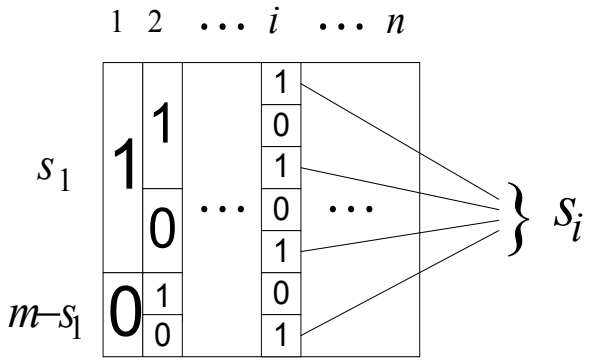
Definition. T_m is called $\{D_0,\cdots,D_n\}$ summary differential balanced tree if the summary balance on i -th layer equals to D_i, $i=0,\cdots,n$.

Let $T_{i_1},\cdots,T_{i_p}$ are subtrees rooted at i -th layer having the weights $m_{i_1},\cdots,m_{i_p}$ correspondingly, and let l_{i_j} and r_{i_j} are the weights of left and right subtrees of T_{i_j}. Then $R_i = \sum_{j=1}^{p} r_{i_j}$ and $L_i = \sum_{j=1}^{p} l_{i_j}$.

Usually the balance criterion restricts the weights of subtrees, making it possible to optimize the height of the tree. The summary balance allows any weights for subtrees, requiring only satisfying the given summary constraints on layers. This is a week form constraints for constructing an optimized decision tree. The most asymmetric trees are very diverse in this case. In next point the combinatorial origin of these differential schemes will be described. In classes of summary balanced trees the problem is in existence of a binary tree with the given characteristics $D_0,\cdots,D_n$, and in case of existence – the algorithmic construction of such trees. In special cases the issue of construction of trees might be the interest, when an additional functional for optimization is given. In particular, optimality might be required as for subtree weights on layers, for number of subtrees on layers, a special functional optimization, etc.



Here is the combinatorial counterpart of the scheme of the summary differential balances. Given an integer vector $S=(s_1,\cdots,s_n)$, where $0\le s_i\le m$, $i=1,\cdots,n$. The interest is in (0,1) -matrices of size $m\times n$ (m is the number of rows) with s_i 1's in i-th column and with different rows. This is the existence problem. The corresponding optimization problem is in minimization of the number of the possible repeated rows. The problem might be solved, in particular, by algorithms, constructing the matrices in a column-by-column fashion, by partitioning the sets of similar (equal) rows received in a previous step. The first column has been constructed substituting s_1 1's and $m-s_1$ 0's. Without loss of generality we assume that the 1's are substituted on the first s_1 rows. The second column has been constructed by partitioning the intervals (sets of similar rows) of the first column (of lengths s_1 and $m-s_1$) - substituting 1's and 0's on these intervals such that the summary length of intervals one-intervals (where all 1's are substituted), is equal to s_2, and the summary length of zero-intervals (where 0's are substituted), is equal to $m-s_2$. The partitioning of intervals for current k-th column is arbitrary, providing only that the summary length of all one-intervals is equal to s_k. Such construction provides the following property: for each pair of rows, (i,j), where rows i and j belong to different intervals, we have that the i-th and j-th rows are different. Within each interval we have sets of equal rows. The intervals with 1 length in each column don't participate in further partitioning, but they are used (substituting 1 or 0) to provide the summary values s_k and $m-s_k$ on the current k-th column. When in some column there are all 1 length intervals, then all rows are different, and the required matrix is constructed. The remainder columns might be constructed arbitrarily. The graphical scheme is the following:



This is the existence problem as was mentioned. In the similar optimization problem the row repetitions is to be minimized. Let, in a current state of construction we have intervals of lenghts $m_{n_1},\cdots,m_{n_p}$ (greater than 1) on

the n-th column. Each of the $m_{n_1},\cdots,m_{n_p}$ intervals consists of the same rows repetitions. The number of pairs of rows - (i,j), where rows i and j are the same, equals $\sum_{j=1}^{p} C^2_{m_{n_j}} = \frac{1}{2}\sum_{j=1}^{p} m_{n_j}(m_{n_j}-1)$, so this is the subject for optimization.

The construction of (0,1)-matrices might be represented by binary trees. We construct a tree T_m with m leaves. The matrix with m rows corresponds to the root vertex. The submatrix with the first s_1 rows from the first step corresponds to the right subtree, and the submatrix with $m-s_1$ rows corresponds to the left subtree, etc. When the current submatrix consists of a single row, we get a leaf. When for any k, $k \le n$, the k-th layer contains leaves only, then the construction is completed. Otherwise as a result of construction on n-th layer we receive a set of subtrees of weights $m_{n_1},\cdots,m_{n_p}$. The constructed trees belong to the class of summary differential balanced trees with summary balances $D_1,\cdots,D_n$, for the given balances $D_i = s_i - (m-s_i),\quad i=1,\cdots,n$.

[S, 1986], [S, 1995] provide an approximation greedy algorithm, which constructs the target (0,1)-matrices in the above described column-by-column fashion of partitioning. The algorithm provides the optimal construction of each column – i.e. the construction, which provides the maximal number of new (i,j) pairs of different rows in each step. It is proven that the optimal construction of each column is provided by partitioning, which distributes the difference $s_k-(m-s_k)$ "homogeneously" on all current non atomic intervals. Returning to the trees terminology, the matrix constructed by the greedy algorithm implies subtrees on each layer of tree, partitioned such that the difference $R_i - L_i = D_i$ is distributed "equally" on all current subtrees.

So this describes the construction of trees in class of summary differential balanced trees providing the local optimum for the functional from the related combinatorial problem of (0,1)-matrices.

A last note. Let the subtrees of i-th layer with weights $m_{i_1},\cdots,m_{i_p}$ are partitioned into the subtrees with weights $l_{i_1},r_{i_1},\cdots,l_{i_p},r_{i_p}$ correspondingly. We denote $d_{i_1}=r_{i_1}-l_{i_1},\cdots,d_{i_p}=r_{i_p}-l_{i_p}$. Then the differential balance on i-th layer is equal to $\max_{1\le j\le p} d_{i_j}$. Since D_i is distributed "equally" by the greedy partition, $\max_{1\le j\le p} d_{i_j}$ will have the minimum value among all possible partitions. This is the following property: an algorithm, which is locally optimal by means of (0,1)-matrices, is locally optimal also by means of construction of trees of minimal height in class of summary differential balanced trees.

Conclusion

Resuming, - in problem of constructing the summary balanced binary trees with given differential balances of layers, and with height minimization, it is possible to apply the given above combinatorial greedy algorithm, and then the resulting tree has a property that the maximal value of the differential balances on tree layers are optimal - minimal. In terms of search trees this is an extension of perfect balanced trees on layers, when additional constraints are applied.

Bibliography

[K, 1973] D. Knut,.The Art of Computer Programming, vol.3. Sorting and Searching, Addison-Wesley Publishing Company, 1973.

[R, 1977] E. Reingold, J. Nivergelt and N. Deo. Combinatorial Algorithms, Theory and Practice, Prentice-Hall, 1977.

[A, 1989] A. Andersson. Improving Partial Rebuilding by Using Simple Balance Criteria. In Frank Dehne, Jorg-Rudiger Sack, and Nicola Santoro, editors, 1st International Workshop on Algorithms and Data Structures, Lecture Notes in Computer Science, volume 382, pages 393--402. Springer-Verlag, 1989.

[A, 1999] A. Andersson. General Balanced Trees, Journal of Algorithms 30, 1-18, (1999).

[R, 1966] H. J. Ryser. Combinatorial Mathematics, 1966.

[S, 1986] H. A. Sahakyan. Greedy Algorithms for Synthesis of (0,1) Matrices, Doklady Akademii Nauk Arm. SSR, v. 83, 5, pp. 207-209 (1986).

[S, 1995] H. Sahakyan. Hierarchical Procedures with the Additional Constraints, II Russia, with participation of NIS Conference, Pattern Recognition and Image Analysis: New Information Technologies, Ulianovsk, 1995, pp.76-78.

Author information

Hasmik Sahakyan - Institute for Informatics and Automation Problems, NAS Armenia, P.Sevak St. 1, Yerevan-14, Armenia; e-mail: hasmikl@ipia.sci.am

Levon Aslanyan – Institute for Informatics and Automation Problems, NAS Armenia, P.Sevak St. 1, Yerevan-14, Aremenia; e-mail: lasl@sci.am

KNOWLEDGE LEARNING TECHNOLOGY FOR INTELLIGENT TUTORING SYSTEMS

Taran T.A., Sirota S.V.

Abstract*: In this work we suggest the technology of creation of intelligent tutoring systems which are oriented to teach knowledge. It is supposed the acquisition of expert's knowledge by using of the Formal Concept Analysis method, then construction the test questions which are used for verification of the pupil's knowledge with the expert's knowledge. Then the further tutoring strategy is generated by the results of this verification.*

Keywords*: Computer tutoring, Formal Concept Analysis.*

Introduction

The conception of the up today's computer tutoring systems is based on teaching knowledge which represented like a pack of the facts and rules. In [3] the learning types systematization was suggested accordant to cognitive levels used in studying. There are four cognitive levels marked: (1) creative, (2) analogy-generalization, (3) explanation and (4) programming. The 4-th level means the training to solve certain type of tasks, in the 3-d level the studying goes by explanation, the aidless work with exercises and problems needs the analogy-generalization, the highest, creative, level is supposed the pupil to ideate himself the concepts and relations between them. The interaction between the pupil and the tutoring system can hit different cognitive levels. It is clear that using of all four levels is optimal strategy of teaching. But up to day tutoring systems are not able to interact on creative level [3].

In [9] two basic directions of the studying process are marked: the concept learning and training. The concept learning has next stages:

- The description of objects attributes, quality and quantity characteristics of the objects and processes in the domain;
- The eduction of concepts;
- The definition of the relations between concepts;
- The definition of dependences on the sets of attributes and characteristics of objects and processes;

In concept learning tutoring systems the cardinal problems are representation of the facts, description of the attributes and relations and rigorous definition of the concepts to be memorized. The memorizing is the main procedure pupil to do. The further action of the system is directed to monitoring of knowing the main concepts of domain. The training started when the pupil have got certain set of knowledge. Then we create some problems to make pupil use his knowledge for solving. Often the pupil is suggested the gradually more difficult tasks. During the solving the pupil can return to the material studied and get some help refilling his knowledge.

In present tutoring systems the most attention is paid to the second stage. The training subsystems are most active component while during concept learning the developers are sated with passive presentation. As researches showed [8], the pupil's knowledge must to reach certain level to avoid antagonism for further tutoring.

In this paper we suggest the tutoring systems technology of interactive conceptual learning including the test of pupil's knowledge. This technology is embodied in software tools. The main features of these software are: the extraction of expert knowledge; the formalization of the domain's concepts; the construction of the knowledge base; the semi automatic production of the test questions; the construction of the pupil's cognitive model; the estimation of the pupils knowledge; the creation of the tutoring strategy.

The main task of the primary learning is habituation with the domain concepts. For the formalization of this concepts we use the Formal Concept Analysis.

Such approach represents the formal concept like as a pair <intent; extent> where the extent represents the set of objects and the intent is the set of their attributes. For the concept extraction we use the formal context for the corresponding part of the domain. The formal context is represented by the table <object; attribute> in witch every attribute is marked if it is the proper of certain object. For every formal concept maximal nested full submatrix corresponds. The set of concepts is ordered by the relation superconcept - subconcept and forms a full lattice. For testing of the pupil we use the set of questions witch is generated automatically using this lattice. The purpose of testing is to estimate how the pupil grasped the concepts and relations between them. The test questions are mainly of the closed type. The main problem of their recomposing is a choice of destructors. For this we suggest to use the nearest to the concept testing conceptual lattice elements. The new conceptual lattice is creating using the pupils answers. This lattice represents the domain in the pupils mind. Then we compare it with the master lattice. The differences between them are used to generate a tutoring strategy.

The Concept Presentation in Tutoring System

The method of Formal Concept Analysis is used for knowledge representation of problem domain. It was suggested by R. Wille [4, 5] and at the present time is successfully applying in the problems of data-mining and computer tutoring [1, 2, 10]. The backbone is follows. Let us consider the set of objects V and the set of attributes A with an arbitrary relation $I \subseteq V \times A$, such that pIa, where $p \in V$, $a \in A$, if and only if a is the attribute of object p. Then $K = (V, A, I)$ is called the *formal context.* The binary matrix defines the correspondence of objects and attributes. Let define the correspondence [6]:

$$P' := \{y \in A \mid xIy \text{ for all } x \in P\}, \text{ for } P \subseteq V,$$
$$G' := \{x \in V \mid xIy \text{ for all } y \in G\}, \text{ for } G \subseteq A.$$

Then the pairs (P, G) satisfying $P \subseteq V$, $G \subseteq A$, $P' = G$, $G' = P$ are called *the formal concepts* of the formal context $K = (V, A, I)$. The set of objects P amounts the *extent* of concept and the set of all their attributes amounts its *intent.* Every object $p \in P$ have all attributes from subset G. So, the formal concept is the set of objects from domain such that every one of them have all attributes from certain subset of attributes of that objects.

The set of formal concepts (P, G), where $P \subseteq V$, $G \subseteq A$, is partially ordered by the relation: $(P_1, G_1) \leq (P_2, G_2)$, if $P_1 \subseteq P_2$ and $G_2 \subseteq G_1$, and form a complete lattice $L(K)$, called *the concept lattice* of the context K [4]. The pair (P_1, G_1) is called the subconcept of the concept (P_2, G_2), and the pair (P_2, G_2) is called the superconcept of the concept (P_1, G_1).

The concept lattice can be represented by the line diagram (Hasse diagram) in which every node of the concept lattice is corresponded by the concept from context. The dual isomorphism on the concept lattice reflects the inverse between the intent and extent of the concepts: the bigger is extent the less is intent

Example. The formal context "Geometry Figures" is represented in the Table 1. This context is based on the set of geometry figures and the set of their attributes. Maximal nested full submatrix corresponds for every concept.

For example, marked submatrix in Table 1 corresponds to formal concept {<*Triangle*, *Tetragon* (*quadrangle*), *Pentagon*, *Hexagon*>, <*Vertexes*, *Area*, *Sides*, *Angle*>} witch can be defined like a "*polygon*". Hereby formal concept is the set of objects from domain such that everyone of them has all attributes from some subset of domain attributes.

Table 1. Context «Geometry Figures»

	Has Vertexes	Has Length	Is Line	Has Area	Has Sides	Has Angle
Point						
Straight Line			x			
Half Line	x		x			
Straight Line Segment	x	x	x		x	
Angle	x				x	x
Circle				x		
Circumference		x	x			
Curve Line		x	x			
Poly Line	x	x	x		x	x
Triangle	x			x	x	x
Tetragon (quadrangle)	x			x	x	x
Pentagon	x			x	x	x
Hexagon	x			x	x	x

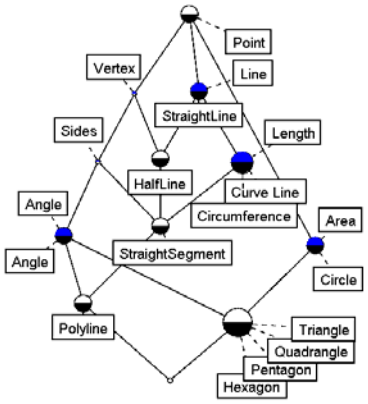


Fig. 1. Conceptual lattice of the context «Geometry figures»

The diagram of conceptual lattice is represented in a Figure 1. By this lattice we can easy find which attributes corresponds to proper object. One concept *(P, G)* corresponds to each node on the diagram. *P* includes the labels of all objects directly underlying, and *G* includes labels of all attributes lying directly above of the given node. The node on the diagram is marked by the label *(p, a)*, where *p* includes the labels of objects, and *a* – labels of attributes directly appropriate to the given node. For example, in Fig. 1 the node with a label «*angle*» and the attribute «*has angle*» corresponds to the concept {<*Angle, Poly line, Triangle, Tetragon (quadrangle), Pentagon, Hexagon*>, <*Vertexes, Sides, Angle*>}. In this context the concept with the empty set of attributes {<*Point*>,∅} corresponds to the lattice unit. The point is a primary concept which can not be defined by the set of properties (attributes).

On a concept lattice it is easy to find the *common* attributes for several objects and the *unique* attributes corresponding only to one of objects. For example objects «*triangle*» and «*circle*» has one common attribute "*area*" Likewise easy to find all objects that have certain attribute. For example we can see that all of "Straight Line", "Half Line", "Straight Line Segment", "Circumference", "Curve Line", "Poly Line" are "lines"

One more important result is that the construction of a concept lattice defines the partial order on a set of objects and attributes. Partial ordering of objects and attributes allows us to reveal the dependence between them. Let $K = (V, A, I)$ is a formal context, $X \subseteq A$, $Y \subseteq A$. Then $X \rightarrow Y$, т.е. *X implies Y,* or the set of attributes *Y depends* on the set of of attributes *X,* if all objects from $P \subseteq V$, that have attributes from *X,* also have all attributes from *Y,* i.e. $X' \subseteq Y'$ (or $Y \supseteq X'$). In this case the concept containing *X* stands lower of the concept containing *Y* on the diagram of a concept lattice.

For example we can see on the diagram that the property "*to have a angle*" is followed by the properties "*to have sides*" and "*to have a vertex*", and property «*to have a length*» is followed by the property «*to be a line*».

The described above features of the concept lattices allow to arrange the pupil's knowledge about subject domain and from other side allow to semi automate the process of the test questions production for the checkout of a pupil's knowledge.

Representation of the Subject Domain Knowledge

The suggested tools environment uses material is represented by traditional means of hypermedia such as texts, pictures, animation. For the automation of the tutoring and knowledge checkup process we need the structures reflected the connection between the parts of material studied and connections between the main concepts inside every part. The common structure of the subject domain is represented by the ontology. The main structure witch corresponds the knowledge by the fragments is the semantic networks. The construction of the semantic network is process to be difficult formalized. Mainly supposed, that the expert is to do it. He describes the connections between the concepts and objects of the subject domain by the means of graphics visualization [7]. Using the formal contexts allows automating partially the semantic network construction. It is easier to preset the objects and their properties and then using concept lattice to define the main concepts and relations between them. The same process can be used for the checkout of the pupil's knowledge. Constructing his conceptual lattice the pupil reflects the concepts system like he ideates it himself. The main criteria of the full retention of the material are the isomorphism between the conceptual model of the pupil and the master model created by expert.

Transformation of the concept lattice into the semantic network is under next rules. We define the context objects like primary concepts of the semantic network. Then single-place predicates-attributes like *PredicateName*(*X*) *we* transform into two-place predicates like *Function*(*PredicateName, X*), where the *Function* means belonging *X* to the class, type, set or describes other connection between object and attribute (the sets of the objects and attributes can intersect i. e. the same essence can be an object and an attribute. (In our example such essence is a "*angle*"). Than we specify the predicates: every variable is replaced by the object from the predicate truth set according the formal context matrix.

For illustration let us see example above. In context «Geometry Figures» every attribute is a single-place predicate defined on preset set of objects – Geometry figures. Case in point "*X is a Line", "X has a Area"* etc. The predicates truth sets marked in a Table 1. We transform all this predicates into two-place predicates *Is(X, Y)* and *Has(X, Y)*. There are only two in our case. Decomposition of the lattice connections allows to construct hierarchy semantic network (fig. 2).

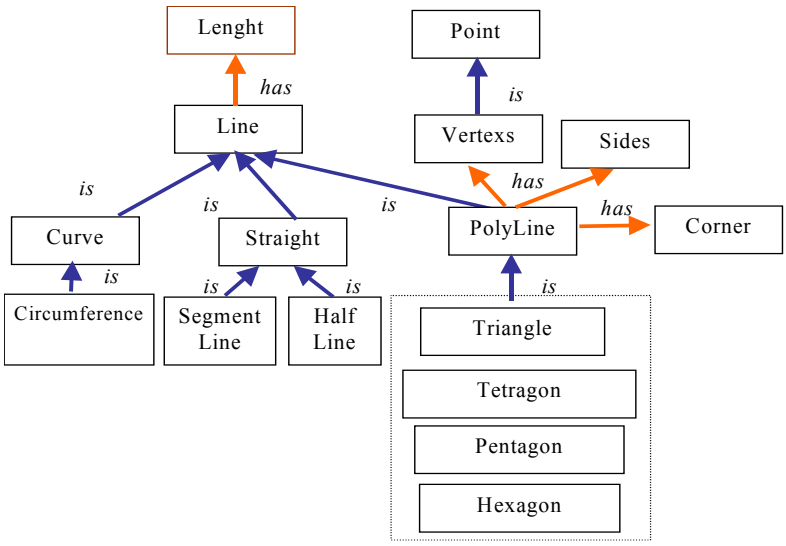


Fig. 2 Hierarchical semantic network for the Context «Geometry Figures»

The data input is realized by creating and filling the context tables. It is possible to organize the row and column titles like hyperlinks to pages that describes the definitions of objects, backbones of their properties and explanations the attributes.

Some objects can have attributes only if the attributes of higher level exist. This allows extracting the attributes of lower level into separate context linked with the main context by the attribute of higher level. Only objects with this attribute are included in it. For example the figures attributed "*Has area*" can be separated to the new context.

From the other side some objects can have additional attributes not immanent to other objects. In this case nested context can be created. For example nested context for the object "*Triangle*" will has attribute "*Equilateral*", "*Isosceles*", "*Right*" etc. As the result the context hierarchy (Hypercontext) can be constructed. This structure has a lot of advances. Hypercontext can be processed whole and partially with free choice of processing deepness levels, separate branches, or single contexts. Thus exclude dismissing of the dependent attributes, if the object does not have determinative attribute of higher level. Often necessity of using quantitative attributes appears. Than we use many-valued contexts [4].

Context processing includes next steps: generating of the concept set, creating the line diagram of the concept lattice, extracting of the implications basis. Resulted knowledge system is used by the next way:

- Concepts are raw data for the test generation;
- Implications are used by expert for checkout of context fullness and for test generation;
- Concept lattice is used by test constructor and choice of the tutoring strategy;
- Line diagram is used for visualization of the domain structure fragment.

The implications proved from context can be differed into three groups.

A) Non common sensitive implications. These are ones having false premise on the set of all context objects. Such implications are excluded from further processing.
B) Right (True) implications witch reflects the domain relations correctly. These are relations where the intent of premise and consequence coincides. The accuracy of such implications is 100%.
C) Plausible implications witch reflects the domain relations correctly, but not for all objects, i.e. the intent of premise is bigger than intent of consequence.

For example in context «Geometry Figures» the implication basis includes the right implications:

1. If the Object «*has sides*», that it «*has vertex*» (7 objects);
2. If the Object «*has angle*», «*has vertex and sides*» and (6 objects);
3. If the Object «*has length*», that it «*is line*» (4 objects);
4. If the Object «*has vertex and area*», that it «*has sides and angle*» (4 objects;
5. If the Object «*is line and has vertex and has sides*», that it «*has length*» (2 objects;
6. If the Object «*is line and has vertex and length*», that it «*has sides*» (2 objects;
7. If the Object «*is line and has area* », that it «*has vertex, length, sides, and angle*» (0 objects).
   These implications are apparent, because their accuracy in this context is 100%. Besides this there are plausible implications.
8. If the Object «*has vertex*», that it «*has sides*» (accuracy – 86%).
9. If the Object «*has vertex and sides*», that it «*has angle*» (accuracy – 83%).

All this implications from B and C are suggested for experts checkout as the questions like: «Is it true that IF <common name objects> <attribute A_{11}> & <attribute A_{12}> & … & <attribute A_{1n}>, THAN <attribute A_{21}> & <attribute A_{22}> & … & <attribute A_{2n}>?». Expert can adopt implication or do not. If the expert can not answer "yes" for this question he must to suggest counter-example from the domain and include it into context, or to correct given context if it has a mistake and reprocess it. The process will be done, when expert accepted all implications.

In the context above implication 7 which is false for any objects is excluded. The implications 1-6 which were apparent are accepted as a rule. Implication 8 is false for one object "*segment*" and implication 9 is false for the object "*half-line*". These implications can be cased as a rule with elimination because the concept «*segment*» is basic for such concepts like «*poly-lines*» and "*Polygon*". "*Half-line*" is basic for such concepts like "*angle*". It is possible to add the predicate - «*x consist of* ». Thus our conceptualization will be more complete.

Testing of the pupils knowledge

Interactive tutoring environment supposes closed iterative cycle of learning which includes such components like: presentation of new material, testing of the pupils knowledge, constructing of his cognitive model, its comparison with a master one, generation of the tutoring strategy.

This software can automatically generate the tests witch can be used for estimation of pupil's knowledge about main concepts and relations between them.

Any question consists of premise and subject. The subject determines the set of possible answers (explicatively and implicatively). The premise adjects the subject with instruction using witch it is necessary to choose the right answer. The formal concepts let us to test the knowing of their attributes. By the concept (*P, G*) we can generate two types of questions:

- For given object *P* to define the set of their attributes *G*;
- For given set of attributes *G* to define the set of objects *P* witch have all properties from *G.*

Beside this it is possible to recompose tests by intersection of concepts intents and/or extents.

- What common properties has objects $P_i \cup P_j$;
- What objects have properties $G_i \cap G_j$.

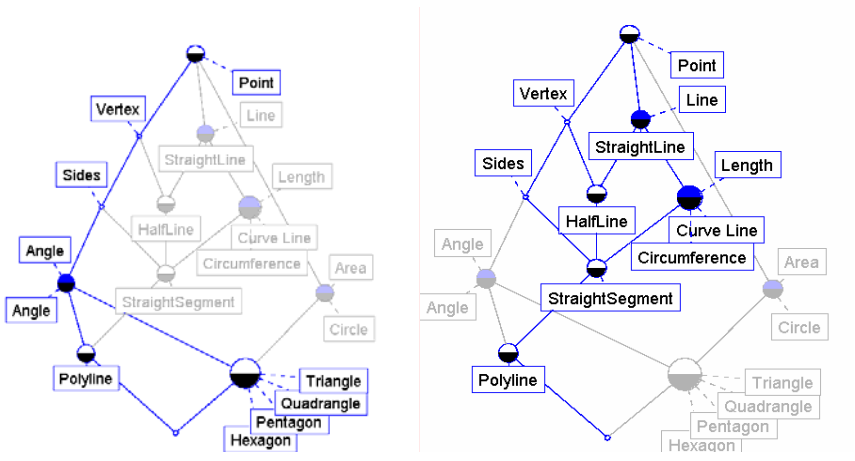
For example, the question can looks like: «*What attributes has circumference*?»; «*What objects has an area*?». The tests like that let us to compose the cognitive model of pupil as the formal context (lattice).

Using implications we can test how the pupil understood the logic and rules of given subject. In the questions composed using implications the rules of the conjunctions elimination are used. For example the question can be like "What *attributes have figures having sides*?" Such questions may be difficult for the pupil. That is why we use not only open form questions but closed form, meaning the questions where one answer from a proposed list must be chosen ('Yes" or "Not"). The exampled question in closed form can be like: "*Is it true that if the object has sides, than it has vertex*?" (answer: «Yes» or «Not»).

The main problem of automatic generation of questions is the choice of destructors i.e. most believable alternatives of answer. Traditionally this problem is solved by the open testing (without the predefined alternatives) and most resent wrong answers become destructors. This solution is quite expensive. That's why we suggest follows.

Let $J(a)$ – ideal, generated by the concept *a*, $a \in L$, where L – initial lattice; $D(a)$ – dual ideal, generated by the concept *a.* It contains all upper elements having path to *a.* For selection of destructors it is necessary to take sequentially concepts $x \in D(a) \setminus \{a\}$ under criteria of minimal distance from *a*, and generate new ideals $J(x)$. The destructors will be objects from all concepts *y,* where $y \in J(x) \setminus J(a)$. For attributes testing we generate new dual ideals $D(x)$ and choose destructors from the set $D(x) \setminus D(x)$.

For example, we can see on fig. 3,a ideal and dual ideal, generated by the concept ({*Angle, Poly-line, Triangle, Tetragon, Pentagon, Hexagon*}, {*has angle, has sides, has vertex*}).



a) *b)*

Fig. 3. Ideals of lattice "Geometry Figures"

The question may be like: «What common properties have *Angle, Poly-line, Triangle, Tetragon, Pentagon, Hexagon?".* To the answer list beside right answers we must add some destructors. Let us construct the ideal generated by the nearest upper concept node (labeled "*sides*"). It includes the object *"segment".* For this node we

construct the dual ideal (fig. 3, b). We see that the object *"segment"* has attributes «*is line*» and «*has length*» which belongs to difference between this dual ideal and previous one. This attributes will be taken as destructors.

The questions in which it is necessary to decide if the sentence is correct suppose the answers: "Yes", "No". Questions of this type can be constructed by implications and concepts as well. The questions of this type contains always true sentence. For the construction of the incorrect sentences we can replace some premises between two implications or some attribute sets between concepts. More complete questions suggest arranging the elements by given criteria, combining elements into the groups or to specify the value of … (volume, weight, etc.) of the listed objects". For this type we supposed to use the many-valued contexts. The quantity attributes are contained directly in the concept. That is why this type is not difficult.

Conclusions

Nowadays computer technologies let us to create intelligent tutoring systems where the knowledge about subject studied meta-knowledge about tutoring control and pupil's knowledge estimation are represented distinctively. Developed technology and software tools using FCA let us not only to visualize concept system of domain but also automate the generating of exercises for acquisition of pupil's knowledge for its testing.

Bibliography

[1] Burmeister P. Formal Concept Analysis with ConImp: Introduction to basic features // G.Stumme, R.Wille (eds.) Begriffiche Wissensverarbeitung: Methoden und Anwendungen. Springen-Verlag. – 1999.

[2] Kuznetsov S.O., Ob'edkov S.A. Algorithms for the construction of the Set of All Concepts and Their Line Diagram, Dresden, 2000.

[3] Stefanuk V. L. Leaning Level Analysis in Intelligent Tutoring Systems // Japan-CIS Symposium on Knowledge-Based Software Engineering, 1994 (JCKBSE'94). Pereslavl-Zalesski, May 10 - 13, 1994, P. 9 -13.

[4] Wille R. Ganter D. Formal Concept Analysis. Springer –Verlag. Berlin. - 1999.

[5] Wille R. Knowledge Acquisition by methods of Formal Concept Analysis // Preprint № 1238. Technische Hochschule Darmstadt. 1989.

[6] Birkgoff G. Lattice theory. Amer. Math. Soc. Providence / R.I. 1967.

[7] Gavrilova T.A., Choroshevski V.F. Knowledge bases of intelligent systems. Sankt-Petersburg. Piter. 2000 (in Russian).

[8] Golitsin G.A., Petrov V.M. Information – behavior – creation. - M.: Nauka, 1991.

[9] Petrushin V.A. Expert – tutoring systems. – Kiev: Naukova dumka, 1992.

[10] Taran T.A., Sirota S.V. Concept tutoring in intelligent tutoring systems based on formal concept analysis. // J. Artificial Intelligence. Donetsk. №3. 2000. (in Russian).

Author information

Tatyana A. Taran - National Technical University of Ukraine "KPI" Applied Mathematic Department Av. Peremogy, 37, 03056, Kiev, Ukraine e-mail: taran@pma.ntu-kpi.kiev.ua

Sergiy V.Sirota - Prosvita publishing Ltd. 46-Shevchenka blwd. Kiev Ukraine Tel/Fax: + 38044 2349523 e-mail: sirota@prosvita.kiev.ua

KNOWLEDGE ACQUISITION AND USING IN A PATTERN RECOGNITION SYSTEM

A. D. Zakrevskij

***Abstract**: A common logical approach is proposed to solve the problems of data mining and pattern recognition in finite spaces of Boolean or multi-valued attributes. It is based on a special form of knowledge representation, called implicative regularities, and combines two powerful tools of modern logic: the inductive inference and the deductive inference. The first one is used for extracting the knowledge from the data. The second is applied when the knowledge is used for calculation of the goal attribute values. A set of efficient algorithms was developed for that, dealing with Boolean functions and finite predicates represented by logical vectors and matrices.*

Introduction

Knowledge is the central concept in a wide variety of investigations dedicated to pattern recognition problems [1, 3, 6]. Solving them begins with the choice of a proper "world model" - an abstract artificial world reflecting some important qualities of real subject areas – important from the point of view of the problems to be solved.

In this paper, we use a model which defines this abstract world as a set W of logical n-vectors presenting values of attributes composing the set $X = \{x_1, x_2, ..., x_n\}$. The attributes could be binary (Boolean) or multi-valued. In the latter case each of the attributes x_i is characterized by a corresponding finite set V_i of alternative values. The Cartesian product of these sets $V_1 \times V_2 \times ... \times V_n$ (or $\{0, 1\}^n$ in the binary case) constitutes the space of attributes M. Elements from W can be regarded as abstract models of real objects of a natural subject area. The world as a whole is represented by a relation $W \subseteq M$ or by the corresponding finite predicate $\varphi(x_1, x_2, ..., x_n)$ which takes value 1 on the elements of the set W. In case of two-valued attributes this predicate is a Boolean function $f(x_1, x_2, ..., x_n)$. That approach is rather simple, inasmuch the world is considered only as the set of its elements, but it is sufficient for solving many practical problems.

Usually, only partial information about the world W is known, represented in terms of data and knowledge. Suppose that the data present information concerning some separate elements from W, describing these elements by corresponding logical vectors. Taken together, these vectors represent a so called sampling population (a reliable selection from the subject area) and constitute the set F serving further as a data base. As a rule, $|W| << |M|$ and $|F| << |W|$. The knowledge, on the contrary, presents information about qualities of the whole subject area, expressed by some inherent in the regarded subject area regularities which establish ties between attributes.

The pattern recognition process, taken as a whole, may be roughly divided into two main stages: obtaining some knowledge by data mining and predicting values of goal attributes by using this knowledge. The methods of inductive and deductive inference are applied at these stages, accordingly. Their efficiency depends greatly on the form in which the knowledge is presented. A special attention is paid below to this point.

The Knowledge – Concept and Format

Within the framework of our world model, the knowledge is defined as a set of regularities. The key question is to choose a proper model for them. Starting from general assumptions it is accepted that any regarded regularity defines a logical connection between some attributes: it means that some combinations of attribute values are declared impossible (prohibited).

Evidently, the less attributes are connected by some regularity, the stronger is the latter. That is confirmed, for instance, by the long history of investigations in physics and other nature sciences. On the other hand, if we choose several attributes and decide to connect them by one regularity, it will be the weakest when it forbids only one combination of those attributes values.

In the Boolean case such a regularity can be expressed by the logical equation $k_i = 0$ or by the equivalent to it equation $d_i = 1$, where k_i is a conjunct formed of some attributes (in direct or inverse mode) from the set X, d_i is a

disjunct, and $d_i = \neg k_i$. For instance, equations $ab'c = 0$ and $a' \vee b \vee c' = 1$ (' is the symbol of inversion, equivalent to $\neg$) represent the same regularity, which prohibits the combination $a = 1$, $b = 0$, $c = 1$.

A regularity of this kind is called *implicative* (more general than functional one) [5]. It prohibits a set of attribute value combinations forming an interval in the Boolean space M over X - the characteristic set of the conjunct k_i. The size of that interval (the number of its elements – they all are prohibited) equals 2^{n-r}, where n is the number of all attributes and r is the rank of the implicative regularity – the number of attributes coming into it. It becomes clear now how the strength of the regularity is defined by its rank.

Suppose that $X = \{a, b, c, d, e, f\}$ and consider the implicative regularity $ab'e = 0$ forbidding the combination 101 of values of the attributes $a, b, e,$ accordingly. The corresponding empty interval of the space M contains eight elements: 100010, 100011, 100110, 100111, 101010, 101011, 101110 and 101111. The equation $ab'e = 0$ may be changed for the equivalent equation $ab'e \rightarrow 0$ with the implication operator $\rightarrow$ (if… then…), known as a *sequent* (its left part is always a conjunction, and the right part – a disjunction). The latter equation may be subjected to equivalence transformations consisting in transferring arbitrary literals between the left part (conjunction) and the right one (disjunction), changing each time their type (positive for negative or *vice versa*). In such a way we could obtain the following set of the equivalent equations $ae \rightarrow b$ (if $a = 1$ and $e = 1$, then $b = 1$), $ab' \rightarrow e'$, $a \rightarrow b \vee e'$, …, $1 \rightarrow a' \vee b \vee e'$. The last one could be changed for the disjunctive equation $a' \vee b \vee e' = 1$.

A set of regularities given in such a form can be presented by a ternary disjunctive matrix ***D***, called below a *knowledge* matrix. For example, the knowledge matrix

	a	*b*	*c*	*d*	*e*	*f*	*g*	*h*
	1	–	–	0	–	–	0	–
D =	–	–	–	1	–	1	–	–
	0	1	–	–	–	–	–	–

affirms that every object of the regarded area must satisfy the equations

$$a \vee d' \vee g' = 1, \quad d \vee f = 1 \text{ and } a' \vee b = 1.$$

In other words, in the considered Boolean space there exists no object which has any of the following combinations of values of some attributes: ($a = 0$, $d = 1$, $g = 1$), ($d = 0$, $f = 0$) and ($a = 1$, $b = 0$). The set of these equations can be reduced to one equation $D = 1$ where D is a CNF (conjunctive normal form) represented by the matrix ***D***.

$$D = (a \vee d' \vee g')\,(d \vee f)\,(a' \vee b) = 1.$$

By inverting both left and right parts of the equation $D = 1$ we get the equivalent equation $C = 0$ with the left part $C = \neg$ D called a *veto function*: it defines the prohibition area. For the regarded example

$$C = a'dg \vee d'f' \vee ab' = 0.$$

The suggested form of implicative regularities turned out to be extremely convenient on the stage of deductive inference, where the methods developed for theorem proving automation are successfully applied [2]. As it is shown below, regularities of the considered type could be rather easily discovered in the data base, and it is not difficult to evaluate their strength and plausibility, which is very important for their further application.

In the case of finite predicates generalized conjuncts and disjuncts could be used to present the knowledge [7, 8]. Any interval in the space of multi-valued attributes is defined as a direct product of non-empty subsets α_i taken by one from each set V_i. Its characteristic function is defined as a conjunct, and the negation of the latter is a disjunct.

Suppose $X = \{x, y, z\}$, and the attributes x, y, z select their values from the corresponding sets $V_1 = \{a, b, c\}$, $V_2 = \{a, e, f, g\}$, $V_3 = \{h, i\}$ (note that these sets may intersect). Let $\alpha_1 = \{a\}$, $\alpha_2 = \{a, e, g\}$, $\alpha_3 = \{h, i\}$. The interval $I = \alpha_1 \times \alpha_2 \times \alpha_3$ presented by the vector 100.1101.11 has the characteristic function (conjunct)

$$k = (x = a) \wedge ((y = a) \vee (y = e) \vee (y = g)) \wedge ((z = h) \vee (z = i)),$$

which could be simplified to

$$k = (x = a) \wedge ((y = a) \vee (y = e) \vee (y = g)),$$

inasmuch as $(z = h) \vee (z = i) = 1$. If this product enters the equation $k = 0$ which reflects a regular connection between x and y, then $I \cap W = \varnothing$, i. e. the interval I turns out to be empty.

As it can be seen from the given example, the structure of a conjunctive term in the finite predicate algebra is more intricate compared with that of the binary case - the two-stage form of the type $\wedge \vee$ is inherent in it. One can avoid that complexity changing the equation $k = 0$ for the equivalent equation $\neg k = 1$ and transforming $\neg k$ into a one-stage disjunctive term d. Such transformation is based on de-Morgan rule and changes expressions $\neg (x_i \in \alpha_i)$ for equivalent expressions $x_i \in V_i \backslash \alpha_i$. This is possible since all sets V_i are finite.

For the considered example

$d = \neg k = (x \neq a) \vee ((y \neq a) \wedge (y \neq e) \wedge (y \neq g)) = (x = b) \vee (x = c) \vee (y = f)$.

Adhering to the tradition, let us call similar expressions as disjuncts. Suppose that the knowledge obtained either from experts or by induction from the data is represented by a set of disjuncts $d_1, d_2, ..., d_m$. Generated by them, equations $d_i = 1$ are interpreted as conditions which should be satisfied for any objects of the world, and it is possible to reduce them (equations) to a single equation $D = 1$ the left part of which is presented in the conjunctive normal form - CNF $D = d_1 \wedge d_2 \wedge ... \wedge d_m$. It follows from here that in the finite predicate algebra the CNF has some advantage over the disjunctive normal form - DNF $K = k_1 \vee k_2 \vee ... \vee k_m$ which is used in the equivalent equation $K = 0$. Indeed, DNF has three stages $(\vee \wedge \vee)$, whereas CNF - only two $(\wedge \vee)$.

In the case of multi-valued attributes, it is more convenient to use sectional Boolean vectors and matrices introduced for representation of finite predicates [7]. A sectional Boolean vector consists of some sections (domains) corresponding to attributes and each section has several binary digits corresponding to the attribute values indicating definite properties. For example, the section corresponding to the attribute *color*, which has the values *blue*, *red*, *green*, *yellow*, *brown*, *black* and *white*, should have 7 bits. For the example given above, the vector 010.1000.01 describes an object with the value b of the attribute x, the value a of the attribute y and the value i of the attribute z. Obviously, if a vector represents some element of the space M of multi-valued attributes, it has the only 1 in each section. The situation is different in the case of some fuzziness. The vector 011.1001.01 can be interpreted as presenting a partial information about the object, when we know only that $x \neq a$, $y \neq e$, $y \neq f$ and $z \neq h$. Note, that each of these inequalities serves as an *information quanta* and is marked by a zero in the corresponding component of the vector.

Giving an example of presenting the knowledge, suppose that $X = \{a, b, c\}$, $V_1 = \{1, 2, 3\}$, $V_2 = \{1, 2, 3, 4\}$ and $V_3 = \{1, 2\}$. Then the knowledge matrix

$$\boldsymbol{D} = \begin{array}{ccc} a & b & c \\ 0\,0\,1 & 0\,0\,1\,0 & 0\,0 \\ 1\,1\,0 & 0\,0\,1\,1 & 0\,1 \\ 0\,1\,0 & 1\,1\,0\,0 & 1\,0 \\ 0\,0\,1 & 0\,1\,0\,0 & 0\,1 \end{array}$$

may be interpreted as a set of disjunctive equations

$(a = 3) \vee (b = 3) = 1$,
$(a = 1) \vee (a = 2) \vee (b = 3) \vee (b = 4) \vee (c = 2) = 1$,
$(a = 2) \vee (b = 1) \vee (b = 2) \vee (c = 1) = 1$,
$(a = 3) \vee (b = 2) \vee (c = 2) = 1$

or as one equation with a CNF in the left part:

$((a = 3) \vee (b = 3)) \wedge ((a = 1) \vee (a = 2) \vee (b = 3) \vee (b = 4) \vee (c = 2)) \wedge \quad \wedge ((a = 2) \vee (b = 1) \vee (b = 2) \vee (c = 1)) \wedge ((a = 3) \vee (b = 2) \vee (c = 2)) = 1$.

Data Mining

A very important part of the pattern recognition problem is obtaining knowledge from data [3]. The data could be represented by a sampling population F - a set of some randomly selected elements from the regarded world W.

As it was formulated above, we solve that problem by analyzing the distribution of elements of the set F in the space M (suppose it is Boolean) and revealing implicative regularities which are reflected by empty intervals (not intersecting with F). That operation can be reduced to observing a Boolean data matrix $\boldsymbol{K}$ and looking for such combinations of attribute values which do not occur there.

The number of attributes coming into an implicative regularity is called its rank. It coincides with the rank of the corresponding interval. Remind that the less attributes are tied with a regularity, the stronger is the tie. So, it is worthwhile to look for regularities of smaller rank.

Consider, for example, the following data matrix ***K***:

a	*b*	*c*	*d*	*e*	*f*
1	0	0	1	1	0
0	1	1	1	0	0
1	1	0	1	0	1
0	0	0	1	1	0
0	1	0	1	1	0
0	0	1	0	1	0
1	1	1	1	0	0
1	0	0	0	1	1

There are no empty intervals of the rank 1, because each column contains 1s and 0s. So we look further for empty intervals of the rank 2 and find five of them, corresponding to the following combinations: (a = 0, f = 1), (b = 1, d = 0), (b = 0, e = 0), (c = 1, f = 1), (d = 0, e = 0). In a more compact form these intervals may be represented by conjuncts *a'f*, *bd'*, *b'e'*, *cf*, *d'e'*. Can we consider that these found empty intervals reflect real regularities inherent in the world from which the data were extracted? Such conclusions could be accepted if only they are plausible enough.

Consider the general case of n binary attributes and m elements in the sampling population (selection) F. Suppose, we have found an empty interval of the rank r (comprising 2^{n-r} elements of the Boolean space M and put forward the corresponding hypothesis, affirming that this interval is free of any elements from the regarded world W. May we rely on it and make with its help some logical conclusions when recognizing an object with the unknown value of the goal attribute? The problem is to estimate the plausibility of that hypothesis.

We should take into account that the regarded interval could be empty quite accidentally, while in reality the selection F is taken by random from the whole space M - in that case there could be no regularities in the disposition of the elements from F in M.

It would be useful to express the probability p of such an event as a function $p(n, m, r)$ of the parameters n, m, r. The hypothesis can be accepted and used further in procedures of deductive inference if only this probability is small enough. Its calculation is rather difficult, so it was proposed in [5] to approximate it by the mathematical expectation $E(n, m, r)$ of the number of empty intervals of the rank r.

That value can be calculated by the formula

$$E(n, m, r) = C_n^r \, 2^r \, (1-2^{-r})^m ,$$

where C_n^r is the number of r-element subsets of an n-element set, $C_n^r \, 2^r$ is the number of intervals of the rank r in the space M, and $(1-2^{-r})^m$ is the probability of some concrete interval of the rank r to be empty, not containing any elements from F.

Some empty intervals could intersect, hence $E(n, m, r) \geq p(n, m, r)$. The question is how big could be the difference $E(n, m, r) - p(n, m, r)$? It was shown, that it becomes negligible small for small values of $E(n, m, r)$. But that is just the case of interest for us.

It turns out that the value of the function $E(n, m, r)$ grows very rapidly with rising r. That is evident from the Table 1 of the dependence E on r under fixed values of other parameters: n = 100 and m = 200.

Table 1. The dependence E on r under fixed n and m

r	1	2	3	4	5	6
$E(100, 200, r)$	1.24×10^{-58}	2.04×10^{-21}	3.26×10^{-6}	1.56×10^{2}	4.21×10^{6}	3.27×10^{9}

It is clear that the search for empty intervals and putting forward corresponding hypotheses can be restricted in this case by the relation $r < 4$. If some empty interval of the rank $r < 4$ is found, we can formulate the corresponding regularity with good reason, but there are no grounds for that if $r \geq 4$. So, when n = 100 and

m = 200, there is no sense in looking for empty intervals of the ranks more then 3. The search for regularities could be strongly restricted in that case by checking for emptiness only intervals of the rank 3, which number is $C_{100}^3 \times 2^3$ = 1,293,600. Not much, compared with the number 3^{100} of all intervals in the Boolean space of 100 variables, approximately 5.15×10^{47}.

A threshold ω may be introduced to decide whether it is reasonable to regard an empty interval as presenting some regularity: the positive answer should be given when $E < \omega$. Its choice depends on the kind of problems to be solved on the base of found regularities.

Suppose ω = 0.01. Then the maximum rank r_{max} of intervals which should be analyzed when looking for regularities could be found from Table 2, showing its dependence on n and m.

Table 2. The dependence of the maximum rank r_{max} on parameters n and m

$n \setminus m$	20	50	100	200	500	1000
10	1	2	3	4	5	6
30	1	2	2	3	4	5
100	1	1	2	3	4	5

Two conclusions, justified for the regarded range of parameters, could follow from this table. First, in order to increase r_{max} by one it is necessary to double the size of the experiment – the number m of elements in F. Second, approximately the same result could be achieved by reducing by a factor of 10 the number of attributes used for the description of the regarded objects.

Suppose r_{max} = 2 which is enough when the selection F is rather small. In that case we have to pay attention only to pairs of attributes, looking for some forbidden combinations of their values. This task can be executed by an incremental algorithm. It analyzes the elements of the selection F consecutively, one by one, and fix such two-element combinations which have occur, using a symmetrical square Boolean $2n \times 2n$ matrix $\boldsymbol{S}$ for that, with rows and columns corresponding to the values $x_1 = 0$, $x_1 = 1$, $x_2 = 0$, $x_2 = 1$, etc. Its elements corresponding to occurring combinations are marked with 1. The rest combinations (not occurring) are presented by zero (empty) elements and accepted as forbidden. The regularities presented by them connect some attributes in pairs and are called syllogistic [6]. For example, regarding the following selection F (only to illustrate the algorithm, despite the fact that the selection is too small for r_{max} = 2):

a	*b*	*c*	*d*	*e*	
0	1	0	0	1	1
1	1	0	1	1	2
1	0	0	1	1	3
0	1	1	0	0	4
1	0	0	1	1	5
0	1	1	0	0	6

we shall find in the end ten two-element combinations which do not occur in F, and consider them as syllogistic regularities. They can be presented by the following ternary knowledge matrix $\boldsymbol{D}$:

	a	*b*	*c*	*d*	*e*
	0	0	–	–	–
	0	–	–	1	–
	1	–	1	–	–
	1	–	–	–	0
$\boldsymbol{D}$ =	–	0	1	–	–
	–	0	–	0	–
	–	0	–	–	0
	–	–	1	1	–
	–	–	1	–	1
	–	–	–	1	0

When the selection F is noticeably bigger compared with the number of attributes, the maximum rank r_{max} of implicative regularities could be 3, 4 or even more. The run-time for their finding swiftly increases. Nevertheless it is restricted, because the number of intervals to be checked could be approximated by $C_n^3\, 2^3$, $C_n^4\, 2^4$, etc.

It is a little more difficult to extract knowledge from the space of n multi-valued attributes $x_1, x_2, \ldots, x_n$ [9, 11]. To begin with, define the probability p that some concrete disjunct will be satisfied by an accidentally chosen element of the space. It could be calculated by the formula

$$p = 1 - \prod_{i=1}^{n} (r_i/s_i),$$

where s_i is the number of all values of the attribute x_i, and r_i – the number of those of them which do not enter this disjunct. For instance, $p = 1 - 2/2 \times 3/4 \times 1/3 = 3/4$ for the disjunct 00.1000.101. Let us divide all disjuncts into classes D_j, forming them from disjuncts with the same value of p. And let us number these classes in order of increasing p and introduce the following conventional signs: q_j – the number of disjuncts in the class D_j, p_j – the value of p for elements from D_j.

Find now the mathematical expectation E_j of the number of disjuncts from the class D_j, which do not contradict to the random m-element selection from the regarded space:

$$E_j = q_j (p_j)^m,$$

and introduce the analogous quantity E_k^+ for the union of classes $D_1, D_2, \ldots, D_k$:

$$E_k^+ = \sum_{i=1}^{k} E_j.$$

Inductive inference is performed by consecutive regarding classes D_j in order of their numbers and summarizing corresponding values E_j until the sum surpasses a threshold t, which is introduced with taking into account the specific of the problems to be solved. All disjuncts belonging to these classes are accepted as regularities if they do not contradict the data, i. e. if they are satisfied by any element of the selection F.

The expert may fix several thresholds and assign accordingly different levels of plausibility to the found regularities. For example, regularities obtained by thresholds 10^{-10}, 10^{-6}, 10^{-3} could be estimated as *absolutely plausible*, *usually*, *most likely*. This differentiation gives some flexibility to recognition procedures. Choosing a proper level of plausibility one can use only some of regularities contained in the knowledge base and vary in such a way the plausibility of the logical conclusions obtained during recognition. For example, using only the most plausible regularities can result in obtaining a little number of logical conclusions, but more reliable ones, while extending the used part of the knowledge base extends also the set of obtained logical conclusions, at the expense of their plausibility.

We do not regard here the important problem (touched in [12]) of extracting knowledge from partial data - when values of some attributes of the elements from F remain unknown.

Solvind Equations of Deductive Inference

The recognition problem can be regarded as the problem of a closer definition of qualities of some observed object not belonging to the experimental selection from the subject area [5, 14]. It is formulated in terms of logical equations, Boolean or predicate, and the tree searching technique of deductive inference is applied for their solution [4, 10, 13].

Suppose, we know the values of s from n attributes of this object. That is equivalent to location of the object in a certain interval of the Boolean space M presented by the corresponding elementary conjunction k of the rank s. The problem is to define by logical reasoning, as sure as possible, the values of the remaining $n - s$ attributes, using for that the information contained in the knowledge ternary matrix $\boldsymbol{D}$ and in the corresponding veto function V.

Let us regard the set X_k of attributes with known values and the set of all forbidden combinations of values of the rest attributes − for the considered object. The latter set can be described by a proper Boolean veto function $V(k)$ that could be easily obtained from V. Indeed, it is sufficient for that to transform the formula representing the function V by changing symbols of attributes presented in k for values (0 or 1) satisfying the equation $k = 1$. Denote this operation as $V(k) = V{:}k$.

Suppose that we want to know the value of an attribute x_i which does not come into X_k. The necessary and sufficient condition for the prohibition of the value 1 of that attribute is presented by the formal implication $kx_i \Rightarrow$

V, i. e. belonging the interval presented by the conjunction kx_i to the prohibition region described by the function V. Analogously, the necessary and sufficient condition for the prohibition of the value 0 is presented by $kx_i' \Rightarrow V$.

It is not difficult to deduce from here forecasting rules to define the value of the goal attribute x_i of the object characterized by k. These rules are shown in a compressed form in Table 3 presenting the decision (a set of possible values of x_i - the bottom row) as a function of predicates $kx_i \Rightarrow V$ and $kx_i' \Rightarrow V$.

Table 3. Forecasting the value of the attribute x_I

$kx_i \Rightarrow V$	0	0	1	1
$kx_i' \Rightarrow V$	0	1	0	1
x_i	{0, 1}	{1}	{0}	∅

Note that four outcomes could appear at this approach. On a level with finding the only value (0 or 1) for the attribute x_i, such situations could be met when both values are acceptable or neither of them satisfies the veto function V. At the last case the existence of the object α characterized by k contradicts the knowledge base, and that could stimulate some correction of the latter. However, the probability of such an event is low enough, taking into account the way of forming the knowledge base.

For example, if

$$V = acf' \vee be'f \vee a'd'e \vee b'df \vee b'c'd'$$

and $k = abf$, then $V(k) = V{:}abf = e'$. It could be concluded from this that the regarded object α has value 1 of attribute e, but there are no restrictions on other attributes (c and d). If by the same function V the object α is characterized by $k = c'e'f$, then

$$V(k) = b \vee b'd \vee b'd' = 1 \text{ (all is forbidden)},$$

and that means that the object contradicts the knowledge.

The predicates $kx_i \Rightarrow V$ and $kx_i' \Rightarrow V$ are accordingly equivalent to the predicates $V{:}kx_i = 1$ and $V{:}kx_i' = 1$, and that allows us to reduce their calculation to checking corresponding submatrices of the knowledge matrix $\boldsymbol{D}$ for consistency. Fixing values of some attributes in the function V is changed for selecting the corresponding minor of the matrix $\boldsymbol{D}$ by deleting some rows and columns, which could be followed by further possible simplification.

Suppose, we regard the same (already minimized) knowledge matrix $\boldsymbol{D}$ corresponding to the veto function $V = acf' \vee be'f \vee a'd'e \vee b'df \vee b'c'd'$ and know that for the observed object $a = 1$ and $c = 1$. Taking into account this new information we transform the matrix $\boldsymbol{D}$. We delete from it the columns marked with a and c because these variables became constant, and delete also the rows 3 and 5 now satisfied by these constants. Further simplification is rather evident, using the following rule: $x\,(x' \vee H) = x\,H$, where x is a Boolean variable and H – an arbitrary Boolean formula.

$$\boldsymbol{D}^* = \begin{array}{cccccc|c} a & b & c & d & e & f & \\ 0 & - & 0 & - & - & 1 & 1 \\ - & 0 & - & - & 1 & 0 & 2 \\ 1 & - & - & 1 & 0 & - & 3 \\ - & 1 & - & 0 & - & 0 & 4 \\ - & 1 & 1 & 1 & - & - & 5 \end{array} \qquad \begin{array}{cccc} b & d & e & f \\ - & - & - & 1 \\ 0 & - & 1 & 0 \\ 1 & 0 & - & 0 \end{array} \qquad \begin{array}{cccc} b & d & e & f \\ - & - & - & 1 \\ 0 & - & 1 & - \\ 1 & 0 & - & - \end{array}$$

We can conclude now that $f = 1$, by necessity. As to the remaining attributes, their values cannot be forecasted uniquely. They obey the next two conditions: $b' \vee e = 1$ and $b \vee d' = 1$. This system of logical equations has two solutions. Either $b = d = 0$ (with an arbitrary value of e), or $b = e = 1$ (with an arbitrary value of d).

Conclusion

Implicative regularities were proposed to fix the knowledge extracted from data on the stage of inductive inference and to play the role of axioms on the stage of deductive inference, when the values of goal attributes should be forecasted. A set of algorithms was developed to implement those operations. The suggested means were used when constructing several expert systems of various purposes where the pattern recognition problem was the central one. The computer experiments testified the high efficiency of the proposed approach.

References

1. Bongard M. Pattern recognition. – Spartan Books, New York, 1970.
2. Chang C. L., Lee R. C. T. Symbolic logic and mechanical theorem proving. - Academic Press, New York - San Francisco - London, 1973.
3. Frawley W. J., Piatetsky-Shapiro G., and Matheus C. J. Knowledge discovery in data bases: an overview. - In: Knowledge discovery in data bases (ed. by Piatetsky-Shapiro and Frawley), Cambridge, Mass: AAAI/MIT Press, 1991, pp. 1-27.
4. Nilsson N. J. Problem-solving methods in artificial intelligence. – McGraw-Hill Book Company, New York, 1971.
5. Zakrevskij A. D. Revealing of implicative regularities in the Boolean space of attributes and pattern recognition. - Kibernetika, No 1, 1982, pp. 1-6 (in Russian).
6. Zakrevskij A. D. Logic of recognition. - Minsk: Nauka i tekhnika, 1988 (in Russian).
7. Zakrevskij A. D. Matrix formalism for logical inference in finite predicates. - Philosophical bases of non-classical logics. Institute of Philosophy AN SSSR, Moscow, 1990, pp. 70-80 (in Russian).
8. Zakrevskij A. D., Levchenko V. I., Pechersky Yu. N. Instrumental expert system EDIP, based on finite predicates. - Applied systems of artificial intelligence. Mathematical researches, issue 123, Inst. Math. AN Mold. SSR, Kishinev, 1991, pp. 24-40 (in Russian).
9. Zakrevskij A. D. EXSYLOR - expert system for logical recognition. - Upravlyayushchie sistemy i mashiny, 1992, No 5/6, pp. 118-124 (in Russian).
10. Zakrevsky A. D. Logical recognition by deductive inference based on finite predicates. - Proceedings of the Second Electro-technical and Computer Science Conference ERK'93, Slovenia Section IEEE, Ljubljana,1993, v.B, pp.197-200.
11. Zakrevsky A. Logical recognition in the space of multivalued attributes. - Computer Science Journal of Moldova, 1994, v. 2, No 2, pp. 169-184.
12. Zakrevskij A. D., Vasylkova I. V. Inductive inference in systems of logical recognition in case of partial data. - Proceedings of the Fourth International Conference on Pattern Recognition and Information Processing, Minsk-Szczecin, May 1997, v. 1, pp. 322-326.
13. Zakrevskij A. D. Pattern recognition as solving logical equations. – Special Issue 1999 - SSIT'99 (AMSE), pp. 125-136.
14. Zakrevskij A. D. A logical approach to the pattern recognition problem. – Proceedings of the International Conference KDS-2001 "Knowledge - Dialog - Solution", S.- Peterburg, June 2001, v. 1, pp. 238-245.

Author information

Arkadij Zakrevskij - The United institute of informatics problems of the NAS of Belarus;
Surganov Str. 6, 220012 Minsk, Belarus; e-mail: zakr@newman.bas-net.by

Section 6: Logical Inference

К ЛОГИЧЕСКОМУ ВЫВОДУ НА ОСНОВЕ НЕЧЕТКОЙ ИМПЛИКАЦИИ

Г. Бакан, О. Кононенко

Аннотация: *На основе определения нечеткого множества как подграфика функции принадлежности введены нечеткие отношения для логических операций. Для нечеткой импликации предложенная процедура вывода при частичном выполнении ссылки.*

Ключевые слова*: нечеткое множество, функция принадлежности, нечеткая логическая операция, нечеткое отношение.*

Введение

Одной из актуальных проблем в теории экспертных систем есть проблема учета неполноты или неточности информации, которая помещается в данных и знаниях. Неопределенность так или иначе присутствует во многих практических задачах, которые решаются экспертными системами.

Один из хорошо известных и довольно простых подходов для получения вывода на основе неточных данных и ненадежных правил был реализован в системе MYCIN [1]. Здесь как посылка правил, так и сами правила снабжались коэффициентами или показателями неопределенности.

Эти показатели указывали на степень истинности данных и правил. На основе простых эмпирических формул можно получить показатели определенности результирующих выводов.

Более традиционным и в то же время теоретически обоснованным подходом к решению проблемы является вероятностный подход. В частности, наибольшее распространение получил метод логического вывода в условиях неопределенности, основанный на классической формуле Байеса [2]. Суть метода состоит в редукции априорной вероятности некоторой гипотезы к апостериорной вероятности заключения. Для этого привлекаются дополнительные аргументы, с помощью которых формируется некоторая условная вероятность, которая выполняет роль функции правдоподобия.

Если теоретическая безупречность метода относится к его важным преимуществам, то к недостаткам относятся прежде всего трудности в определении априорных вероятностей, а также необоснованность обычно используемой гипотезы о независимости факторов, связанных с решаемой задачей.

В последнее время все большее распространение получает подход, который использует идею нечеткости [3]. Сюда относится нечеткая логика, теория возможностей и, наконец, теория нечетких множеств. В результате сформировалась теория нечетких выводов. В ее основе нечеткая интерпретация основных логических операций, с использованием вместе логических лингвистических переменных и связанных с ними функций принадлежности нечетких множеств.

Нечеткие логические операции.

Пусть $\mu_A(\bullet): D_A \to R^1$ - функция принадлежности нечеткого множества A. Нечеткое множество A будем отождествлять с подграфиком этой функции, а именно:

$$A = \{(x,\alpha): \mu_A(x) \geq \alpha, \alpha \in I, x \in D_A\}, \quad (1)$$

где $D_A \subset X \equiv R^n$, $I \subset R^1$ интервал [0,1].

Здесь для определенности будем считать, что D_A – компактное и выпуклое множество.

При условии (1) каждой точке $x \in D_A$ будет отвечать интервал $[0, \mu(x)]$, а все множество A может быть представлена в виде

$$A = \bigcup_{x \in D_A} \{x\} \times [0, \mu(x)] \subset X \times R^1 .$$

Удобство этого обобщения понятия нечеткого множества состоит в том, что с подграфиками можно делать обычные теоретико-множественные операции, получая при этом как следствие соответствующие функции принадлежности. Например, пусть $B \subset X \times R^1$ нечеткое множество-подграфик функции принадлежности $\mu_B(x)$, которая заданна на выпуклом компакте $D_B \subset X$. Пусть $D_A \cap D_B \neq \varnothing$, тогда для пересечения $A \cap B$ можно записать:

$$A \cap B = \bigcup_{x \in D_A \cap D_B} \{x\} \times ([0, \mu_A(x)] \cap [0, \mu_B(x)]) = \{(x,\alpha): \mu_{A\cap B}(x) \geq \alpha, \alpha \in [0, \alpha_1], x \in D_A \cap D_B\} ,$$

где $\mu_{A\cap B}(x) = \min_x \{\mu_A(x), \mu_B(x)\}$, $\alpha_1 = \max_x \mu_{A\cap B}(x)$.

Здесь и дальше, где это приводит к неоднозначности области изменения аргумента x функции принадлежности будут опускаться.

По известному правилу [3] множество $A \cap B$ можно нормализовать. Тогда получим $A \cap B = \{(x,\alpha): \mu_{A\cap B}(x) \geq \alpha \cdot \alpha_1, \alpha \in I\} \subset X \times R^1$.

Аналогичным образом можно определить функции принадлежности и для других множеств, которые выступают в роли результатов тех или других теоретико-множественных операций.

Вытекая с [4], введем к рассмотрению нечеткие логические операции (конъюнкцию, дизъюнкцию, отрицание, импликацию и эквиваленцию). Их определим с помощью соответствующих теоретико-множественных операций, а именно,

$$A \wedge B \equiv A \cap B,$$
$$A \vee B \equiv A \cup B,$$
$$\neg A \equiv \overline{A},$$
$$A \to B \equiv \overline{A} \cup B,$$
$$A \leftrightarrow B \equiv (\overline{A} \cup B) \cap (\overline{B} \cup A).$$

С использованием функций принадлежности для введенных здесь нечетких логических операций можно записать

$$A \wedge B = \{(x,\alpha): \mu_A(x) \wedge \mu_B(x) \geq \alpha \cdot \alpha_1, \alpha \in I\} ,$$
$$A \vee B = \{(x,\alpha): \mu_A(x) \vee \mu_B(x) \geq \alpha, \alpha \in I\} ,$$
$$\neg A = \{(x,\alpha): \overline{\mu}_A(x) \equiv 1 - \mu_A(x) \geq \alpha, \alpha \in I\} \text{ и т.п.}$$

Здесь

$$\mu_{A\wedge B}(x) \equiv \mu_A(x) \wedge \mu_B(x) = \min_x \{\mu_A(x), \mu_B(x)\},$$
$$\mu_{A\vee B}(x) \equiv \mu_A(x) \vee \mu_B(x) = \max_x \{\mu_A(x), \mu_B(x)\} .$$

С помощью нечетких логических операций можно формализовать высказывания или утверждение естественным языком, которые содержат лингвистические переменные. Например, высказывание:

"ЕСЛИ прибыль предприятия большая, ТО оно имеет хороший рейтинг".

Здесь принимают участие две лингвистические переменные "большая прибыль" и "хороший рейтинг". Предполагается, что они характеризуются соответствующими нечеткими множествами, например, A и B. В результате на формальном уровне данному высказыванию можно поставить в соответствие нечеткую импликацию $A \to B$.

Нечеткие логические отношения

Если множества A и B заданы в разных пространствах (как в приведенному выше примере), тогда с помощью нечетких логических операций можно ввести нечеткие логические отношения.

Итак, пусть $(*)$- некоторая нечеткая бинарная логическая операция. Пусть, далее $A \subset X \times R^1$ и $B \subset Y \times R^1$ нечеткие множества с функциями принадлежности $\mu_A(x)$ та $\mu_B(y)$, которые определенны на $D_A \subset X$ и $D_B \subset Y$ соответственно. Нечетким логическим отношением, которое соответствует операции $(*)$ будем называть нечеткое множество-отношение

$$R_{A*B} = \{(x, y, \alpha) : \mu_{A*B}(x, y) \geq \alpha, \alpha \in I, (x, y) \in D_A \times D_B\},$$

где $\mu_{A*B}(x, y) = \mu_A(x) * \mu_B(y)$.

В частности, для нечеткого отношения-импликации имеем:

$$\mu_{A \to B}(x, y) = \mu_{B|A}(x, y) = \overline{\mu_A(x)} \wedge \mu_B(y), \quad (2)$$

где $\overline{\mu_A(x)} \wedge \mu_B(y) = \min_{x,y} \{\overline{\mu_A(x)}, \mu_B(y)\}$.

Функции принадлежности других отношений определяются аналогично.

На рис.1 представлено нечеткое отношение, которое задается нечеткой логической операцией дизъюнкцией. Здесь нечеткие множества A и B заданные функциями принадлежности треугольной формы. Жирными линиями показаны контуры графика $\mu_{A \vee B}(x, y)$ отношения $A \vee B \subset X \times Y \times R^1$.

На рис.2 для тех же начальных данных показано логическое отношение конъюнкции.

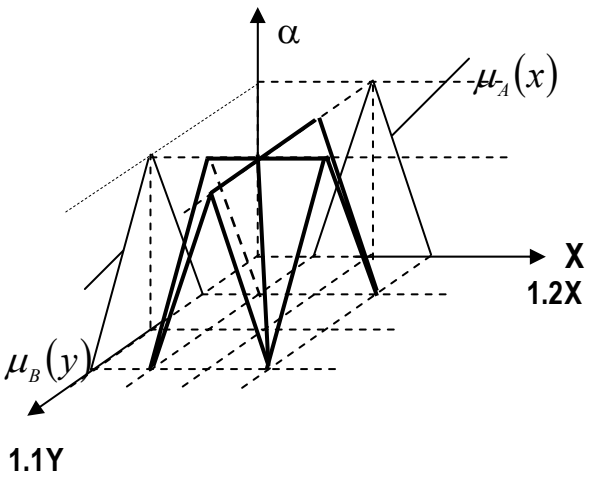


рис.1

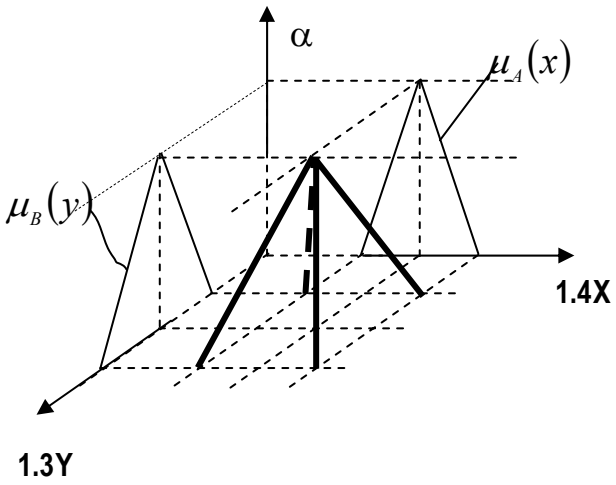


рис.2

Проекции нечетких логических отношений

Пусть $R_{A*B} \subset X \times Y \times R^1$ нечеткое логическое отношение. Введем сечение этого отношения по фиксированному элементу $x \in X$. По определению сечение можно записать

$$R_{A*B}(x) = \{(y, \alpha) : \mu_{A*B}(x, y) > \alpha, \alpha \in I\} \subset Y \times R^1 .$$

Аналогичным образом можно ввести сечение $R_{A*B}(y) \subset X \times R^1$ по y.

Назовем конъюнктивной проекцией отношения $R_{A*B} \subset X \times Y \times R^1$ на пространство $Y \times R^1$ множество

$$\hat{P} r_{Y\times R^1} R_{A*B} = \underset{x\in D_A}{\cap} R_{A*B}(x) = \left\{(y,\alpha): \underset{x}{\wedge}(\mu_A(x) * \mu_B(y)) \geq \alpha\right\} \subset Y \times R^1 .$$

Множество $\overset{\vee}{P} r_{Y\times R^1} R_{A*B} = \underset{x\in D_A}{\cup} R_{A*B}(x) = \left\{(y,\alpha): \underset{x}{\vee}(\mu_A(x) * \mu_B(y)) \geq \alpha\right\} \subset Y \times R^1$ будем называть дизъюнктивной проекцией. В этих выражениях операторы $\underset{x}{\wedge}$ и $\underset{x}{\vee}$ отвечают операциям минимизации и максимизации.

Удобство использования логической символики в записи отношений и их проекций, в частности, состоит в том, что для преобразования условий, которые определяют эти множества, можно пользоваться соответствующими законами логики.

Пользуясь этим, покажем, что конъюнктивное и дизъюнктивное проектирование разрешает найти исходные множества A и B по заданным отношением $R_{A\vee B}$ и $R_{A\wedge B}$ соответственно. Иначе говоря, покажем, что

$$\hat{P} r_{Y\times R^1} R_{A\vee B} = B, \quad \hat{P} r_{X\times R^1} R_{A\vee B} = A \text{ и } \overset{\vee}{P} r_{Y\times R^1} R_{A\wedge B} = B, \quad \overset{\vee}{P} r_{X\times R^1} R_{A\wedge B} = A .$$

В частности, для случая проекций на $Y \times R^1$ воспользуемся распределительным законом, а именно,

$$\underset{x}{\wedge}(\mu_A(x) \vee \mu_B(y)) = \left(\underset{x}{\wedge}\mu_A(x)\right) \vee \mu_B(y) = \mu_B(y),$$
$$\underset{x}{\vee}(\mu_A(x) \wedge \mu_B(y)) = \left(\underset{x}{\vee}\mu_A(x)\right) \wedge \mu_B(y) = \mu_B(y),$$

где учтено, что $\underset{x}{\wedge}\mu_A(x) = 0$ и $\underset{x}{\vee}\mu_A(x) = 1$.

Проекции на $X \times R^1$ находятся аналогично.

Приведенные здесь операции проектирования позволяют находить соответствие между множествами A и B. В рассмотренных случаях B можно рассматривать как образ множества A при отображении его с помощью соответствующих отношений. Типичная задача, которая здесь возникает– это найти образ B', если в качестве исходного выступает не все множество A, а его собственное подмножество. Подобная задача для случая отношения, которое задается операцией нечеткой импликации, рассматривается в следующем разделе.

Вывод на основе нечеткой импликации

Пусть $A \subset X \times R^1$ и $B \subset Y \times R^1$ - нечеткие множества с функциями принадлежности $\mu_A(x)$ и $\mu_B(y)$. Рассмотрим импликацию “ЕСЛИ А, ТО В”. Здесь нечеткое множество A рассматривается как нечеткая посылка, а B - как нечеткое заключение. Пусть на самом деле (в результате наблюдения) реализовалась посылка A' такая, что $A' \cap A \neq \varnothing$. Нужно найти заключение B', которое отвечает условию $A' \cap A$.

Для решения задачи введем нечеткое логическое отношение

$$R_{A\to B} = \left\{(x,y,\alpha): \mu_{B|A}(x,y) \geq \alpha, \alpha \in I\right\},$$

где $\mu_{B|A}(x,y)$ определяется выражением (2).

Искомое множество B' найдем как конъюнктивную проекцию

$$B' = \left\{(y,\alpha): \mu_{B'}(y) \geq \alpha \cdot \alpha_0\right\} = \left\{(y,\alpha): \underset{x}{\wedge}\left(\overline{\mu_{A\cap A'}}(x) \vee \mu_B(y)\right) \geq \alpha \cdot \alpha_0, \alpha \in I\right\},$$

где α_0 - нормирующий параметр.

Для дополнения $\overline{B}'$ будем иметь:

$$B'=\{(y,\alpha):\overline{\mu_{B'}}(y)\geq\alpha\cdot\alpha_0\}=\{(y,\alpha):\vee_x(\mu_{A\cap A'}(x)\wedge\overline{\mu}_B(y))\geq\alpha\cdot\alpha_0\}.$$

С учетом распределительного свойства операции дизъюнкции по отношению к конъюнкции можем записать:

$$\overline{B}'=\{(y,\alpha):(\vee_x\mu_{A\cap A'}(x))\wedge\overline{\mu}_B(y)\geq\alpha\cdot\alpha_0\}.$$

Поскольку пересечение $A'\cap A\neq\varnothing$, то существует максимум $\vee_x\mu_{A\cap A'}(x)$. Пусть $\vee_x\mu_{A\cap A'}(x)=a$, где $a\in I$. Тогда $\overline{B}'=\{(y,\alpha):a\wedge(1-\mu_B(y))\geq\alpha\cdot\alpha_0\}$. Нормирующий параметр α_0 здесь будет равный a. Тогда для искомого множества получим: $\overline{B}'=\left\{(y,\alpha):1-\min_y\left(1,\frac{1-\mu_B(y)}{a}\right)\geq\alpha,\alpha\in I\right\}$. Для функции $\mu'_B(y)$ принадлежности можем записать $\mu'_B(y)=\begin{cases}0, якщо\,(1-\mu_B(y))/a>1,\\ 1-(1-\mu_B(y))/a, якщо\,(1-\mu_B(y))/a\leq 1\end{cases}$.

Соответствующие построения выполнены на рис.3.

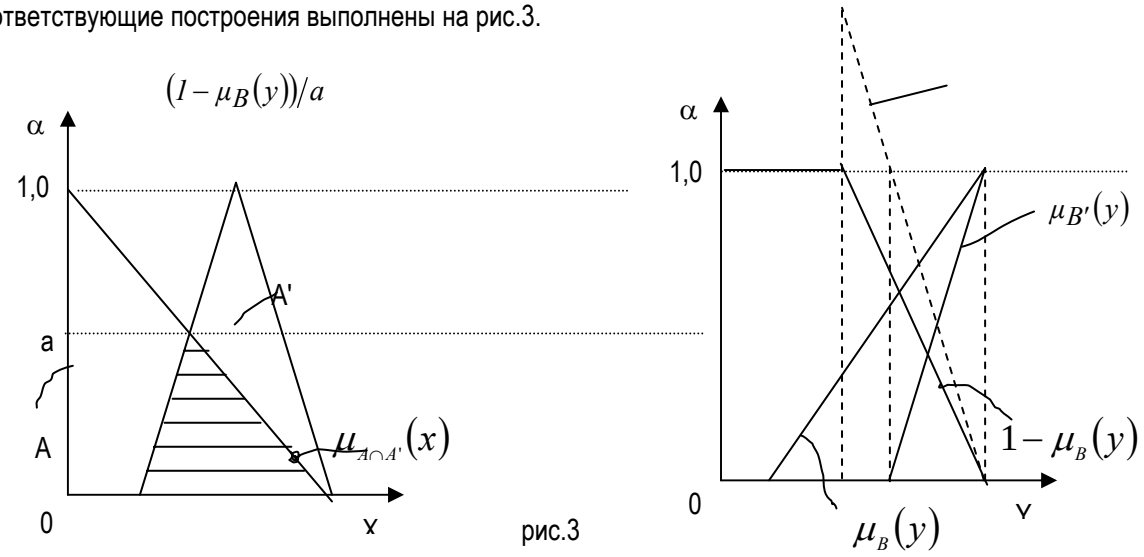


рис.3

На рисунке графически представленные все множества, которые принимают участие в приведенной схеме вывода. Функции принадлежности $\mu_A(x)$, $\mu_{A'}(x)$, $\mu_B(y)$, $\mu_{B'}(y)$, $\mu_{B_H}{}'(y)$– кусочно-линейные. Соответствующие нечеткие множества – это подграфики этих функций. Пересечение множеств A и A' обозначено штриховкой. Множество $B_H{}'$ – это искомое заключение к нормализации.

Из рисунка видно, что множество B' по площади меньше B. Это отвечает уменьшению значения посылки в результате пересечения A и A'.

Вывод.

Представление модели нечеткого множества как подграфика функции принадлежности позволяет строить нечеткие множества-отношения для типичных логических операций. На этой основе для нечеткой

импликации предложенна процедура вывода в случае частичного совпадения исходной посылки с его реализацией, которая наблюдается.

Предложенную здесь процедуру нечеткого вывода можно использовать и в более общих случаях.

Литература.

[Попов,1988] Э.В.Попов. Экспертные системы. М.: Наука, 1988- 283с.
[Форсайт,1987] Р.Форсайт (ред). Экспертные системы. М.: Радиосвязь, 1987-220с.
[Поспелов,1986] Д.А.Поспелов (ред). Нечеткие множества в моделях управления и искусственного интеллекта. М.: Наука, 1986- 312с.
[Змитрович,1997] Змитрович А.И. Интеллектуальные информационные системы. Минск, 1997-368с.

Авторы.

Геннадий Бакан - Институт прикладного системного анализа НАН Украины и Миннауки; 03056 Киев-56, Украина; e-mail:ipn@com.ua
Ольга Кононенко – 03680 Киев-680, Украина; e-mail: okononenko@unicyb.kiev.ua

REPRESENTING REFLECTIVE LOGIC IN MODAL LOGIC

Frank M. Brown

Abstract*: The nonmonotonic logic called Reflective Logic is shown to be representable in a monotonic Modal Quantificational Logic whose modal laws are stronger than S5. Specifically, it is proven that a set of sentences of First Order Logic is a fixed-point of the fixed-point equation of Reflective Logic with an initial set of axioms and defaults if and only if the meaning of that set of sentences is logically equivalent to a particular modal functor of the meanings of that initial set of sentences and of the sentences in those defaults. This result is important because the modal representation allows the use of powerful automatic deduction systems for Modal Logic and because unlike the original Reflective Logic, it is easily generalized to the case where quantified variables may be shared across the scope of the components of the defaults thus allowing such defaults to produce quantified consequences. Furthermore, this generalization properly treats such quantifiers since all the laws of First Order Logic hold and since both the Barcan Formula and its converse hold.*

Keywords*: Reflective Logic, Modal Logic, Nonmonotonic Logic.*

1. Introduction

One of the simplest nonmonotonic logics which inherently deals with entailment conditions in addition to possibility conditions in its defaults is the so-called Reflective Logic [Brown 1989]. The basic idea of Reflective Logic is that there are some assumptions Γ_i and some non-logical "inference rules" of the form:

$$\frac{\alpha_i : \beta_{i1} \ldots \beta_{im_i}}{\chi_i}$$

which are intended to suggest that χ_i may be inferred whenever α_i is inferable and each $\beta_{i1} \ldots \beta_{im_i}$ is consistent with everything that is inferable. Such "inference rules" are not recursive and are circular in that the determination as to whether χ_i is derivable depends on whether β_{ij} is consistent which in turn depends on what was derivable

from this and other defaults. Thus, tentatively applying such inference rules by checking the consistency of $\beta_{i1}...\beta_{im_i}$ with only the current set of inferences produces a χ_i result which may later have to be retracted. For this reason valid inferences in a nonmonotonic logic such as Reflective Logic are essentially carried out not in the original nonmonotonic logic, but rather in some (monotonic) metatheory in which that nonmonotonic logic is defined. [Brown 1989] explicated this intuition[1] by defining Reflective Logic in terms of the set theoretic proof theory metalanguage of First Order Logic (i.e. FOL) with the following fixed point expression:

$$'\kappa=(rl\ '\kappa\ \{'\Gamma_i\}\ '\alpha_i{:}'\beta_{ij}/'\chi_i)$$

where rl is defined as: $(rl\ '\kappa\ \{'\Gamma_i\}\ '\alpha_i{:}'\beta_{ij}/'\chi_i) =_{df} (fol(\{'\Gamma_i\}\cup\{'\chi_i{:}('\alpha_i\varepsilon'\kappa)\wedge\wedge_{j=1,m_i}('(\neg\beta_{ij})\notin'\kappa)\}))$

where $'\alpha_i$, β_{ij}, and $'\chi_i$ are the closed sentences of FOL occurring in the ith "inference rule" and $\{'\Gamma_i\}$ is a set of closed sentences of FOL and $'\Gamma_i$ is the ith sentence in that set. A closed sentence is a sentence without any free variables. fol is a function which produces the set of theorems derivable in FOL from the set of sentences to which it is applied. The quotations appended to the front of these Greek letters indicate references in the metalanguage to sentences of the FOL object language. Interpreted doxastically this fixed point equation states:

> the set of closed sentences which are believed is equal to:
> the set of closed sentences derived in FOL from
> the union of the set of closed sentences: $\{'\Gamma_i\}$,
> and the set of closed sentences of the form $'\chi_i$ such that for each i,
> the closed sentence $'\alpha_i$ is believed and for each j, the closed sentence $'\beta_{ij}$ is believable.

The purpose of this paper is to show that all this metatheoretic machinery including the formalized syntax of FOL, the proof theory of FOL, the axioms of a strong set theory, and the set theoretic fixed-point equation is not needed and that the essence of Reflective Logic in representable as a necessary equivalence in a simple (monotonic) Modal Quantificational Logic. Interpreted as a doxastic logic this necessary equivalence states:

> that which is believed is logically equivalent to
> for each i , Γ_I and for each i, if α_i is believed and for each j, β_{ij} is believable then χ_i

thereby eliminating all mention of any metatheoretic machinery.

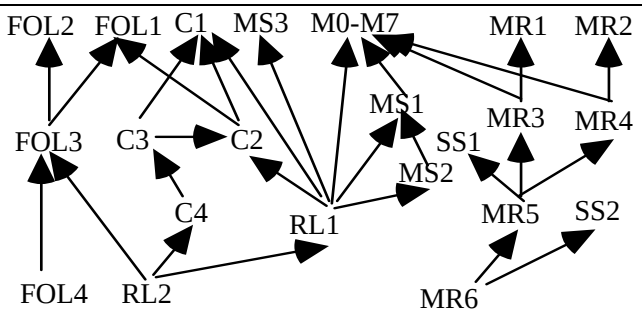


Figure 1: Dependencies among the Theorems

The remainder of this paper proves that this modal representation is equivalent to Reflective Logic. Section 2 describes a formalized syntax for a FOL object language. Section 3 describes the part of the proof theory of FOL needed herein (i.e. theorems FOL1-FOL4). Section 4 describes the Intensional Semantics of FOL which includes

[1] This explication is simpler but less sophisticated in its properties than that of Default Logic [Reiter 1980]. The fixed-points of both logics obey the laws: $'\kappa=(fol\ '\kappa)$, $'\kappa\supseteq\{'\Gamma_i\}$, and $((\alpha_i\varepsilon'\kappa)\wedge\wedge_{j=1,m_i}('(\neg\beta_{ij})\notin'\kappa))\rightarrow('\chi_i\varepsilon'\kappa)$. However, the fixed points of Default Logic are a subset of the fixed-points of Reflective Logic, but the converse is in general not true. Moreover, the fixed-points of Reflective Logic are the kernels of the fixed points of Autoepistemic Logic [Moore 1985].

laws giving the meaning of FOL sentences: M0-M7, theorems giving the meaning of sets of FOL sentences: MS1, MS2, MS3, and laws specifying the relationship of meaning and modality to the proof theory of FOL (i.e. the laws R0, A1, A2, and A3 and the theorems: C1, C2, C3, and C4). The modal version of Reflective Logic, called RL, is defined in section 5 and explicated with theorems MR1-MR6 and SS1-SS2. In section 6, this modal version is shown by theorems RL1 and RL2 to be equivalent to the set theoretic fixed point equation for Reflective Logic. Figure 1 outlines the relationship of all these theorems in producing the final theorems RL2, FOL4, and MR6.

2. Formal Syntax of First Order Logic

We use a First Order Logic (i.e. FOL) defined as the six tuple: ($\rightarrow$, #f, $\forall$, *vars*, *predicates*, *functions*) where $\rightarrow$, #f, and $\forall$ are logical symbols, *vars* is a set of variable symbols, *predicates* is a set of predicate symbols each of which has an implicit arity specifying the number of associated terms, and *functions* is a set of function symbols each of which has an implicit arity specifying the number of associated terms. The sets of logical symbols, variables, predicate symbols, and function symbols are pairwise disjoint. Lower case Roman letters possibly indexed with digits are used as variables. Greek letters possibly indexed with digits or lower case roman letters are used as syntactic metavariables. γ, γ_1,...γ_n, range over the variables, ξ, ξ_1...ξ_n range over sequences of variables of an appropriate arity, π,π_1...π_n range over the predicate symbols, ϕ, ϕ_1...ϕ_n range over function symbols, δ, δ_1...δ_n, σ range over terms, and α, α_1...α_n, β, β_1...β_n,χ, χ_1...χ_n, Γ_1,...Γ_n,φ range over sentences. The terms are of the forms γ and (ϕ δ1...δn), and the sentences are of the forms ($\alpha\rightarrow\beta$), #f, ($\forall\gamma$ α), and (π δ_1...δ_n). A nullary predicate π or function ϕ is written as a sentence or a term without parentheses. $\varphi\{\pi/\lambda\xi\alpha\}$ represents the replacement of all occurrences of π in φ by $\lambda\xi\alpha$ followed by lambda conversion. The primitive symbols are shown in Figure 2 with their intuitive interpretations.

Symbol	Meaning
$\alpha\rightarrow\beta$	if α then β.
#f	falsity
$\forall\gamma\ \alpha$	for all γ, α.

Figure 2: Primitive Symbols of First Order Logic

The defined symbols are listed in Figure 3 with their definitions and intuitive interpretations.

Symbol	Definition	Meaning		Symbol	Definition	Meaning
$\neg\alpha$	$\alpha\rightarrow$ #f	not α		$\alpha\wedge\beta$	$\neg(\alpha\rightarrow\neg\beta)$	α and β
#t	$\neg$ #f	truth		$\alpha\leftrightarrow\beta$	$(\alpha\rightarrow\beta)\wedge(\beta\rightarrow\alpha)$	α if and only if β
$\alpha\vee\beta$	$(\neg\alpha)\rightarrow\beta$	α or β		$\exists\gamma\ \alpha$	$\neg\forall\gamma\neg\alpha$	for some γ , α

Figure 3: Defined Symbols of First Order Logic

The FOL object language expressions are referred in the metalanguage (which also includes a FOL syntax) by inserting a quote sign in front of the object language entity thereby making a structural descriptive name of that entity. In addition to referring to object language sentences, the formalized metalanguage also needs to refer to sets of sentences of FOL. Generally, a set of sentences is represented as: {'Γ_i} which is defined as: {'Γ_i: #t} which in turn is defined as: {s: $\exists$i(s='Γ_i)} where i ranges over some range of numbers (which may be finite or non-infinite). With a slight abuse of notation we also write 'κ, 'Γ to refer to such sets.

3. Proof Theory of First Order Logic

First Order Logic (i.e. FOL) is axiomatized with a recursively enumerable set of theorems as the set of axioms is itself recursively enumerable and its inference rules are recursive. The axioms and inference rules of FOL [Mendelson 1964] are those given in Figure 4. They form a standard set of axioms and inference rules for FOL.

MA1: $\alpha \rightarrow (\beta \rightarrow \alpha)$ MR1: from α and $(\alpha \rightarrow \beta)$ infer β

MA2: $(\alpha \rightarrow (\beta \rightarrow \rho)) \rightarrow ((\alpha \rightarrow \beta) \rightarrow (\alpha \rightarrow \rho))$ MR2: from α infer $(\forall\gamma\ \alpha)$

MA3: $((\neg\alpha) \rightarrow (\neg\beta)) \rightarrow (((\neg\alpha) \rightarrow \beta) \rightarrow \alpha)$

MA4: $(\forall\gamma\ \alpha) \rightarrow \beta$ where β is the result of substituting an expression (which is free for the free positions of γ in α) for all the free occurrences of γ in α.

MA5: $((\forall\gamma(\alpha \rightarrow \beta)) \rightarrow (\alpha \rightarrow (\forall\gamma\ \beta)))$ where γ does not occur in α.

Figure 4: Inferences Rules and Axioms of FOL

In order to talk about sets of sentences we include in the metatheory set theory symbolism as developed along the lines of [Quine 1976]. This set theory includes the symbols ε, $\notin$, $\supseteq$, $=$, $\cup$ as is defined therein.

The derivation operation (i.e. fol) of any First Order Logic obeys the Inclusion (i.e. FOL1) and Idempotence (i.e. FOL2) properties:

FOL1: (fol 'κ)$\supseteq$'κ Inclusion

FOL2: (fol 'κ)$\supseteq$(fol(fol 'κ)) Idempotence

From these two properties we prove:

FOL3: (rl 'κ 'Γ 'α_i:'β_{ij}/'χ_i))=(fol(rl 'κ 'Γ 'α_i:'β_{ij}/'χ_i))

proof: FOL1 and FOL2 imply that (fol(fol 'κ))=(fol 'κ). Since rl begins with fol this implies: 'κ=(fol(rl 'κ)) QED.

FOL4: ('κ=(rl 'κ 'Γ 'α_i:'β_{ij}/'χ_i))$\rightarrow$('κ=(fol 'κ))

proof: From the hypothesis and FOL3: 'κ=(fol(rl 'κ 'Γ 'α_i:'β_{ij}/'χ_i)) is derived. Using the hypothesis to replace (rl 'κ 'Γ 'α_i:'β_{ij}/'χ_i) by 'κ in this result gives: 'κ=(fol 'κ). QED.

4. Intensional Semantics of FOL

The meaning (i.e. mg) [Brown 1978, Boyer&Moore 1981] or rather disquotation of a sentence of First Order Logic (i.e. FOL) is defined to satisfy the laws given in Figure 5 below[2]. mg is defined in terms of mgs which maps each FOL object language sentence and an association list into a meaning. Likewise, mgn maps a FOL object language term and an association list into a meanings. An association list is simply a list of pairs consisting of an object language variable and the meaning to which it is bound.

M0: (mg 'α) =df (mgs '$(\forall\gamma 1 \ldots \gamma n\ \alpha)$'()) where '$\gamma 1 \ldots$'$\gamma n$ are all the free variables in 'α

M1: (mgs '$(\alpha \rightarrow \beta)$a) $\leftrightarrow$ ((mgs 'α a)$\rightarrow$(mgs 'β a))

M2: (mgs '#f a) $\leftrightarrow$ #f

M3: (mgs '$(\forall\ \gamma\ \alpha)$a) $\leftrightarrow$ $\forall$x(mgs 'α(cons(cons 'γ x)a))

M4: (mgs '$(\pi\ \delta_1 \ldots \delta_n)$a) $\leftrightarrow$ $(\pi$(mgn 'δ_1 a)...(mgn 'δ_n a)) for each predicate symbol 'π.

M5: (mgn '$(\phi\ \ \delta_1 \ldots \delta_n)$a) = $(\phi$(mgn 'δ_1 a)...(mgn 'δ_n a)) for each function symbol 'ϕ.

M6: (mgn 'γ a) = (lookup 'γ a) where (lookup 'γ a) is the value associated wit 'γ in the association list a.

Figure 5: The Meaning of FOL Sentences

For example, the meaning of the sentence "Everything is less than something" is the proposition that everything is less than something. Thus the meaning operator disquotes its argument. Here is an example derivation:

[2] The laws M0-M7 are analogous to Tarski's definition of truth except that finite association lists are used to bind variables to values rather than infinite sequences. mg is interpreted as being meaning rather than truth since it is also intended to satisfy the modal case: (mgs '([] α)a) $\leftrightarrow$ ([](mgs α a)) since it could be necessary that a particular sentence 'α were true: ([](istrue 'α a)) without it being true that it was necessary: (istrue '([] α) a).

(mg '(∀x∃y(< x y)))

Replacing the defined symbols of the object language by primitive symbols of the object language gives:

(mg '(∀x((∀y((< x y)→#f))→#f))). By M0 this is equivalent to: (mgs '(∀x((∀y((< x y)→#f))→#f)) '())

By M3 this is equivalent to: ∀x(mgs '((∀y((< x y)→#f))→#f) (cons(cons 'x x)'()))

By M1 this is equivalent to: ∀x((mgs '(∀y((< x y)→#f)) (cons(cons 'x x)'())) →(mgs '#f (cons(cons 'x x)'())))

By M2 this is equivalent to: ∀x((mgs '(∀y((< x y)→#f)) (cons(cons 'x x)'()))→#f)

We would now like to apply M3 to: (mgs '(∀y((< x y)→#f)) (cons(cons 'x x)'()))

but we cannot since the bound variable x in M3 would capture the variable x which is free in this expression. In order to apply M3 we must first rename the bound variable x in M3 to be some other variable which will not capture any free variables in this expression. In this case we rename the bound x in M3 to be y, and then use that version of M3 to produce the equivalent expression:

∀x((∀y(mgs '((< x y)→#f) (cons(cons 'y y)(cons(cons 'x x)'()))))) →#f)

By M1 this is equivalent to:

∀x((∀y((mgs '(< x y) (cons(cons 'y y)(cons(cons 'x x)'())))) →(mgs '#f (cons(cons 'y y)(cons(cons 'x x)'()))))) →#f)

By M2 this is equivalent to: ∀x∃y((mgs '(< x y) (cons(cons 'y y)(cons(cons 'x x)'())))) By M4 this is equivalent to: ∀x∃y(<(mgn 'x(cons(cons 'y y)(cons(cons 'x x)'())))) (mgn 'y (cons(cons 'y y)(cons(cons 'x x)'()))))

By M6 twice this is equivalent to: ∀x∃y(< x y)

The meaning of a set of sentences is defined in terms of the meanings of the sentences in the set as:

(ms 'κ) =df ∀s((sε'κ)→(mg s))

MS1: (ms{'α: Γ}) ↔ ∀ξ(Γ→α) where ξ is the sequence of all the free variables in 'α and where Γ is any sentence of the intensional semantics.

proof: (ms{'α:Γ}) Unfolding ms and the set pattern abstraction symbol gives: ∀s((sε{s: ∃ξ((s='α)∧Γ)})→(mg s))

where ξ is a sequence of the free variables in 'a. This is equivalent to: ∀s((∃ξ((s='α)∧Γ)))→(mg s))

which is logically equivalent to: ∀s∀ξ (((s='κ)∧Γ)→(mg s)) which is equivalent to: ∀ξ(Γ→(mg 'α))

Unfolding mg using M0-M7 then gives: ∀ξ(Γ→α) QED

The meaning of the union of two sets of FOL sentences is the conjunction of their meanings (i.e. MS1) and the meaning of a set is the meaning of all the sentences in the set (i.e. MS2):

MS2: (ms{'Γi}) ↔ ∀i∀ξiΓi

proof: (ms{'Γi}) Unfolding the set notation gives: (ms{'Γi: #t})

By MS1 this is equivalent to: ∀i∀ξi(#t→Γi) which is equivalent to: ∀iΓi QED.

MS3: (ms('κ∪'Γ)) ↔ ((ms 'κ)∧(ms 'Γ))

proof: Unfolding ms and union in: (ms('κ∪'Γ)) gives: ∀s((sε{s: (sε'κ)∨(sε'Γ)})→(mg s)) or rather:

∀s(((sε'κ)∨(sε'Γ))→(mg s)) which is logically equivalent to: (∀α((sε'κ)→(mg s)))∧(∀s((sε'Γ)→(mg s)))

Folding ms twice then gives:((ms 'κ)∧(ms 'Γ)) QED.

The meaning operation may be used to develop an Intensional Semantics for a FOL object language by axiomatizing the modal concept of necessity so that it satisfies the theorem:

C1: ('αε(fol 'κ)) ↔ ([] ((ms 'κ)→(mg 'α)))

for every sentence 'α and every set of sentences 'κ of that FOL object language. The necessity symbol is represented by a box: [] herein constructed from two square brackets. C1 states that a sentence of FOL is a FOL-theorem (i.e. fol) of a set of sentences of FOL if and only if the meaning of that set of sentences necessarily implies the meaning of that sentence. One modal logic which satisfies C1 is the Z Modal Quantificational Logic described in [Brown 1987; Brown 1989] whose theorems are recursively enumerable and which extends the

weaker possibility axioms used in [Lewis 1936; Bressan 1972; Hendry & Pokriefka 1985].[3] We will not discuss this axiomatization herein except to note that Z includes all the laws of S5 modal Logic [Hughes & Cresswell 1968] whose modal axioms and inference rules are given in Figure 6. κ and Γ represent arbitrary sentences of the intentional semantics.

R0: from α infer ([] κ) A2: $([](\kappa \rightarrow \Gamma)) \rightarrow (([]\kappa) \rightarrow ([]\Gamma))$

A1: $([]\kappa) \rightarrow \kappa$ A3: $([]\kappa) \vee ([]\neg[]\kappa)$

Figure 6: The Laws of S5 Modal Logic

These S5 modal laws and the laws of FOL given in Figure 4 constitute an S5 Modal Quantificational Logic similar to [Carnap 1946; Carnap 1956], and a FOL version [Parks 1976] of [Bressan 1972] in which the Barcan formula: $(\forall\gamma([]\kappa)) \rightarrow ([]\forall\gamma\kappa)$ and its converse hold. The R0 inference rule implies that anything derivable in the metatheory is necessary. Thus, in any logic with R0, contingent facts would never be asserted as additional axioms of the metatheory. For example, we would not assert $([](\kappa \leftrightarrow \Gamma))$ as an axiom and then try to prove $([](\kappa \rightarrow \alpha))$. Instead we would try to prove that $([](\kappa \leftrightarrow \Gamma)) \rightarrow ([](\kappa \rightarrow \alpha))$.

The defined Modal symbols used herein are listed in Figure 7 with their definitions and interpretations.

Symbol	Definition	Meaning		Symbol	Definition	Meaning
$<>\kappa$	$\neg [] \neg\kappa$	α is logically possible		$[\kappa]\ \Gamma$	$[]\ (\kappa \rightarrow \Gamma)$	β entails α
$\kappa \equiv \Gamma$	$[]\ (\kappa \leftrightarrow \Gamma)$	α is logically equivalent to β		$<\kappa>\ \Gamma$	$<>\ (\kappa \wedge \Gamma)$	α and β is logically possible

Figure 7: Defined Symbols of Modal Logic

For example, folding the definition of entailment, C1 may be rewritten more compactly as:

C1': $('\alpha\varepsilon(\text{fol }'\kappa)) \leftrightarrow ([(\text{ms }'\kappa)](\text{mg }'\alpha))$

This compact notation for entailment is used hereafter.

From the laws of the Intensional Semantics we prove that the meaning of the set of FOL consequences of a set of sentences is the meaning of that set of sentences (C2), the FOL consequences of a set of sentences contain the FOL consequences of another set if and only if the meaning of the first set entails the meaning of the second set (C3), and the sets of FOL consequences of two sets of sentences are equal if and only if the meanings of the two sets are logically equivalent (C4):

C2: $(\text{ms}(\text{fol }'\kappa)) \equiv (\text{ms }'\kappa)$

proof: The proof divides into two cases:

(1) $[(\text{ms }'\kappa)](\text{ms}(\text{fol }'\kappa))$ Unfolding the second ms gives: $[(\text{ms }'\kappa)]\forall s((s\varepsilon(\text{fol }'\kappa)) \rightarrow (\text{mg } s))$

By the soundness part of C1 this is equivalent to: $[(\text{ms }'\kappa)]\forall s(([(\text{ms }'\kappa)](\text{mg } s)) \rightarrow (\text{mg } s))$

By the S5 laws this is equivalent to: $\forall s(([(\text{ms }'\kappa)](\text{mg } s)) \rightarrow [(\text{ms }'\kappa)](\text{mg } s))$ which is a tautology.

(2) $[(\text{ms}(\text{fol }'\kappa))](\text{ms }'\kappa)$ Unfolding ms twice gives: $[\forall s((s\varepsilon(\text{fol }'\kappa)) \rightarrow (\text{mg } s))]\forall s((s\varepsilon'\kappa) \rightarrow (\text{mg } s))$

which is: $[\forall s((s\varepsilon(\text{fol }'\kappa)) \rightarrow (\text{mg } s))]((s\varepsilon'\kappa) \rightarrow (\text{mg } s))$ Backchaining on the hypothesis and then dropping it gives: $(s\varepsilon'\kappa) \rightarrow (s\varepsilon(\text{fol }'\kappa))$. Folding $\supseteq$ gives an instance of FOL1. QED.

C3: $(\text{fol }'\kappa) \supseteq (\text{fol }'\Gamma) \leftrightarrow ([(\text{ms }'\kappa)](\text{ms }'\Gamma))$

proof: Unfolding $\supseteq$ gives: $\forall s((s\varepsilon(\text{fol }'\Gamma)) \rightarrow (s\varepsilon(\text{fol }'\kappa)))$

By C1 twice this is equivalent to: $\forall s(([(\text{ms }'\Gamma)](\text{mg } s)) \rightarrow ([(\text{ms }'\kappa)](\text{mg } s)))$

[3] An S5 modal logic which satisfies a metatheorem analogous C1 for Propositional Logic is the system S5c given in [Hendry and Pokriefka 1985] which has axiom schemes stating that every conjunction of distinct propositional constants is logically possible. This extends the trivial possibility axiom that some proposition is neither #t nor #f used in [Lewis 1936; Bressan 1972]. A modal logic which satisfies C1 for FOL is the Z Modal Quantificational Logic described in [Brown 1987; Brown 1989] whose theorems are recursively enumerable. This logic has the metatheorem: $(<>\Gamma)\{\pi/\lambda\xi\alpha\} \rightarrow (<>\Gamma)$ where Γ is a sentence of FOL.

By the laws of S5 modal logic this is equivalent to: $([(ms\ '\kappa)]\forall s(([(ms\ '\Gamma)](mg\ s)))\rightarrow(mg\ s))$

By C1 this is equivalent to: $[(ms\ '\kappa)]\forall s((s\varepsilon(fol\ '\Gamma))\rightarrow(mg\ s))$. Folding ms then gives: $[(ms\ '\kappa)](ms(fol\ '\Gamma))$

By C2 this is equivalent to: $[(ms\ '\kappa)](ms\ '\Gamma)$. QED.

C4: $((fol\ '\kappa)=(fol\ '\Gamma))\leftrightarrow((ms\ '\kappa)\equiv(ms\ '\Gamma))$

proof: This is equivalent to $(((fol\ '\kappa)\supseteq(fol\ '\Gamma))\wedge((fol\ '\Gamma)\supseteq(fol\ '\kappa)))\leftrightarrow([(ms\ '\kappa)](ms\ '\Gamma))\wedge([(ms\ '\Gamma)](ms\ '\kappa))$ which follows by using C3 twice.

5. Reflective Logic Represented in Modal Logic

The fixed point equation for Reflective Logic may be expressed as a necessary equivalence in an S5 Modal Quantificational Logic as follows: $\kappa\equiv(RL\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)$ where RL is defined as: $(RL\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)$ =df $\Gamma\wedge\forall i((([\kappa]\alpha_i)\wedge(\wedge_{j=1,mi}(<\kappa>\beta_{ij})))\rightarrow\chi_i)$ where Γ, α_i, β_{ij}, and χ_i are propositions of FOL. When the context is obvious $\Gamma\ \alpha_i{:}\beta_{ij}/\chi_i$ is omitted and just $(RL\ \kappa)$ is written. Given below are some simple properties of RL. The first two theorems state that RL entails Γ and any conclusion χ_i of a default whose entailment condition holds in κ and whose possible conditions are possible with κ.

MR1: $[(RL\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)]\Gamma$

proof: By R0 it suffices to prove: $(RL\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)\rightarrow\Gamma$. Unfolding RL gives:
$\Gamma\wedge\forall_i((([\kappa]\alpha_i)\wedge(\wedge_{j=1,mi}(<\kappa>\beta_{ij})))\rightarrow\chi_i)\rightarrow\Gamma$ which is a tautology. QED.

MR2: $(([\kappa]\alpha_i)\wedge(\wedge_{j=1,mi}(<\kappa>\beta_{ij})))\rightarrow([(RL\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)]\chi_i)$

proof: Unfolding RL gives: $(([\kappa]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow([\Gamma\wedge\forall_i((([\kappa]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow\chi_i)]\chi_i)$

Using the hypotheses on the ith instance gives:
$(([\kappa]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow([\Gamma\wedge\forall_i((([\kappa]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow\chi_i)\wedge\chi_i]\chi_i)$ which is a tautology. QED.

The concept (i.e. ss) of the combined meaning of all the sentences of the FOL object language whose meanings are entailed by a proposition is defined as follows:

$(ss\ \kappa)$ =df $\forall s(([\kappa](mg\ s))\rightarrow(mg\ s))$

SS1 shows that a proposition entails the combined meaning of the FOL object language sentences that it entails. SS2 shows that if a proposition is necessarily equivalent to the combined meaning of the FOL object language sentences that it entails, then there exists a set of FOL object language sentences whose meaning is necessarily equivalent to that proposition:

SS1: $[\kappa](ss\ \kappa)$

proof: By R0 it suffices to prove: $\kappa\rightarrow(ss\ \kappa)$. Unfolding ss gives: $\kappa\rightarrow\forall s(([\kappa](mg\ s))\rightarrow(mg\ s))$

which is equivalent to:$\forall s(([\kappa](mg\ s))\rightarrow(\kappa\rightarrow(mg\ s)))$ which is an instance of A1. QED.

SS2: $(\kappa\equiv(ss\ \kappa))\rightarrow\exists s(\kappa\equiv(ms\ s))$

proof: Letting s be $\{s: ([\kappa](mg\ s))$ gives $(\kappa\equiv(ss\ \kappa))\rightarrow(\kappa\equiv(ms\{s: ([\kappa](mg\ s))))$. Unfolding ms and lambda conversion gives: $(\kappa\equiv(ss\ \kappa))\leftrightarrow(\kappa\equiv\forall s(([\kappa](mg\ s))\rightarrow(mg\ s)))$. Folding ss gives a tautology. QED.

The theorems MR3 and MR4 are analogous to MR1 and MR2 except that RL is replaced by the combined meanings of the sentences entailed by RL.

MR3: $[ss(RL\ \kappa\ \forall i\Gamma_i\ \alpha_i{:}\beta_{ij}/\chi_i)]\forall i\Gamma_i$

proof: By R0 it suffices to prove: $(ss(RL\ \kappa\ \forall i\Gamma_i\ \alpha_i{:}\beta_{ij}/\chi_i))\rightarrow\forall i\Gamma_I$ which is equivalent to:

$(ss(RL\ \kappa\ \forall i\Gamma_i\ \alpha_i{:}\beta_{ij}/\chi_i))\rightarrow\Gamma_i$. Unfolding ss gives:: $\forall s((([(RL\ \kappa\ \forall i\Gamma_i\ \alpha_i{:}\beta_{ij}/\chi_i)](mg\ s))\rightarrow(mg\ s)))\rightarrow\Gamma_i$

which by the meaning laws M0-M8 is equivalent to: $(\forall s((([(RL\ \kappa\ \forall i\Gamma_i\ \alpha_i{:}\beta_{ij}/\chi_i)](mg\ s))\rightarrow(mg\ s)))\rightarrow(mg\ '\Gamma_i)$

Backchaining on (mg $'\Gamma_i$) with s in the hypothesis being $'\Gamma_i$ in the conclusion shows that it suffices to prove:
([(RL $\kappa\ \forall i\Gamma_i\ \alpha_i{:}\beta_{ij}/\chi_i$)](mg $'\Gamma_i$)) which by the meaning laws: M0-M8 is equivalent to: ([(RL $\kappa\ \forall i\Gamma_i\ \alpha_i{:}\beta_{ij}/\chi_i$)]$\Gamma_i$)
which by the laws of S5 Modal Logic is equivalent to: ([(RL $\kappa\ \forall i\Gamma_i\ \alpha_i{:}\beta_{ij}/\chi_i$)]$\forall i\Gamma_i$)
which is an instance of theorem MR1. QED.

MR4: (([κ]α_i)$\wedge$($\wedge_{j=1,mi}$(<κ>β_{ij})))$\rightarrow$ ([ss(RL $\kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i$)]$\chi_i$)
proof: Unfolding the last ss gives: (([κ]α_i)$\wedge\wedge_{j=1,mi}$(<κ>β_{ij}))$\rightarrow$ ([$\forall$s(([(RL $\kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i$)](mg s))$\rightarrow$(mg s))]$\chi_i$)
Instantiating s in the hypothesis to $'\chi_i$ and then dropping the hypothesis gives:
(([κ]α_i)$\wedge\wedge_{j=1,mi}$(<κ>β_{ij}))$\rightarrow$ ([([(RL $\kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i$)](mg $'\chi_i$))$\rightarrow$(mg $'\chi_i$))]χ_i)
Using the meaning laws M0-M7 gives: (([κ]α_i)$\wedge\wedge_{j=1,mi}$(<κ>β_{ij}))$\rightarrow$ ([([(RL $\kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i$)]$\chi_i$)$\rightarrow\chi_i$)]$\chi_i$)
Backchaining on χ_i shows that it suffices to prove:(([κ]α_i)$\wedge\wedge_{j=1,mi}$(<κ>β_{ij}))$\rightarrow$([(RL $\kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i$)]$\chi_i$)
which is an instance of theorem MR2. QED.

Finally MR5 and MR6 show that talking about the meanings of sets of FOL sentences in the modal representation of Reflective Logic is equivalent to talking about propositions in general.
MR5: (ss(RL $\kappa(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i$))$\equiv$(RL $\kappa(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i$) proof: In view of SS1, it suffices to prove
:([(ss(RL $\kappa(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i$))](RL $\kappa(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i$)). Unfolding the second occurrence of RL gives:[(ss(RL $\kappa(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i$))]($\forall i\Gamma_i\wedge\forall i$((([$\kappa$]$\alpha_i$)$\wedge\wedge_{j=1,mi}$<$\kappa$>$\beta_{ij}$)$\rightarrow\chi_i$) which holds by theorems MR3 and MR4. QED.

MR6: ($\kappa\equiv$(RL $\kappa(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i$))$\rightarrow\exists$s($\kappa\equiv$(ms s))
proof: From the hypothesis and MR5 $\kappa\equiv$(ss(RL $\kappa\ \forall i\Gamma_i\ \alpha_i{:}\beta_{ij}/\chi_i$)) is derived. Using the hypothesis to replace (RL $\kappa(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i$) by κ in this result gives: $\kappa\equiv$(ss(RL $\kappa(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i$)), By SS2 this implies the conclusion. QED.

Conclusion: The Relationship between Reflective Logic and the Modal Logic

The relationship between the proof theoretic definition of Reflective Logic [Brown 1989] and the modal representation is developed and proven in two steps. First theorem RL1 shows that the meaning of the set rl is the proposition RL and then theorem RL2 shows that a set of FOL sentences which contains its FOL theorems is a fixed-point of the fixed-point equation of Reflective Logic with an initial set of axioms and defaults if and only if the meaning (or rather disquotation) of that set of sentences is logically equivalent to RL of the meanings of that initial set of sentences and those defaults.

RL1: (ms(rl(fol $'\kappa$)$\{'\Gamma_i\}'\alpha_i{:}'\beta_{ij}/'\chi_i$))$\equiv$(RL(ms $'\kappa$)$(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i$))
proof: (ms(rl(fol $'\kappa$)$\{'\Gamma_i\}'\alpha_i{:}'\beta_{ij}/'\chi_i$))
Unfolding the definition of rl gives: ms(fol($\{'\Gamma_i\}\cup\{'\chi_i{:}('\alpha_i\varepsilon$(fol $'\kappa$))$\wedge\wedge_{j=1,mi}('(\neg\beta_{ij})\notin$(fol $'\kappa$))$\}$))
By C2 this is equivalent to: ms($\{'\Gamma_i\}\cup\{'\chi_i{:}('\alpha_i\varepsilon$(fol $'\kappa$))$\wedge\wedge_{j=1,mi}('(\neg\beta_{ij})\notin$(fol $'\kappa$))$\}$)
Using C1 twice gives: ms($\{'\Gamma_i\}\cup\{'\chi_i$: (([(ms $'\kappa$)](mg $'\alpha_i$)) $\wedge\wedge_{j=1,mi}\neg$([(ms $'\kappa$)](mg $'(\neg\beta_{ij})$))))$\}$)
Using MS3 gives: (ms $\{'\Gamma_i\}$)$\wedge$(ms$\{'\chi_i$: (([(ms $'\kappa$)](mg $'\alpha_i$)) $\wedge\wedge_{j=1,mi}\neg$([(ms $'\kappa$)](mg $'(\neg\beta_{ij})$))))$\}$)
Using MS2 gives:: ($\forall i\Gamma_i$)$\wedge$(ms$\{'\chi_i$: (([(ms $'\kappa$)](mg $'\alpha_i$)) $\wedge\wedge_{j=1,mi}\neg$([(ms $'\kappa$)](mg $'(\neg\beta_{ij})$))))$\}$)
Using MS1 gives: ($\forall i\Gamma_i$)$\wedge\forall i$((([(ms $'\kappa$)](mg $'\alpha_i$))$\wedge\wedge_{j=1,mi}\neg$([(ms $'\kappa$)](mg $'(\neg\beta_{ij})$)))$\rightarrow$(mg $'\chi_i$))
Using M0-M7 gives:: ($\forall i\Gamma_i$)$\wedge\forall i$((([(ms $'\kappa$)]α_i)$\wedge\wedge_{j=1,mi}\neg$([(ms $'\kappa$)]$\neg\beta_{ij}$))$\rightarrow\chi_i$)
Folding the definition of RL then gives: (RL(ms $'\kappa$)$(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i$) QED.

RL2: ((fol $'\kappa$)=(rl(fol $'\kappa$)$\{'\Gamma_i\}$ $'\alpha_i{:}'\beta_{ij}/'\chi_i$))$\leftrightarrow$((ms $'\kappa$)$\equiv$(RL(ms $'\kappa$)$(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i$)))
proof: (fol $'\kappa$)=(rl(fol $'\kappa$)$\{'\Gamma_i\}$ $'\alpha_i{:}'\beta_{ij}/'\chi_i$) By FOL3 this is equivalent to: (fol $'\kappa$)=(fol(rl(fol $'\kappa$)$\{'\Gamma_i\}$ $'\alpha_i{:}'\beta_{ij}/'\chi_i$))

By C4 this is equivalent to: ((ms 'κ)≡(ms(rl(fol 'κ){'Γ_i} 'α_i:'β_{ij}/'χ_i)))

By RL1 this is equivalent to: (ms 'κ)≡(RL(ms 'κ)($\forall i\Gamma_i$) α_i:β_{ij}/χ_i)) QED.

Theorem RL2 shows that the set of theorems: (fol 'κ) of a set 'κ is a fixed-point of a fixed point equation of Reflective Logic if and only if the meaning (ms 'κ) of 'κ: is a solution to the necessary equivalence. Furthermore, by FOL4 there are no other fixed-points (such as a set not containing all its theorems) and by MR6 there are no other solutions (such as a proposition not representable as a sentence in the FOL object language). Therefore, the Modal representation of Reflective Logic (i.e. RL), faithfully represents the set theoretic description of Reflective Logic (i.e. rl).Finally, we note that $\forall i\Gamma_i$ and (ms 'κ) may be generalized to be arbitrary propositions Γ and κ giving the more general modal representation: κ≡(RL κ Γ α_i:β_{ij}/χ_i).

Acknowledgements

This research was supported by National Science Foundation grants: #9818341 and #9972843.

Bibliography

[Anatoniou 1997] Antoniou, Grigoris 1997. *NonMonotonic Reasoning*, MIT Press.

[Bressan 1972] Bressan, Aldo 1972. *A General Interpreted Modal Calculus,* Yale University Press.

[Brown 1978] Brown,F.M., "A Semantic Theory for Logic Programming", *Colloquia Mathematica Societatis Janos Bolyai 26*, *Mathematical Logic in Computer Science*, Salgotarjan, Hungry, 1978.

[Boyer&Moore 1981] R. S. Boyer and J. Strother Moore, "Metafunctions: proving them correct and using them efficiently as new proof procedures," *The Correctness Problem in Computer Science,* R. S. Boyer and J. Strother Moore, eds., Academic Press, New York, 1981.

[Brown 1986] Brown, Frank M. 1986. "Reasoning in a Hierarchy of Deontic Defaults", *Proceedings of the Canadian Artificial Intelligence Conference CSCI 86*, Montreal, Canada, Morgan-Kaufmann, Los Altos.

[Brown 1987] Brown, Frank M. 1987. "The Modal Logic Z", In *The Frame Problem in AI"; Proc. of the 1987 AAAI Workshop*, Morgan Kaufmann, Los Altos, CA .

[Brown 1989] Brown, Frank M. 1989. "The Modal Quantificational Logic Z Applied to the Frame Problem", advanced paper *First International Workshop on Human & Machine Cognition*, May 1989 Pensacola, Florida. Abbreviated version published in *International Journal of Expert Systems Research and Applications, Special Issue: The Frame Problem. Part A.* eds. Keneth Ford and Pattrick Hayes, vol. 3 number 3, pp169-206 JAI Press 1990. Reprinted in *Reasoning Agents in a Dynamic World: The Frame problem*, editors: Kenneth M. Ford, Patrick J. Hayes, JAI Press 1991.

[Carnap 1946] Carnap, Rudolf 1946. "Modalities and Quantification" *Journal of Symbolic Logic*, vol. 11, number 2.

[Carnap 1956] Carnap, Rudolf 1956. *Meaning and Necessity: A Study in the Semantics of Modal Logic*, The University of Chicago Press.

[Fine 1970] Fine, K. 1970. "Propositional Quantifiers in Modal Logic" *Theoria* 36, p336--346.

[Hendry & Pokriefka 1985] Hendry, Herbert E. and Pokriefka, M. L. 1985. "Carnapian Extensions of S5", *Journal of Phil. Logic 14*.

[Hughes & Cresswell 1968] Hughes, G. E. & Creswell, M. J., 1968. *An Introduction to Modal Logic*, Methuen & Co. Ltd., London.

[Leasure 1993] Leasure, David E., 1993. The Modal Logic Z Applied to Lifschitz's Benchmark problems for Formal Nonmonotonic Reasoning, University of Kansas Dissertation, University of Kansas Library.

[Lewis 1936] Lewis, C. I. 1936. Strict Implication, *Journal of Symbolic Logic*, vol I.

[Mendelson 1964] Mendelson, E. 1964. *Introduction to Mathematical Logic*, Van Norstrand, Reinhold Co., New York.

[Moore 1985] Moore, R. C. 1985. "Semantical Considerations on Nonmonotonic Logic" *Artificial Intelligence*, 25.

[Parks 1976] Parks, Z. 1976. "An Investigation into Quantified Modal Logic", Studia Logica 35, p109-125.

[Quine 1969] Quine, W.V.O., Set Theory and Its Logic, revised edition, Oxford University Press, London, 1969.

[Reiter 1980] Reiter, R. 1980. "A Logic for Default Reasoning" *Artificial Intelligence*, 13.

Author information

Frank M. Brown- Artificial Intelligence Laboratory, University of Kansas, Lawrence, Kansas, 66045, e-mail: brown@ku.edu.

REPRESENTING DEFAULT LOGIC IN MODAL LOGIC

Frank M. Brown

***Abstract**: The nonmonotonic logic called Default Logic is shown to be representable in a monotonic Modal Quantificational Logic whose modal laws are stronger than S5. Specifically, it is proven that a set of sentences of First Order Logic is a fixedpoint of the fixedpoint equation of Default Logic with an initial set of axioms and defaults if and only if the meaning or rather disquotation of that set of sentences is logically equivalent to a particular modal functor of the meanings of that initial set of sentences and of the sentences in those defaults. This result is important because the modal representation allows the use of powerful automatic deduction systems for Modal Logic and because unlike the original Default Logic, it is easily generalized to the case where quantified variables may be shared across the scope of the components of the defaults thus allowing such defaults to produce quantified consequences. Furthermore, this generalization properly treats such quantifiers since both the Barcan Formula and its converse hold.*

***Keywords**: Default Logic, Modal Logic, Nonmonotonic Logic.*

1. Introduction

One of the most well known nonmonotonic logics [Antoniou 1997] which inherently deals with entailment conditions in addition to possibility conditions in its defaults is the so-called Default Logic [Reiter 1980]. The basic idea of Default Logic is that there is a set of axioms Γ and some non-logical default "inference rules" of the form:

$$\frac{\alpha : \beta_1 ... \beta_m}{\chi}$$

which suggest that χ may be inferred from α whenever each $\beta_1,...,\beta_m$ is consistent with everything that is inferable. Such "inference rules" are not recursive and are circular in that the determination as to whether χ is derivable depends on whether β_i is consistent which inturn depends on what was derivable from this and other defaults. Thus, tentatively applying such inference rules by checking the consistency of $\beta_1,...,\beta_m$ with only the current set of inferences produces a χ result which may later have to be retracted. For this reason, valid inferences in a nonmonotonic logic such as Default Logic are essentially carried out not in the original nonmonotonic logic, but rather in some (monotonic) metatheory in which that nonmonotonic logic is monotonically defined. [Reiter 1980] explicated this intuition by defining Default Logic in terms of the set theoretic proof theory metalanguage of First Order Logic (i.e. FOL) with the following fixed point expression: $'\kappa = (dl\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i)$

where dl is: $(dl\ '\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i) =_{df} \cap\{p: (p \supseteq (fol\ p)) \wedge (p \supseteq '\Gamma) \wedge \forall_i(((' \alpha_i \varepsilon p) \wedge \wedge_{j=1,mi}('(\neg\beta_{ij}) \notin '\kappa)) \rightarrow (' \chi_i \varepsilon p))\}$

where $'\alpha_i$, $'\beta_{ij}$, and $'\chi_i$ are the closed sentences of FOL occurring in the ith default "inference rule" and $'\Gamma$ is a set of closed sentences of FOL. A closed sentence is a sentence without any free variables. fol is a function which produces the set of theorems derivable in FOL from the set of sentences to which it is applied. The quotations appended to the front of these Greek letters indicate references in the metalanguage to the sentences of the FOL object language. Interpreted doxastically this fixed point equation states:

> The set of closed sentences which are believed is equal to
> the intersection of all sets of closed sentences which are potentialially believed such that:
> the closed sentences derived by the laws of FOL from the potential beliefs are themselves potentially believed,
> the closed sentences in $'\Gamma$ are potentially believed,
> and for each i,
> if the closed sentence $'\alpha_i$ is potentially believed and for each j, the closed sentence $'\beta_{ij}$ is believable
> then the closed sentence $'\chi_i$ is potentially believed.

The purpose of this paper is to show that all this metatheoretic machinery including the formalized syntax of FOL, the proof theory of FOL, the axioms of a strong set theory, and the set theoretic fixedpoint equation is not needed and that the essence of Default Logic in representable as a necessary equivalence in a simple (monotonic) Modal Quantificational Logic. Interpreted as a doxastic logic this necessary equivalence states:

> That which is believed is logically equivalent to
> some potential belief such that:
> Γ is potentially believed
> and for each i, if α_i is potentially believed and for each j, β_{ij} is believable then χ_i is potentially believed.

thereby eliminating all mention of any metatheoretic machinery.
The remainder of this paper proves that this modal representation is equivalent to Default Logic. Section 2 describes a formalized syntax for a FOL object language. Section 3 describes the part of the proof theory of FOL needed herein (i.e. theorems FOL1-FOL9). Section 4 describes the Intensional Semantics of FOL including the meaning operator (i.e. the laws M0-M7) and the relationship of meaning and modality to the proof theory of FOL (i.e. the laws R0, A1, A2 and A3 and the theorems C1, C2, C3, and C4). The modal version of Default Logic, called DL, is defined in section 5 and explicated with theorems MD1-MD7 and SS1-SS2. In section 6, this modal version is shown by theorems DL1 and DL2 to be equivalent to the set theoretic fixed-point equation for Default Logic. Figure 1 outlines the relationship of all these theorems to the final theorems DL2, FOL9, and MD7.

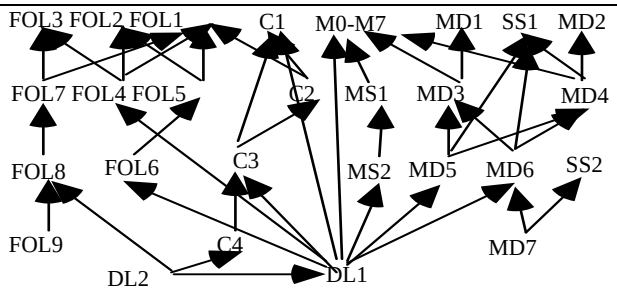


Figure 1: Dependencies among the Theorems

2. Formal Syntax of First Order Logic

We use a First Order Logic (i.e. FOL) defined as the six tuple: ($\rightarrow$, #f, $\forall$, *vars*, *predicates*, *functions*) where $\rightarrow$, #f, and $\forall$ are logical symbols, *vars* is a set of variable symbols, *predicates* is a set of predicate symbols each of which has an implicit arity specifying the number of associated terms, and *functions* is a set of function symbols each of which has an implicit arity specifying the number of associated terms. The sets of logical symbols, variables, predicate symbols, and function symbols are pairwise disjoint. Lower case Roman letters possibly indexed with digits are used as variables. Greek letters possibly indexed with digits or lower case roman letters are used as syntactic metavariables. γ, γ_1,...γ_n, range over the variables, ξ, ξ_1...ξ_n range over sequences of variables of an appropriate arity, π,π_1...π_n range over the predicate symbols, ϕ, ϕ_1...ϕ_n range over function symbols, δ, δ_1...δ_n, σ range over terms, and α,α_1...α_n, β, β_1...β_n,χ, χ_1...χ_n, Γ_1,...Γ_n,φ range over sentences. The terms are of the forms γ and (ϕ δ1...δn), and the sentences are of the forms ($\alpha\rightarrow\beta$), #f, ($\forall\gamma$ α), and (π δ_1...δ_n). A nullary predicate π or function ϕ is written as a sentence or a term without parentheses. $\varphi\{\pi/\lambda\xi\alpha\}$ represents the replacement of all occurrences of π in φ by $\lambda\xi\alpha$ followed by lambda conversion. The primitive symbols are shown in Figure 2 with their intuitive interpretations.

Symbol	Meaning
$\alpha\rightarrow\beta$	if α then β.
#f	falsity
$\forall\gamma$ α	for all γ, α.

Figure 2: Primitive Symbols of First Order Logic

The defined symbols are listed in Figure 3 with their definitions and intuitive interpretations.

Symbol	Definition	Meaning		Symbol	Definition	Meaning
$\neg\alpha$	$\alpha \rightarrow$ #f	not α		$\alpha\wedge\beta$	$\neg(\alpha \rightarrow \neg\beta)$	α and β
#t	$\neg$ #f	truth		$\alpha\leftrightarrow\beta$	$(\alpha\rightarrow\beta)\wedge(\beta\rightarrow\alpha)$	α if and only if β
$\alpha\vee\beta$	$(\neg\alpha)\rightarrow\beta$	α or β		$\exists\gamma\ \alpha$	$\neg\forall\gamma\ \neg\alpha$	for some γ , α

Figure 3: Defined Symbols of First Order Logic

The FOL object language expressions are referred in the metalanguage (which also includes a FOL syntax) by inserting a quote sign in front of the object language entity thereby making a structural descriptive name of that entity. A set of sentences is represented as: $\{'\Gamma_i\}$ which is defined as: $\{'\Gamma_i: \#t\}$ which in turn is defined as: $\{s: \exists i(s='\Gamma_i)\}$ where i ranges over some range of numbers (which may be finite or non-infinite). With a slight abuse of notation we also write $'\kappa$, $'\Gamma$ to refer to such sets.

3. Proof Theory of First Order Logic

First Order Logic (i.e. FOL) is axiomatized with a recursively enumerable set of theorems as the set of axioms is itself recursively enumerable and its inference rules are recursive. The axioms and inference rules of FOL [Mendelson 1964] are those given in Figure 4. They form a standard set of axioms and inference rules for FOL.

MA1: $\alpha \rightarrow (\beta\rightarrow \alpha)$ MR1: from α and $(\alpha\rightarrow \beta)$ infer β

MA2: $(\alpha\rightarrow (\beta\rightarrow \rho)) \rightarrow ((\alpha\rightarrow \beta)\rightarrow (\alpha\rightarrow \rho))$ MR2: from α infer $(\forall\gamma\ \alpha)$

MA3: $((\neg\alpha)\rightarrow(\neg\beta))\rightarrow(((\neg\alpha)\rightarrow\beta)\rightarrow\alpha)$

MA4: $(\forall\gamma\ \alpha)\rightarrow\beta$ where β is the result of substituting an expression (which is free for the free positions of γ in α) for all the free occurrences of γ in α.

MA5: $((\forall\gamma(\alpha \rightarrow \beta)) \rightarrow (\alpha\rightarrow(\forall\gamma\ \beta)))$ where γ does not occur in α.

Figure 4: Inferences Rules and Axioms of FOL

In order to talk about sets of sentences we include in the metatheory set theory symbolism as developed along the lines of [Quine 1976]. This set theory includes the symbols ε, $\notin$, $\supseteq$, $=$, $\cup$ as is defined therein.

The derivation operation (i.e. fol) of any First Order Logic obeys the Inclusion (i.e. FOL1), Idempotence (i.e. FOL2), and Monotonic (i.e. FOL3) properties:

FOL1: $(\text{fol } '\kappa)\supseteq'\kappa$ Inclusion

FOL2: $(\text{fol } '\kappa)\supseteq(\text{fol}(\text{fol } '\kappa))$ Idempotence

FOL3: $('\kappa\supseteq'\Gamma) \rightarrow ((\text{fol } '\kappa)\supseteq (\text{fol } '\Gamma))$ Monotonicity

From these three properties we prove the following theorems of the proof theory of First Order Logic:

FOL4 $((\text{fol } '\kappa)\supseteq(\text{fol } '\Gamma))\leftrightarrow((\text{fol } '\kappa)\supseteq'\Gamma)$ proof: The proof divides into two parts: (1) $((\text{fol } '\kappa)\supseteq(\text{fol } '\Gamma)) \rightarrow ((\text{fol } '\kappa)\supseteq'\Gamma)$. By FOL1 the hypothesis implies the conclusion. (2) $((\text{fol } '\kappa)\supseteq'\Gamma)\rightarrow((\text{fol } '\kappa)\supseteq(\text{fol } '\Gamma))$ By FOL3 the hypothesis implies $(\text{fol}(\text{fol } '\kappa))\supseteq(\text{fol } '\Gamma)$ which by FOL2 implies the conclusion. QED.

FOL5: $\forall p((p=(\text{fol } p))\rightarrow\alpha)\leftrightarrow\forall p(\alpha\{p/(\text{fol } p)\})$ and $\exists p((p=(\text{fol } p))\wedge\alpha)\leftrightarrow\exists p(\alpha\{p/(\text{fol } p)\})$

proof: The universal quantifier version follows from the existential quantifier version by running negation through both sides of the bi-implication. The existential version is proven as follows. There are two cases:

(1) $((p=(\text{fol } p))\wedge\alpha)\rightarrow \exists p(\alpha\{p/(\text{fol } p)\})$. The existentially quantified p is replaced by p giving:

$((p=(\text{fol } p))\wedge\alpha)\rightarrow(\alpha\{p/(\text{fol } p)\})$ The the hypothesis is used to replace p in α by (fol p) giving the conclusion.

(2) $(\alpha\{p/(\text{fol } p)\})\rightarrow \exists p((p=(\text{fol } p))\wedge\alpha)$ Letting p in the conclusion be (fol p) gives:

$(\alpha\{p/(\text{fol } p)\})\rightarrow (((\text{fol } p)=(\text{fol}(\text{fol } p)))\wedge(\alpha\{p/(\text{fol } p)\}))$ which holds by FOL1 and FOL2.

FOL6: $(\cap\{p: (p\supseteq(\text{fol } p))\wedge\varphi\})=\{s:\forall p((\varphi\{p/(\text{fol } p)\})\rightarrow(s\varepsilon(\text{fol } p)))\}$ proof: $\cap\{p: (p\supseteq(\text{fol } p))\wedge\varphi\}$ By FOL1 this is equivalent to: $\cap\{p: (p=(\text{fol } p))\wedge\varphi\}$. Unfolding the definition of intersection gives:$\{s:\forall p((p\varepsilon\{p: (p=(\text{fol } p))\wedge\varphi\})\rightarrow(s\varepsilon p))\}$ which is equivalent to: $\{s:\forall p(((p=(\text{fol } p))\wedge\varphi)\rightarrow(s\varepsilon p))\}$. By FOL5 this is equivalent to: $\{s:\forall p((\varphi\{p/(\text{fol } p)\})\rightarrow(s\varepsilon(\text{fol } p)))\}$ QED.

FOL7: If α is a sentence of proof theory then: $(\cap\{p: (p\supseteq(\text{fol } p))\wedge\alpha\})=(\text{fol}(\cap\{p: (p\supseteq(\text{fol } p))\wedge\alpha\}))$

proof: From FOL1 it suffices to prove: $(s\varepsilon(\text{fol}(\cap\{p: (p\supseteq(\text{fol } p))\wedge\alpha\})))\rightarrow(s\varepsilon(\cap\{p: (p\supseteq(\text{fol } p))\wedge\alpha\}))$. Unfolding the intersections and simplifying gives: $(s\varepsilon(\text{fol}\{s: \forall p(((p\supseteq(\text{fol } p))\wedge\alpha)\rightarrow(s\varepsilon p))\}))\rightarrow\forall p(((p\supseteq(\text{fol } p))\wedge\alpha)\rightarrow(s\varepsilon p))$which is equivalent to: $((s\varepsilon(\text{fol}\{s:(s\varepsilon p)\wedge\forall p(((p\supseteq(\text{fol}p))\wedge\alpha)\rightarrow(s\varepsilon p))\}))\wedge(p\supseteq(\text{fol } p))\wedge\alpha)\rightarrow(s\varepsilon p)$. Folding intersection then gives: $((s\varepsilon(\text{fol}(\{s:(s\varepsilon p)\}\cap\{s:\forall p(((p\supseteq(\text{fol}p))\wedge\alpha)\rightarrow(s\varepsilon p))\})))\wedge(p\supseteq(\text{fol } p))\wedge\alpha)\rightarrow(s\varepsilon p)$. Using the second hypothesis to replace p by (fol p) and then dropping the second and third hypotheses gives: $(s\varepsilon(\text{fol}(p\cap\{s:\forall p(((p\supseteq(\text{fol}p))\wedge\alpha)\rightarrow(s\varepsilon p))\})))\rightarrow(s\varepsilon(\text{fol } p))$. Folding $\supseteq$ gives: $(\text{fol } p)\supseteq(\text{fol}(p\cap\{s:\forall p(((p\supseteq(\text{fol } p))\wedge\alpha)\rightarrow(s\varepsilon p))\}))$. Generalizing, it suffices to prove for all α: $(\text{fol } p)\supseteq(\text{fol}(p\cap\alpha))$. Since $p\supseteq(p\cap\alpha)$ this follows by FOL3. QED.

FOL8: $(\text{dl }'\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i)=(\text{fol}(\text{dl }'\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i))$ proof: Unfolding dl gives:$\cap\{p: (p\supseteq(\text{fol } p))\wedge(p\supseteq'\Gamma)\wedge\forall i(((\text{'}\alpha_i\varepsilon p)\wedge\wedge_{j=1,m_i}(\text{'}(\neg\beta_{ij})\notin'\kappa))\rightarrow(\text{'}\chi_i\varepsilon p))\}$. By FOL7 this is equivalent to: $\text{fol}(\cap\{p: (p\supseteq(\text{fol } p))\wedge(p\supseteq'\Gamma)\wedge\forall i(((\text{'}\alpha_i\varepsilon p)\wedge\wedge_{j=1,m_i}(\text{'}(\neg\beta_{ij})\notin'\kappa))\rightarrow(\text{'}\chi_i\varepsilon p))\})$ Folding dl then proves the theorem: $\text{fol}(\text{dl }'\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i)$ QED.

FOL9: $(\text{'k}=(\text{dl }'\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i))\rightarrow(\text{'}\kappa=(\text{fol }'\kappa))$ proof: From the hypothesis and FOL8 $\text{'k}=(\text{fol}(\text{dl }'\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i))$ is derived. Using the hypothesis to replace $(\text{dl }'\kappa\ '\Gamma\ '\alpha_i{:}'\beta_{ij}/'\chi_i)$ by $'\kappa$ in this result gives: $(\text{'}\kappa=(\text{fol }'\kappa))$ QED.

4. Intensional Semantics of FOL

The meaning (i.e. mg) [Brown 1978, Boyer&Moore 1981] or rather disquotation of a sentence of First Order Logic (i.e. FOL) is defined to satisfy the laws given in Figure 5 below mg is defined in terms of mgs which maps each FOL object language sentence and an association list into a meaning. Likewise, mgn maps a FOL object language term and an association list into a meanings. An association list is simply a list of pairs consisting of an object language variable and the meaning to which it is bound.

M0: $(\text{mg }'\alpha)$ =df $(\text{mgs }'(\forall\gamma1...\gamma n\ \alpha)'())$ where $'\gamma1...'\gamma n$ are all the free variables in $'\alpha$

M1: $(\text{mgs }'(\alpha\rightarrow\beta)a)\leftrightarrow((\text{mgs }'\alpha\ a)\rightarrow(\text{mgs }'\beta\ a))$

M2: $(\text{mgs }'\#f\ a)\leftrightarrow\#f$

M3: $(\text{mgs }'(\forall\ \gamma\ \alpha)a)\leftrightarrow\forall x(\text{mgs }'\alpha(\text{cons}(\text{cons }'\gamma\ x)a))$

M4: $(\text{mgs }'(\pi\ \delta1...\delta n)a)\leftrightarrow(\pi(\text{mgn }'\delta1\ a)...(\text{mgn }'\delta n\ a))$ for each predicate symbol $'\pi$.

M5: $(\text{mgn }'(\phi\ \delta1...\delta n)a)=(\phi(\text{mgn }'\delta1\ a)...(\text{mgn }'\delta n\ a))$ for each function symbol $'\phi$.

M6: $(\text{mgn }'\gamma\ a)=(\text{lookup }'\gamma\ a)$ where $(\text{lookup }'\gamma\ a)$ is the value associated wit $'\gamma$ in the association list a.

Figure 5: The Meaning of FOL Sentences

The meaning of a set of sentences is defined in terms of the meanings of the sentences in the set as:

$(\text{ms }'\kappa)$ =df $\forall s((s\varepsilon'\kappa)\rightarrow(\text{mg } s))$

MS1: $(\text{ms}\{'\alpha:\Gamma\})\leftrightarrow\forall\xi(\Gamma\rightarrow\alpha)$ where ξ is the sequence of all the free variables in $'\alpha$ and where Γ is any sentence of the intensional semantics. proof: $(\text{ms}\{'\alpha{:}\Gamma\})$ Unfolding ms and the set pattern abstraction symbol gives: $\forall s((s\varepsilon\{s:\exists\xi((s='\alpha)\wedge\Gamma)\})\rightarrow(\text{mg } s))$ where ξ is a sequence of the free variables in 'a. This is equivalent to: $\forall s((\exists\xi((s='\alpha)\wedge\Gamma)))\rightarrow(\text{mg } s))$ which is logically equivalent to: $\forall s\forall\xi\ (((s='\kappa)\wedge\Gamma)\rightarrow(\text{mg } s))$ which is equivalent to: $\forall\xi(\Gamma\rightarrow(\text{mg }'\alpha))$ Unfolding mg using M0-M7 then gives: $\forall\xi(\Gamma\rightarrow\alpha)$ QED

The meaning of the union of two sets of FOL sentences is the conjunction of their meanings (i.e. MS1) and the meaning of a set is the meaning of all the sentences in the set (i.e. MS2):

MS2: $(\text{ms}\{'\Gamma_i\})\leftrightarrow\forall i\forall\xi_i\Gamma_i$ proof: $(\text{ms}\{'\Gamma_i\})$ Unfolding the set notation gives: $(\text{ms}\{'\Gamma_i{:}\ \#t\})$

By MS1 this is equivalent to: $\forall_i \forall \xi_i (\#t \rightarrow \Gamma_i)$ which is equivalent to: $\forall i \Gamma_i$ QED.

MS3: $(ms('\kappa \cup '\Gamma)) \leftrightarrow ((ms\ '\kappa) \wedge (ms\ '\Gamma))$ proof: Unfolding ms and union in: $(ms('\kappa \cup '\Gamma))$ gives: $\forall s((s\varepsilon\{s: (s\varepsilon'\kappa) \vee (s\varepsilon'\Gamma)\}) \rightarrow (mg\ s))$ or rather: $\forall s(((s\varepsilon'\kappa) \vee (s\varepsilon'\Gamma)) \rightarrow (mg\ s))$ which is logically equivalent to: $(\forall \alpha((s\varepsilon'\kappa) \rightarrow (mg\ s))) \wedge (\forall s((s\varepsilon'\Gamma) \rightarrow (mg\ s)))$. Folding ms twice then gives:$((ms\ '\kappa) \wedge (ms\ '\Gamma))$ QED.

The meaning operation may be used to develop an Intensional Semantics for a FOL object language by axiomatizing the modal concept of necessity so that it satisfies the theorem:

C1: $('\alpha\varepsilon(fol\ '\kappa)) \leftrightarrow ([]\ ((ms\ '\kappa) \rightarrow (mg\ '\alpha)))$

for every sentence $'\alpha$ and every set of sentences $'\kappa$ of that FOL object language. The necessity symbol is represented by a box: []. C1 states that a sentence of FOL is a FOL-theorem (i.e. fol) of a set of sentences of FOL if and only if the meaning of that set of sentences necessarily implies the meaning of that sentence. One modal logic which satisfies C1 is the Z Modal Quantificational Logic described in [Brown 1987; Brown 1989] whose theorems are recursively enumerable and which extends the weaker possibility axioms used in [Lewis 1936; Bressan 1972; Hendry & Pokriefka 1985]. Z includes all the laws of S5 modal Logic [Hughes & Cresswell 1968] whose laws are given in Figure 6. κ and Γ represent arbitrary sentences of the intentional semantics.

R0: from α infer $([]\ \kappa)$ A2: $([](\kappa \rightarrow \Gamma)) \rightarrow (([]\kappa) \rightarrow ([]\Gamma))$

A1: $([]\kappa) \rightarrow \kappa$ A3: $([]\kappa) \vee ([]\neg[]\kappa)$

Figure 6: The Laws of S5 Modal Logic

These S5 modal laws and the laws of FOL given in Figure 4 constitute an S5 Modal Quantificational Logic similar to [Carnap 1946; Carnap 1956], and a FOL version [Parks 1976] of [Bressan 1972] in which the Barcan formula: $(\forall \gamma([]\kappa)) \rightarrow ([]\forall \gamma \kappa)$ and its converse hold. The R0 inference rule implies that anything derivable in the metatheory is necessary. Thus, in any logic with R0, contingent facts would never be asserted as additional axioms of the metatheory. The defined Modal symbols used herein are listed in Figure 7.

Symbol	Definition	Meaning		Symbol	Definition	Meaning
$<>\kappa$	$\neg [] \neg \kappa$	α is logically possible		$[\kappa]\ \Gamma$	$[]\ (\kappa \rightarrow \Gamma)$	β entails α
$\kappa \equiv \Gamma$	$[]\ (\kappa \leftrightarrow \Gamma)$	α is logically equivalent to β		$<\kappa>\ \Gamma$	$<>\ (\kappa \wedge \Gamma)$	α and β is logically possible

Figure 7: Defined Symbols of Modal Logic

From the laws of the Intensional Semantics we prove that the meaning of the set of FOL consequences of a set of sentences is the meaning of that set of sentences (C2), the FOL consequences of a set of sentences contain the FOL consequences of another set if and only if the meaning of the first set entails the meaning of the second set (C3), and the sets of FOL consequences of two sets of sentences are equal if and only if the meanings of the two sets are logically equivalent (C4):

C2: $(ms(fol\ '\kappa)) \equiv (ms\ '\kappa)$ proof: The proof divides into two cases: (1) $[(ms\ '\kappa)](ms(fol\ '\kappa))$. Unfolding the second ms gives: $[(ms\ '\kappa)] \forall s((s\varepsilon(fol\ '\kappa)) \rightarrow (mg\ s))$. By the soundness part of C1 this is equivalent to: $[(ms\ '\kappa)] \forall s(([(ms\ '\kappa)](mg\ s)) \rightarrow (mg\ s))$. By the S5 laws this is e: $\forall s(([(ms\ '\kappa)](mg\ s)) \rightarrow [(ms\ '\kappa)](mg\ s))$ which is a tautology.

(2) $[(ms(fol\ '\kappa))](ms\ '\kappa)$ Unfolding ms twice gives: $[\forall s((s\varepsilon(fol\ '\kappa)) \rightarrow (mg\ s))] \forall s((s\varepsilon'\kappa) \rightarrow (mg\ s))$

which is: $[\forall s((s\varepsilon(fol\ '\kappa)) \rightarrow (mg\ s))]((s\varepsilon'\kappa) \rightarrow (mg\ s))$ Backchaining on the hypothesis and then dropping it gives: $(s\varepsilon'\kappa) \rightarrow (s\varepsilon(fol\ '\kappa))$. Folding $\supseteq$ gives an instance of FOL1. QED.

C3: $(fol\ '\kappa) \supseteq (fol\ '\Gamma) \leftrightarrow ([(ms\ '\kappa)](ms\ '\Gamma))$

proof: Unfolding $\supseteq$ gives: $\forall s((s\varepsilon(fol\ '\Gamma)) \rightarrow (s\varepsilon(fol\ '\kappa)))$.By C1 twice this is:$\forall s(([(ms\ '\Gamma)](mg\ s)) \rightarrow ([(ms\ '\kappa)](mg\ s)))$

By the laws of S5 modal logic this is equivalent to: $([(ms\ '\kappa)] \forall s(([(ms\ '\Gamma)](mg\ s))) \rightarrow (mg\ s)))$. By C1 this is: $[(ms\ '\kappa)] \forall s((s\varepsilon(fol\ '\Gamma)) \rightarrow (mg\ s))$. Folding ms then gives: $[(ms\ '\kappa)](ms(fol\ '\Gamma))$. By C2 this is : $[(ms\ '\kappa)](ms\ '\Gamma)$. QED.

C4: $((fol\ '\kappa) = (fol\ '\Gamma)) \leftrightarrow ((ms\ '\kappa) \equiv (ms\ '\Gamma))$ proof: This is equivalent to $(((fol\ '\kappa) \supseteq (fol\ '\Gamma)) \wedge ((fol\ '\Gamma) \supseteq (fol\ '\kappa))) \leftrightarrow ([(ms\ '\kappa)](ms\ '\Gamma)) \wedge ([(ms\ '\Gamma)](ms\ '\kappa))$ which follows by using C3 twice.

5. Default Logic Represented in Modal Logic

The fixed point equation for Default Logic may be expressed as a necessary equivalence in an S5 Modal Quantificational Logic supplemented with propositional quantifiers [Fine 1970; Bressan 1972] which obey the normal laws of Second Order Logic (i.e. laws analogous to MR2, MA4, and MA5 given in Figure 4 where γ is now a propositional variable), as follows: $\kappa \equiv (DL\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)$

where DL is defined as: $(DL\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i) =_{df} \exists p(p \wedge ([p]\Gamma) \wedge \forall_i((([p]\alpha_i) \wedge \wedge_{j=1,mi}(<\kappa>\beta_{ij})) \rightarrow [p]\chi_i))$

where the propositional variable p does not occur in Γ, α_i, β_{ij}, and χ_i. When the context is obvious $\Gamma\ \alpha_i{:}\beta_{ij}/\chi_i$ is omitted and just $(DL\ \kappa)$ is written. The idiom $\exists p(p \wedge \varphi)$ may be intuitively read as a nominal as the (possibly infinite) disjunction of all propositions such that φ. When φ holds for only a finite number of propositions: $\varphi_1,...,\varphi_n$ then $\exists p(p \wedge \varphi))$ is equivalent to: $\varphi_1 \vee ... \vee \varphi_n$, but there is in no requirement that φ holds for only a finite or even only a denumerable number of propositions.

The first two theorems state that DL entails Γ and any conclusion χ_i of a default whose entailment condition holds in DL and whose possible conditions are possible with κ.

MD1: $[(DL\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)]\Gamma$

proof: Unfolding DL gives: $[\exists p(p \wedge ([p]\Gamma) \wedge \forall_i((([p]\alpha_i) \wedge \wedge_{j=1,mi}(<\kappa>\beta_{ij})) \rightarrow ([p]\chi_i)))]\Gamma$. Since p is not free in Γ, pulling $\exists p$ out of the hypothesis of the entailment gives:
$\forall p((([p]\Gamma) \wedge \forall_i((([p]\alpha_i) \wedge \wedge_{j=1,mi}<\kappa>\beta_{ij}) \rightarrow ([p]\chi_i))) \rightarrow ([p]\Gamma))$ which is a tautology. QED.

MD2: $(([(DL\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)]\alpha_i) \wedge \wedge_{j=1,mi}(<\kappa>\beta_{ij})) \rightarrow ([(DL\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)]\chi_i)$

proof: Unfolding both occurrences of DL gives:

$(([\exists p(p \wedge ([p]\Gamma) \wedge \forall_i((([p]\alpha_i) \wedge \wedge_{j=1,mi}<\kappa>\beta_{ij}) \rightarrow ([p]\chi_i)))]\alpha_i) \wedge (\wedge_{j=1,mi}(<\kappa>\beta_{ij})))$

$\rightarrow ([\exists p(p \wedge ([p]\Gamma) \wedge \forall_i((([p]\alpha_i) \wedge \wedge_{j=1,mi}(<\kappa>\beta_{ij})) \rightarrow ([p]\chi_i)))]\chi_i)$

Since p is not free in α_i and χ_i, pulling $\exists p$ out of the hypotheses of the outer two entailments gives:

$((\forall p((([p]\Gamma) \wedge \forall_i((([p]\alpha_i) \wedge \wedge_{j=1,mi}(<\kappa>\beta_{ij})) \rightarrow ([p]\chi_i))) \rightarrow ([p]\alpha_i)) \wedge (\wedge_{j=1,mi}(<\kappa>\beta_{ij})))$

$\rightarrow \forall p((([p]\Gamma) \wedge \wedge_i((([p]\alpha_i) \wedge \wedge_{j=1,mi}(<\kappa>\beta_{ij})) \rightarrow ([p]\chi_i))) \rightarrow ([p]\chi_i))$

Instantiating the p in the hypothesis to the p in the conclusion gives:

$(((([p]\Gamma) \wedge \forall_i((([p]\alpha_i) \wedge \wedge_{j=1,mi}<\kappa>\beta_{ij}) \rightarrow ([p]\chi_i))) \rightarrow ([p]\alpha_i)) \wedge (\wedge_{j=1,mi}(<\kappa>\beta_{ij})) \wedge ([p]\Gamma) \wedge \forall_i((([p]\alpha_i) \wedge \wedge_{j=1,mi}(<\kappa>\beta_{ij})) \rightarrow ([p]\chi_i))) \rightarrow ([p]\chi_i)$

which simplifies to just: $(([p]\alpha_i) \wedge \wedge_{j=1,mi}(<\kappa>\beta_{ij})) \wedge ([p]\Gamma) \wedge \forall_i((([p]\alpha_i) \wedge \wedge_{j=1,mi}(<\kappa>\beta_{ij})) \rightarrow ([p]\chi_i))) \rightarrow ([p]\chi_i)$

Forward chaining using the first and second hypotheses on the fourth proves the theorem. QED.

The concept (i.e. ss) of the combined meaning of all the sentences of the FOL object language whose meanings are entailed by a proposition is defined as follows: $(ss\ \kappa) =_{df} \forall s(([\kappa](mg\ s)) \rightarrow (mg\ s))$. SS1 shows that a proposition entails the combined meaning of the FOL object language sentences that it entails. SS2 shows that if a proposition is necessarily equivalent to the combined meaning of the FOL object language sentences that it entails, then there exists a set of FOL object language sentences whose meaning is necessarily equivalent it:

SS1: $[\kappa](ss\ \kappa)$

proof: By R0 it suffices to prove: $\kappa \rightarrow (ss\ \kappa)$. Unfolding ss gives: $\kappa \rightarrow \forall s(([\kappa](mg\ s)) \rightarrow (mg\ s))$

which is equivalent to: $\forall s(([\kappa](mg\ s)) \rightarrow (\kappa \rightarrow (mg\ s)))$ which is an instance of A1. QED.

SS2: $(\kappa \equiv (ss\ \kappa)) \rightarrow \exists s(\kappa \equiv (ms\ s))$

proof: Letting s be $\{s: ([\kappa](mg\ s))$ gives $(\kappa \equiv (ss\ \kappa)) \rightarrow (\kappa \equiv (ms\{s: ([\kappa](mg\ s))))$. Unfolding ms and lambda conversion gives: $(\kappa \equiv (ss\ \kappa)) \leftrightarrow (\kappa \equiv \forall s(([\kappa](mg\ s)) \rightarrow (mg\ s)))$. Folding ss gives a tautology. QED.

The theorems MD3 and MD4 are analogous to MD1 and MD2 except that DL is replaced by the combined meanings of the sentences entailed by DL.

MD3: [ss(DL κ $\forall i\Gamma_i$ α_i:β_{ij}/χ_i)]$\forall i\Gamma_i$

proof: By R0 it suffices to prove (ss(DL κ $\forall i\Gamma_i$ α_i:β_{ij}/χ_i))$\rightarrow\forall i\Gamma_i$ which is equivalent to:

(ss(DL κ $\forall i\Gamma_i$ α_i:β_{ij}/χ_i))$\rightarrow\Gamma_i$. Unfolding ss gives: $\forall$s(([(DL κ $\forall i\Gamma_i$ α_i:β_{ij}/χ_i)](mg s))$\rightarrow$(mg s)))$\rightarrow\Gamma_i$ which by the meaning laws M0-M8 is equivalent to:($\forall$s(([(DL κ $\forall i\Gamma_i$ α_i:β_{ij}/χ_i)](mg s))$\rightarrow$(mg s)))$\rightarrow$(mg 'Γ_i. Backchaining on (mg 'Γ_i) with s in the hypothesis assigned to be 'Γ_i in the conclusion shows that it suffices to prove:

([(DL κ $\forall i\Gamma_i$ α_i:β_{ij}/χ_i)](mg 'Γ_i)) which by the meaning laws: M0-M8 is equivalent to: ([(DL κ $\forall i\Gamma_i$ α_i:β_{ij}/χ_i)]Γ_i)

which by the laws of S5 is equivalent to: ([(DL κ $\forall i\Gamma_i$ α_i:β_{ij}/χ_i)]$\forall i\Gamma_i$) which is an instance of MD1. QED.

MD4: (([ss(DL κ Γ α_i:β_{ij}/χ_i)]α_i)$\wedge$($\wedge$j=1,mi<κ>β_{ij}))$\rightarrow$ ([ss(DL κ Γ α_i:β_{ij}/χ_i)]χ_i)

proof: Unfolding the last ss gives:

(([ss(DL κ Γ α_i:β_{ij}/χ_i)]α_i)$\wedge\wedge$j=1,mi(<κ>β_{ij}))$\rightarrow$([$\forall$s(([(DL κ Γ α_i:β_{ij}/χ_i)](mg s))$\rightarrow$(mg s))]χ_i)

Instantiating s in the hypothesis to 'χ_i and then dropping the hypothesis gives:

(([ss(DL κ Γ α_i:β_{ij}/χ_i)]α_i)$\wedge\wedge$j=1,mi(<κ>β_{ij}))$\rightarrow$([(([(DL κ Γ α_i:β_{ij}/χ_i)](mg 'χ_i))$\rightarrow$(mg 'χ_i))]χ_i). Using the meaning laws M0-M7 gives: (([ss(DL κ Γ α_i:β_{ij}/χ_i)]α_i)$\wedge\wedge$j=1,mi(<κ>β_{ij}))$\rightarrow$([(([(DL κ Γ α_i:β_{ij}/χ_i)]χ_i)$\rightarrow\chi_i$)]χ_i).Backchaining on χ_i, it suffices to prove: (([ss(DL κ Γ α_i:β_{ij}/(mg 'χ_i))]α_i)$\wedge\wedge$j=1,mi(<κ>β_{ij}))$\rightarrow$([(DL κ Γ α_i:β_{ij}/(mg 'χ_i))]χ_i)

By SS1 and the first hypothesis it suffices to prove:

(([(DLκ Γ α_i:β_{ij}/(mg 'χ_i))]α_i)$\wedge\wedge$j=1,mi(<κ>β_{ij}))$\rightarrow$([(DL κ Γ α_i:β_{ij}/(mg 'χ_i))]χ_i) which is an instance of MD2. QED.

Finally MD5, MD6, and MD7 show that talking about the meanings of sets of FOL sentences in the modal representation of Default Logic is equivalent to talking about propositions in general.

MD5: ($\exists$p((ms p)$\wedge$([(ms p)]($\forall i\Gamma_i$))$\wedge\forall_i$((([(ms p)]α_i)$\wedge\wedge$j=1,mi<κ>β_{ij})$\rightarrow$[(ms p)]χ_i))))$\equiv$(DL κ($\forall i\Gamma_i$)α_i:β_{ij}/χ_i)

proof: The proof divides into two entailments:

(1) [$\exists$p((ms p)$\wedge$([(ms p)]$\forall i\Gamma_i$)$\wedge\forall_i$((([(ms p)]α_i)$\wedge\wedge$j=1,mi(<κ>β_{ij}))$\rightarrow$[(ms p)]χ_i))] (DL κ($\forall i\Gamma_i$)α_i:β_{ij}/χ_i)

DL is unfolded giving: [((ms p)$\wedge$([(ms p)]($\forall i\Gamma_i$))$\wedge\forall_i$((([(ms p)]α_i)$\wedge\wedge$j=1,mi<κ>β_{ij})$\rightarrow$[(ms p)](mg 'χ_i)))]

$\exists$p(p$\wedge$([p]$\forall i\Gamma_i$)$\wedge\forall_i$((([p]α_i)$\wedge\wedge$j=1,mi<κ>β_{ij})$\rightarrow$[p](mg 'χ_i))))

Instantiating the quantified p in the conclusion to be (ms p) produces a tautology.

(2) [(DL κ($\forall i\Gamma_i$)α_i:β_{ij}/χ_i)] $\exists$p((ms p)$\wedge$([(ms p)]$\forall i\Gamma_i$)$\wedge\forall_i$((([(ms p)]α_i))$\wedge\wedge$j=1,mi<κ>β_{ij})$\rightarrow$[(ms p)](mg 'χ_i)))

p is assigned to be the set: {s: [(DL κ($\forall i\Gamma_i$)α_i:β_{ij}/χ_i)](mg s))}.

Since p only occurs in (ms p) and since (ms{s: ([(DL κ)](mg s))}) is equivalent to (ss(DL κ)) we get:

[(DL κ)]((ss(DL κ))$\wedge$([(ss(DL κ))]$\forall i\Gamma_i$)$\wedge\forall_i$((([(ss(DL κ))]α_i)$\wedge\wedge$j=1,mi<κ>β_{ij})$\rightarrow$([(ss(DL κ))](mg 'χ_i))))

which holds by theorems SS1, MD3, and MD4. QED.

MD6: (ss(DL κ($\forall i\Gamma_i$)α_i:β_{ij}/χ_i))$\equiv$(DL κ($\forall i\Gamma_i$)α_i:β_{ij}/χ_i)

proof: In view of SS1, it suffices to prove:([(ss(DL κ))](DL κ). Unfolding the second occurrence of DL gives:

[(ss(DL κ))]$\exists$p(p$\wedge$([p]($\forall i\Gamma_i$))$\wedge\forall_i$((([p]α_i)$\wedge\wedge$j=1,mi<κ>β_{ij})$\rightarrow$[p](mg 'χ_i))). Letting p be (ss(DL κ)) then gives:

[(ss(DL κ))]((ss(DL κ))$\wedge$([(ss(DL κ))](($\forall i\Gamma_i$))$\wedge\forall_i$((([(ss(DL κ))]α_i)$\wedge\wedge$j=1,mi<κ>β_{ij})$\rightarrow$([(ss(DL κ))](mg 'χ_i)))

which holds by theorems MD3 and MD4. QED.

MD7: ($\kappa\equiv$(DL κ($\forall i\Gamma_i$)α_i:β_{ij}/χ_i))$\rightarrow\exists$s($\kappa\equiv$(ms s))

proof: From the hypothesis and MD6 $\kappa\equiv(ss(DL\ \kappa))$ is derived. Using the hypothesis to replace $(DL\ \kappa)$ by κ in this result gives: $\kappa\equiv(ss(DL\ \kappa))$, By SS2 this implies the conclusion. QED.

Conclusion: The Relationship between Default Logic and the Modal Logic

The relationship between the proof theoretic definition of Default Logic [Reiter 1980] and the modal representation is proven in two steps. First theorem DL1 shows that the meaning of the set dl is the proposition DL and then theorem DL2 shows that a set of FOL sentences which contains its FOL theorems is a fixedpoint of the fixedpoint equation of Default Logic with an initial set of axioms and defaults if and only if the meaning (or rather disquotation) of that set of sentences is logically equivalent to DL of the meanings of that initial set of sentences and those defaults.

DL1: $(ms(dl(fol\ '\kappa)\{'\Gamma_i\}\ '\alpha_i{:}'\beta_{ij}/'\chi_i))\equiv(DL(ms\ '\kappa)(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)$

proof: $(ms(dl(fol\ '\kappa)\{'\Gamma_i\}\ '\alpha_i{:}'\beta_{ij}/'\chi_i))$ Unfolding the definition of dl gives:

$ms(\cap\{p:\ (p\supseteq(fol\ p))\wedge(p\supseteq\{'\Gamma_i\})\wedge\forall_i((('\alpha_i\varepsilon p)\wedge\wedge_{j=1,mi}('(\neg\beta_{ij})\notin(fol\ '\kappa)))\rightarrow('\chi_i\varepsilon p))\})$. By FOL6 this is:

$ms\{s:\forall p((((fol\ p)\supseteq\{'\Gamma_i\})\wedge\forall_i((('\alpha_i\varepsilon(fol\ p))\wedge\wedge_{j=1,mi}('(\neg\beta_{ij})\notin(fol\ '\kappa)))\rightarrow('\chi_i\varepsilon(fol\ p))))\rightarrow(s\varepsilon(fol\ p)))\}$

Using C1 four times, C3, and FOL4 this is equivalent to: $ms\{s:\forall p((([(ms\ p)](ms\{'\Gamma_i\}))\wedge\forall_i((([(ms\ p)](mg\ '\alpha_i))\wedge\wedge_{j=1,mi}\neg([(ms\ '\kappa)](mg\ '\neg\beta_{ij})))\rightarrow([(ms\ p)](mg\ '\chi_i))))\rightarrow([(ms\ p)](mg\ s)))\}$

By the meaning laws M0-M7 this is equivalent to:

$ms\{s:\forall p((([(ms\ p)](ms\{'\Gamma_i\}))\wedge\forall_i((([(ms\ p)]\alpha_i)\wedge\wedge_{j=1,mi}\neg([(ms\ '\kappa)]\neg\beta_{ij}))\rightarrow([(ms\ p)]\chi_i)))\rightarrow([(ms\ p)](mg\ s)))\}$

By MS2 this is equivalent to:

$ms\{s:\forall p((([(ms\ p)](\forall i\Gamma_i))\wedge\forall_i((([(ms\ p)]\alpha_i)\wedge\wedge_{j=1,mi}\neg([(ms\ '\kappa)]\neg\beta_{ij}))\rightarrow([(ms\ p)]\chi_i)))\rightarrow([(ms\ p)](mg\ s)))\}$

Folding <> gives: $ms\{s:\forall p((([(ms\ p)](\forall i\Gamma_i))\wedge\forall_i((([(ms\ p)]\alpha_i)\wedge\wedge_{j=1,mi}(<(ms\ '\kappa)>\beta_{ij}))\rightarrow([(ms\ p)]\chi_i)))\rightarrow([(ms\ p)](mg\ s)))\}$

By S5 Modal Quantificational Logic this is equivalent to:

$ms\{s:\ ([\exists p((ms\ p)\wedge([(ms\ p)](\forall i\Gamma_i))\wedge\forall_i((([(ms\ p)]\alpha_i)\wedge\wedge_{j=1,mi}(<(ms\ '\kappa)>\beta_{ij}))\rightarrow([(ms\ p)]\chi_i)))]\ (mg\ s))\}$

By MD5 this is equivalent to: $ms\{s:\ ([(DL(ms\ '\kappa)(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)](mg\ s))\}$

Unfolding ms and lambda conversion gives: $\forall s(([(DL(ms\ '\kappa)(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)](mg\ s))\rightarrow(mg\ s))$

Folding ss gives: $ss(DL(ms\ '\kappa)(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)$. By MD6 is equivalent to:$(DL(ms\ '\kappa)(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)$ QED.

DL2: $((fol\ '\kappa)=(dl(fol\ '\kappa)\{'\Gamma_i\}\ '\alpha_i{:}'\beta_{ij}/'\chi_i))\leftrightarrow((ms\ '\kappa)\equiv(DL(ms\ '\kappa)(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)))$

proof: By FOL8 $(fol\ '\kappa)=(dl(fol\ '\kappa)\{'\Gamma_i\}\ '\alpha_i{:}'\beta_{ij}/'\chi_i))$ is equivalent to: $(fol\ '\kappa)=(fol(dl(fol\ '\kappa)\{'\Gamma_i\}\ '\alpha_i{:}'\beta_{ij}/'\chi_i))$.

By C4 this is equivalent to: $(ms\ '\kappa)\equiv(ms(dl(fol\ '\kappa)\{'\Gamma_i\}\ '\alpha_i{:}'\beta_{ij}/'\chi_i))$.

By DL1 this is equivalent to: $(ms\ '\kappa)\equiv(DL(ms\ '\kappa)(\forall i\Gamma_i)\alpha_i{:}\beta_{ij}/\chi_i)$ QED.

Theorem DL2 shows that the set of theorems: $(fol\ '\kappa)$ of a set $'\kappa$ is a fixedpoint of a fixed point equation of Default Logic if and only if the meaning $(ms\ '\kappa)$ of $'\kappa$ is a solution to the necessary equivalence. Furthermore, by FOL9 there are no other fixedpoints (such as a set not containing all its theorems) and by MD7 there are no other solutions (such as a proposition not representable as a sentence in the FOL object language). Therefore, the Modal representation of Default Logic (i.e. DL), faithfully represents the set theoretic description of Default Logic (i.e. dl).Finally, we note that $(\forall i\Gamma_i)$ and $(ms\ '\kappa)$ may be generalized to be arbitrary propositions Γ and κ giving the more general modal representation: $\kappa\equiv(DL\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)$.

Acknowledgements

This research was supported by National Science Foundation grants: #9818341 and #9972843.

Bibliography

[Anatoniou 1997] Antoniou, Grigoris 1997. *NonMonotonic Reasoning*, MIT Press.

[Bressan 1972] Bressan, Aldo 1972. *A General Interpreted Modal Calculus,* Yale University Press.

[Brown 1978] Brown,F.M., "A Semantic Theory for Logic Programming", *Colloquia Mathematica Societatis Janos Bolyai 26*, *Mathematical Logic in Computer Science*, Salgotarjan, Hungry, 1978.

[Boyer&Moore 1981] R. S. Boyer and J. Strother Moore, "Metafunctions: proving them correct and using them efficiently as new proof procedures," *The Correctness Problem in Computer Science,* R. S. Boyer and J. Strother Moore, eds., Academic Press, New York, 1981.

[Brown 1986] Brown, Frank M. 1986. "Reasoning in a Hierarchy of Deontic Defaults", *Proceedings of the Canadian Artificial Intelligence Conference CSCI 86*, Montreal, Canada, Morgan-Kaufmann, Los Altos.

[Brown 1987] Brown, Frank M. 1987. "The Modal Logic Z", In *The Frame Problem in AI"; Proc. of the 1987 AAAI Workshop*, Morgan Kaufmann, Los Altos, CA .

[Brown 1989] Brown, Frank M. 1989. "The Modal Quantificational Logic Z Applied to the Frame Problem", advanced paper *First International Workshop on Human & Machine Cognition*, May 1989 Pensacola, Florida. Abbreviated version published in *International Journal of Expert Systems Research and Applications, Special Issue: The Frame Problem. Part A.* eds. Keneth Ford and Pattrick Hayes, vol. 3 number 3, pp169-206 JAI Press 1990. Reprinted in *Reasoning Agents in a Dynamic World: The Frame problem*, editors: Kenneth M. Ford, Patrick J. Hayes, JAI Press 1991.

[Carnap 1946] Carnap, Rudolf 1946. "Modalities and Quantification" *Journal of Symbolic Logic*, vol. 11, number 2.

[Carnap 1956] Carnap, Rudolf 1956. *Meaning and Necessity: A Study in the Semantics of Modal Logic*, The University of Chicago Press.

[Fine 1970] Fine, K. 1970. "Propositional Quantifiers in Modal Logic" *Theoria* 36, p336--346.

[Hendry & Pokriefka 1985] Hendry, Herbert E. and Pokriefka, M. L. 1985. "Carnapian Extensions of S5", *Journal of Phil. Logic 14*.

[Hughes & Cresswell 1968] Hughes,G. E. & Creswell,M. J., 1968. *An Introduction to Modal Logic*, Methuen & Co. Ltd., London.

[Leasure 1993] Leasure, David E., 1993. The Modal Logic Z Applied to Lifschitz's Benchmark problems for Formal Nonmonotonic Reasoning, University of Kansas Dissertation, University of Kansas Library.

[Lewis 1936] Lewis, C. I. 1936. Strict Implication, *Journal of Symbolic Logic*, vol I.

[Mendelson 1964] Mendelson, E. 1964. *Introduction to Mathematical Logic*, Van Norstrand, Reinhold Co., New York.

[Parks 1976] Parks, Z. 1976. "An Investigation into Quantified Modal Logic", Studia Logica 35, p109-125.

[Quine 1969] Quine, W.V.O., Set Theory and Its Logic, revised edition, Oxford University Press, London, 1969.

[Reiter 1980] Reiter, R. 1980. "A Logic for Default Reasoning" *Artificial Intelligence*, 13.

[Schwind 1990] Schwind, Camilla 1990. "A Tableaux-Based Theorem Prover for a Decidable Subset of Default Logic", 10th International Conference on Automated Deduction, Kaiserslautern. Springer Verlag. Lecture Notes in AI vol 449.

Author information

Frank M. Brown- Artificial Intelligence Laboratory, University of Kansas, Lawrence, Kansas, 66045, e-mail: brown@ku.edu.

ON THE RELATIONSHIP BETWEEN QUANTIFIED REFLECTIVE LOGIC AND QUANTIFIED DEFAULT LOGIC

Frank M. Brown

***Abstract**: Reflective Logic and Default Logic are both generalized so as to allow universally quantified variables to cross modal scopes whereby the Barcan formula and its converse hold. This is done by representing both the fixed point equation for Reflective Logic and the fixed point equation for Default both as necessary equivalences in the Modal Quantificational Logic Z. and then inserting universal quantifiers before the defaults. The two resulting systems, called Quantified Reflective Logic and Quantified Default Logic, are then compared by deriving metatheorems of Z that express their relationships. The main result is to show that every solution to the equivalence for Quantified Default Logic is a strongly grounded solution to the equivalence for Quantified Reflective Logic. It is further shown that Quantified Reflective Logic and Quantified Default Logic have exactly the same solutions when no default has an entailment condition.*

***Keywords**: Quantified Reflective Logic, Quantified Default Logic, Modal Logic, Nonmonotonic Logic.*

1. Introduction

Two nonmonotonic logics which inherently deal with entailment conditions in addition to possibility conditions in their defaults; are Reflective Logic and Default Logic [Reiter 1980] [Antoniou 1997]. The fixed point solutions to Default Logic are defined by the set theoretic equation $\kappa=(\text{dl } \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)$ where:

$(\text{dl } \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i) =_{df} \cap\{p{:}(p\supseteq(\text{fol } p))\wedge(p\supseteq\Gamma)\wedge \wedge_i(((\alpha_i\varepsilon p)\wedge \wedge_{j=1,m_i}((\neg\beta_{ij})\notin\kappa))\rightarrow(\chi_i\varepsilon p))\}$

where α_i, β_{ij}, and χ_i are closed sentences of First Order Logic and Γ is a set of closed sentences of First Order Logic. $\wedge_{j=1,m_i}$ stands for the conjunction of the formula which follows it as j ranges from 1 to m_i. If $m_i=0$ then it specifies #t. $\wedge_i$ is also a conjunction. By closed it is meant that no sentence may contain a free variable. (fol p) is the set of theorems deducible in First Order Logic from the set p. The fixed point solutions for Reflective Logic, can be defined by the simpler set theoretic equation $\kappa=(\text{rl } \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)$ given in [Brown 1989] where:

$(\text{rl } \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i) =_{df} (\text{fol}(\Gamma\cup\{\chi_i{:}(\alpha_i\varepsilon\kappa)\wedge \wedge_{j=1,m_i}((\neg\beta_{ij})\notin\kappa)\}))$

where α_i, β_{ij}, and χ_i are again closed sentences of First Order Logic and Γ is a set of closed sentences of First Order Logic. When the context is obvious $\Gamma\ \alpha_i{:}\beta_{ij}/\chi_i$ is omitted and instead just $(\text{rl } \kappa)$ is written.

These two nonmonotonic systems have the basic problem that they do not explicate the case where free variables occur in the α_i, β_{ij}, and χ_i sentences and which are universally quantified just over the scope of those sentences. To carry out such an explication we want to transform $(\text{dl } \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)$ into something like:

$$\cap\{p{:}(p\supseteq(\text{fol } p))\wedge(p\supseteq\Gamma)\wedge \wedge_i\underline{\forall\xi_i}(((\alpha_i\varepsilon p)\wedge \wedge_{j=1,m_i}((\neg\beta_{ij})\notin\kappa))\rightarrow(\chi_i\varepsilon p))\}$$

and $(\text{rl } \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)$ into something like:[4] $(\text{fol}(\Gamma\cup\{\Psi{:}\ \vee_i\underline{\exists\xi_i}(\Psi=\chi_i\wedge(\alpha_i\varepsilon\kappa)\wedge \wedge_{j=1,m_i}((\neg\beta_{ij})\notin\kappa))\}))$

where $\underline{\xi}_i$ is a sequence of variables and the universal quantifier really means universal quantification. That is, the Barcan formula and its converse hold [Carnap 46] so that a property universally holds (in κ) if and only if it holds (in κ) for everything: $((\underline{\forall\xi}\alpha)\varepsilon\kappa)\leftrightarrow(\underline{\forall\xi}(\alpha\varepsilon\kappa))$. The problem lies in the fact that α_i, β_{ij}, and χ_i are necessarily closed sentences of First Order Logic.[5]

---

[4] When the set theoretic notation is unraveled the existential quantifiers specified herein are essentially universally quantified over the defaults as can be seen in the equivalent equation: $k=\cap\{p{:}(p\supseteq(\text{folth } p))\wedge(p\supseteq\Gamma)\wedge \wedge_i\underline{\forall\xi}(((\alpha_i\varepsilon k)\wedge\wedge_{j=1,m_i}((\neg\beta_{ij})\notin k))\rightarrow(\chi_i\varepsilon p))\}$

[5] Of course one generally gives a meaning to such a sentence by saying that all the free variables are implicitly universally quantified or that all such variables are implicitly existentially quantified. However, neither approach allows a quantifier to refer to the same free variable in α_i, β_{ij}, and χ_i. This issue is discussed in more detail in section 3.2 in [Antoniou 1987].

However, [Brown 2003a] showed how Reflective Logic can be represented in Modal Logic by the necessary equivalence: $\kappa \equiv (RL\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)$ where:

$(RL\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i) =_{df} \Gamma \wedge \wedge_i((([\kappa]\alpha_i) \wedge (\wedge_{j=1,mi}(<\kappa>\beta_{ij}))) \rightarrow \chi_i)$

Likewise [Brown 2003b] showed how Default Logic can be represented in Modal Logic by the necessary equivalence: $\kappa \equiv (DL\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i)$ where:

$(DL\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i) =_{df} \exists p(p \wedge ([p]\Gamma) \wedge \wedge_i((([p]\alpha_i) \wedge (\wedge_{j=1,mi}(<\kappa>\beta_{ij}))) \rightarrow [p]\chi_i))$

The advantage of the modal representations is that quantifiers can be embedded in them wherever we wish thus allowing inserted universal quantifiers to capture the free variables in α_i, β_{ij}, and χ_i , giving the generalizations:

$(QRL\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i) =_{df} \Gamma \wedge \wedge_i \underline{\forall \xi_i}((([\kappa]\alpha_i) \wedge (\wedge_{j=1,mi}(<\kappa>\beta_{ij}))) \rightarrow \chi_i)$

$(QDL\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i) =_{df} \exists p(p \wedge ([p]\Gamma) \wedge \wedge_i \underline{\forall \xi_i}((([p]\alpha_i) \wedge (\wedge_{j=1,mi}(<\kappa>\beta_{ij}))) \rightarrow [p]\chi_i))$

Having created two new nonmonotonic systems (i.e. QRL and QDL) the question arrises as to how their fixed-point solutions are related. Herein we address this question. Section 2 axiomatizes the Z Modal Quantificational Logic. Quantified Reflective Logic (i.e., QRL) is defined in section 3 and some basic theorem schemata about it are proven. Quantified Default Logic (i.e., QDL) is defined in section 4 and some basic theorem schemata about it are proven. The main result is proven in section 5. Finally, some conclusions are drawn in section 6.

2. Axiomatization of Z Modal Logic

The Modal Quantificational Logic Z [Brown 1987] is a seven tuple: ($\rightarrow$, #f, $\forall$, [], *vars*, *predicates*, *functions*) where $\rightarrow$, #f, $\forall$, and [] are logical symbols, *vars* is a set of variable symbols, *predicates* is a set of predicate symbols each of which has an implicit arity specifying the number of terms associated with that predicate, and *functions* is a set of function symbols each of which has an implicit arity specifying the number of terms associated with that function. The sets of logical symbols, variables, predicate symbols, and function symbols are pairwise disjoint. The set of terms is the smallest set which includes the variables and is closed under the process of forming new terms from other terms using the function symbols of the language. The set of sentences is the smallest set which includes #f, the *variables*, and each of the *predicates* followed by an appropriate number of terms, and is closed under the process of forming new sentences from other sentences using the logical symbols of the language, provided that no variable in any subexpression has free occurrences both as a sentence and as a term. Variables that occur only in term positions are called concept variables. Variables which occur only in sentence positions are called propositional variables. Lower case Roman letters possibly indexed with digits are used as variables of Z. Greek letters are used as syntactic metavariables. γ, $\gamma 1,...\gamma n$, range over the variables, ξ, $\xi 1,...\xi n$ range over a sequence of variables of an appropriate arity, $\pi,\pi 1...\pi n,\rho,\rho 1...\rho n$ range over the predicate symbols, ϕ, $\phi 1...\phi n$ range over function symbols, δ, $\delta 1...\delta n$ range over terms, Δ, $\Delta 1...\Delta n$ range over a sequence of terms of an appropriate arity, and α, $\alpha 1...\alpha n$, β, $\beta 1...\beta n,\chi$, $\chi 1...\chi n$, Γ, and Ψ range over sentences. Thus, the terms are of the forms γ and $(\phi\ \delta 1...\delta n)$, and the sentences are of the forms $(\alpha \rightarrow \beta)$, #f, $(\forall \gamma\ \alpha)$, $([]\alpha)$, $(\pi\ \delta 1...\delta n)$, and γ. A nullary predicate π or function ϕ is written as a sentence or term without parentheses. The primitive symbols of Z are shown in Figure 1.

Symbol	Meaning		Symbol	Meaning
$\alpha \rightarrow \beta$	if α then β.		$\forall \gamma\ \alpha$	for all γ, α.
#f	falsity		$[]\ \alpha$	α is logically necessary

Figure 1: Primitive Symbols of Z

The defined symbols of Z are listed in Figure 2 below with their intuitive interpretations.

Symbol	Definition	Meaning		Symbol	Definition	Meaning
$\neg\alpha$	$\alpha \rightarrow$ #f	not α		$\alpha\wedge\beta$	$\neg(\alpha \rightarrow \neg\beta)$	α and β
#t	$\neg$ #f	truth		$\alpha\leftrightarrow\beta$	$(\alpha\rightarrow\beta)\wedge(\beta\rightarrow\alpha)$	α if and only if β
$\alpha\vee\beta$	$(\neg\alpha)\rightarrow\beta$	α or β		$\exists\gamma\ \alpha$	$\neg\forall\gamma\neg\alpha$	for some γ , α
$<>\alpha$	$\neg[]\neg\alpha$	α is logically possible		$[\beta]\ \alpha$	$([](\beta\rightarrow\alpha))$	β entails α
$\alpha\equiv\beta$	$[](\alpha\leftrightarrow\beta)$	α is logically equivalent toβ		$<\beta>\ \alpha$	$(<>(\beta\wedge\alpha))$	α is possible with β

Figure 2: Defined Symbols of First Order Logic

Z is effectively axiomatized with a recursively enumerable set of theorems as the set of axioms is itself recursively enumerable and its inference rules are recursive. The classical (i.e., non-modal) axioms and inference rules of Z include those of Quantificational Logic [Mendelson 1964] given in Figure 3. The laws MR1, MR2, MA1-MA7 are a standard set of axioms and inference rules for First Order Quantificational Logic except for the following: point: Because γ in MR2, MA4, and MA5 may be a propositional variable these laws constitute a fragment of Second Order Logic. Propositional quantifiers in modal logics have been investigated in [Fine 1970].

MA1: $\alpha \rightarrow (\beta\rightarrow \alpha)$ MR1: from α and $(\alpha\rightarrow \beta)$ infer β

MA2: $(\alpha\rightarrow (\beta\rightarrow \rho)) \rightarrow ((\alpha\rightarrow \beta)\rightarrow (\alpha\rightarrow \rho))$ MR2: from α infer $(\forall\gamma\ \alpha)$

MA3: $((\neg\alpha)\rightarrow(\neg\beta))\rightarrow((\neg\alpha)\rightarrow\beta)\rightarrow\alpha)$

MA4: $(\forall\gamma\ \alpha)\rightarrow\beta$ where β is the result of substituting an expression (which is free for the free positions of γ in α) for all the free occurrences of γ in α.

MA5: $((\forall\gamma\ (\alpha \rightarrow \beta)) \rightarrow\ (\alpha \rightarrow(\forall\gamma\ \beta)))$ where γ does not occur in α.

Figure 3: The Classical Rules and Axioms of Z

The modal inference rule and axioms of Z about logical necessity (i.e., []) are given in Figure 4. R0, A1, A2, and A3 constitute an S5 Modal Logic [Hughes and Cresswell 1968] which, with the nonmodal laws, is an S5 modal quantificational logic similar to [Carnap 1946], [Carnap 1956], and a First Order Logic version [Parks 1976] of [Bressan 1972] in which the Barcan formula: $(\forall\gamma([]\alpha))\rightarrow([]\forall\gamma\alpha)$ and its converse hold. R0 implies that all assertions are logically necessary. Thus, in any logic with R0, contingent facts Γ holding in a knowledgebase κ are specified by asserting $([\kappa]\Gamma)$. If Γ is all that is in κ then $\kappa\equiv\Gamma$ is asserted. The variable κ may occur in Γ.

R0: from α infer $([]\ \alpha)$ A4: $([]\alpha) \rightarrow ([](\alpha\{\pi\ /\ \lambda\xi\beta\}$

A1: $([]p) \rightarrow p$ A5: $([]\alpha) \rightarrow ([](\alpha\{\phi\ /\ \lambda\xi\delta\}$

A2: $([p]q) \rightarrow (([]p)\rightarrow ([]q))$ A6: $\neg\ [](\forall x\forall y(x=y))$

A3: $([]p) \vee ([]\neg[]p))$

Figure 4: The Modal Inference Rule and Axioms of Z

A4 is the key axiom schema of Z. It is far stronger than the trivial possibility axioms such as $\exists pq((\neg[p]q)\wedge(\neg[p]\neg q))$ assumed in [Lewis 1936] and $\exists p((<>p)\wedge(<>\neg p))$ assumed in [Bressan 1972]. It also extends certain axiom schemata used in propositional logic, including the PropPosAx schema in [Brown 1979], S13 [Cocchiarella 1984], and S5c [Hendry and Pokriefka 1985].

3. Quantified Reflective Logic

The formula for Quantified Reflective Logic[6] (i.e., QRL) [Brown 1989][7] is defined in Z as follows:

[6] In the QRL generalization of Autoepistemic Logic the Barcan formula and its converse hold for [k]: $([k]\forall\xi\alpha)\leftrightarrow(\forall\xi[k]\alpha)$ since they are inherited from the S5 modal properties of []. Thus QRL differs from the generalization of Autoepistemic Logic given in [Konolige 1989]

RL0: $(QRL\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i) =df\ \Gamma\wedge\wedge_i\forall\xi_i((([\kappa]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow\chi_i)$

where Γ, α_i, β_{ij}, and χ_i are sentences of Z and κ does not occur in ξ. These sentences may contain free variables some of which may be captured by the $\forall\xi$ quantifier. When the context is obvious $\Gamma\ \alpha_i{:}\beta_{ij}/\chi_i$ is omitted and instead just $(QRL\ \kappa)$ is written. Interpreted as a doxastic logic, the equivalence:

$$\kappa\equiv(QRL\ \kappa)$$

states:

that which is believed is logically equivalent to Γ and for each i, for all ξ if α_i is believed and for each j, β_{ij} is believable then χ_i

Here are some simple properties of QRL, namely that $(QRL\ \kappa)$ entails Γ and any conclusion χ_i of a default whose conditions hold:

R1: $[(QRL\ \kappa)]\Gamma$

proof: Unfolding QRL gives: $[\Gamma\wedge\wedge_i\forall\xi_i((([\kappa]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow\chi_i)]\Gamma$ which is a tautology. QED.

R2: $((([\kappa]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow([(QRL\ \kappa)]\chi_i)$

proof: Unfolding QRL gives: $((([\kappa]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow([\Gamma\wedge\wedge_i\forall\xi_i((([\kappa]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow\chi_i)]\chi_i)$

Using the hypotheses on the ith instance and where the quantified ξ is instantiated to ξ gives:

$((([\kappa]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow([\Gamma\wedge\wedge_i\forall\xi_i((([\kappa]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow\chi_i)\wedge\chi_i]\chi_i)$ which is a tautology. QED.

4. Quantified Default Logic

The formula for Quantified Default Logic (i.e., QDL) [Brown 1989] is defined in Z as follows:

D0: $(QDL\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i) =df\ \exists p(p\wedge([p]\Gamma)\wedge\wedge_i\forall\xi_i((([p]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow[p]\chi_i))$

where Γ, α_i, β_{ij}, and χ_i are sentences of Z without any free occurences of p and neither p nor κ occur in ξ. These sentences may contain free variables some of which may be captured by the $\forall\xi_i$ quantifier. When the context is obvious $\Gamma\ \alpha_i{:}\beta_{ij}/\chi_i$ is omitted and just $(QDL\ \kappa)$ is written. Interpreted as a doxastic logic the equivalence:

$$\kappa\equiv(QDL\ \kappa)$$

states:

that which is believed is logically equivalent to the disjunction of all potential belief states such that: Γ is potentially believed and for each i, for all ξ if α_i is potentially believed and for each j, β_{ij} is believable then χ_i is potentially believed.

Given below are some simple properties of QDL. The first two state that QDL entails Γ and any conclusion χ_i of a default whose entailment condition holds in QDL and whose possible conditions are possible with κ.

D1: $[(QDL\ \kappa)]\Gamma$

proof: Unfolding QDL gives: $[\exists p(p\wedge([p]\Gamma)\wedge\wedge_i\forall\xi_i((([p]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow([p]\chi_i))]\Gamma$

---

where the converse of the Barcan formula: $((L\forall\xi\alpha)\rightarrow(\forall\xi L\alpha))$ is not valid. (In terms of Autoepistemic kernels: this amounts to saying that $((\underline{\forall\xi}\alpha)\varepsilon k)\rightarrow(\underline{\forall\xi}(\alpha\varepsilon k))$ does not hold.

[7] The formula for producing the entire Autoepistemic fixedpoint rather than just the kernel is:
$\Gamma\wedge\wedge_i\forall\xi((([k]\alpha_i)\wedge(\wedge_{j=1,m_i}(<k>\beta_{ij})))\rightarrow\chi_i)\wedge\wedge\Psi\forall\xi((L`\Psi)\leftrightarrow([k]\Psi))$.

Since p is not free in Γ, pulling $\exists p$ out of the hypothesis of the entailment gives:
$\forall p((([p]\Gamma)\wedge\wedge_i\forall\xi_i((([p]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow([p]\chi_i)))\rightarrow([p]\Gamma))$ which is a tautology. QED.

D2: $(([(QDL\ \kappa)]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow([(QDL\ \kappa)]\chi_i)$
proof: Unfolding both occurrences of QDL gives:
$(([\exists p(p\wedge([p]\Gamma)\wedge\wedge_i\forall\xi_i((([p]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow([p]\chi_i)))]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))$
$\rightarrow([\exists p(p\wedge([p]\Gamma)\wedge\wedge_i\forall\xi_i((([p]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow([p]\chi_i)))]\chi_i)$
Since p is not free in α_i and χ_i, pulling $\exists p$ out of the hypotheses of the entailments gives:
$((\forall p((([p]\Gamma)\wedge\wedge_i\forall\xi_i((([p]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow([p]\chi_i)))\rightarrow([p]\alpha_i))\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij}))))$
$\rightarrow\forall p((([p]\Gamma)\ \wedge\wedge_i\forall\xi_i((([p]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow([p]\chi_i)))\rightarrow([p]\chi_i))$
Instantiating the p in the hypothesis to the p in the conclusion gives:
$(((([p]\Gamma)\wedge\wedge_i\forall\xi_i((([p]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow([p]\chi_i)))\rightarrow([p]\alpha_i))$
$\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij}))\wedge([p]\Gamma)\wedge\wedge_i\forall\xi_i((([p]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow([p]\chi_i)))\rightarrow([p]\chi_i)$
which simplifies to just:$((([p]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij}))\wedge([p]\Gamma)\wedge\wedge_i\forall\xi_i((([p]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow([p]\chi_i)))\rightarrow([p]\chi_i)$
Since p is not in ξ, forward chaining using the first and second hypotheses on the fourth proves the theorem. QED.

A slightly stronger version of QDL is defined below:
D3: $(QDL^*\ \kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i) =df \exists p(p\wedge([\kappa]p)\wedge([p]\Gamma)\wedge\wedge_i\forall\xi_i((([p]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow([p]\chi_i)))$

D4: $[(QDL^*\ \kappa)](QDL\ \kappa)$
proof: Unfolding QDL* and QDL gives: $[\exists p(p\wedge([\kappa]p)\wedge([p]\Gamma)\ \wedge\wedge_i\forall\xi_i((([p]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow([p]\chi_i)))]$
$\exists p(p\wedge([p]\Gamma)\wedge\wedge_i\forall\xi_i((([p]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow[p]\chi_i))$
Letting p in the conclusion be the p in the hypothesis results in a tautology. QED.

Theorem D5 shows that QDL and QDL* are logically equivalent whenever κ entails the QDL formula:
D5: $([\kappa](QDL\ \kappa))\rightarrow((QDL\ \kappa)\equiv(QDL^*\ \kappa))$
proof: From Theorem D4, it suffices to prove: $([\kappa](QDL\ \kappa))\rightarrow([(QDL\ \kappa)](QDL^*\ \kappa))$
Unfolding QDL* gives:$([\kappa](QDL\ \kappa))\rightarrow([(QDL\ \kappa)]\exists p(p\wedge([\kappa]p)\wedge([p]\Gamma)\wedge\wedge_i\forall\xi_i((([p]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow([p]\chi_i))))$
Since p and κ are not in ξ and p is not free in Γ, α_i, β_{ij}, and χ_i, letting p be (QDL κ) gives:
$([\kappa](QDL\ \kappa))\rightarrow$
$([(QDL\ \kappa)]((QDL\ \kappa)\wedge([\kappa](QDL\ \kappa))\wedge([(QDL\ \kappa)]\Gamma)\wedge\wedge_i\forall\xi_i((([[(QDL\ \kappa)]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow([(QDL\ \kappa)]\chi_i))))$
which holds by D1, D2, and the hypothesis. QED.

5. Relationship between QRL and QDL

The following theorems characterize the relationship between QDL and QRL:
RD1: $(\kappa\equiv(QDL\ \kappa))\rightarrow[\kappa](QRL\ \kappa)$
proof: Unfolding the definition of QRL gives: $(\kappa\equiv(QDL\ \kappa))\rightarrow[\kappa](\Gamma\wedge\wedge_i\forall\xi_i((([\kappa]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow\chi_i))$
Since κ is not in ξ, pushing $[\kappa]$ to lowest scope using the laws of KU45 modal logic on $[\kappa]$ gives:
$(\kappa\equiv(QDL\ \kappa))\rightarrow\ (([\kappa]\Gamma)\wedge\wedge_i\forall\xi_i((([\kappa]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow([\kappa]\chi_i)))$

Since κ is not in ξ, using the hypothesis to replace the first κ in the conclusion by (QDL κ) gives $[(QDL\ \kappa)]\Gamma$ which by theorem D1 is true. It remains only to prove: $(\kappa\equiv(QDL\ \kappa))\rightarrow((([\kappa]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow([\kappa]\chi_i))$

Since κ is not in ξ_i, replacing two occurrences of κ by using the hypothesis and then dropping the hypothesis gives: $(([(QDL\ \kappa)]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow([(QDL\ \kappa)]\chi_i)$ which by theorem D2 is true. QED.

RD2: $(\kappa\equiv(QDL\ \kappa))\rightarrow[(QRL\ \kappa)]\kappa$

proof: Using the hypothesis to replace the entailed κ in the conclusion gives: $(\kappa\equiv(QDL\ \kappa))\rightarrow[(QRL\ \kappa)](QDL\ \kappa)$

Unfolding QDL in the conclusion gives:

$(\kappa\equiv(QDL\ \kappa))\rightarrow[(QRL\ \kappa)]\exists p(p\wedge([p]\Gamma)\wedge\wedge_i\forall\xi_i((([p]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow([p]\chi_i)))$

Since p and κ are not in ξ and p is not free in Γ, α_i, β_{ij}, and χ_i, letting p be (QRL κ) gives:

$(\kappa\equiv(QDL\ \kappa))\rightarrow[(QRL\ \kappa)]((QRL\ \kappa)\wedge([(QRL\ \kappa)]\Gamma)\wedge\wedge_i\forall\xi_i((([(QRL\ \kappa)]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow([(QRL\ \kappa)]\chi_i)))$

The hypothesis $\kappa\equiv(QDL\ \kappa)$ and RD1 imply $([\kappa](QRL\ \kappa))$ which, since κ is not in ξ, allows the above sentence to be generalized to:

$(\kappa\equiv(QDL\ \kappa))\rightarrow[(QRL\ \kappa)]((QRL\ \kappa)\wedge([(QRL\ \kappa)]\Gamma)\wedge\wedge_i\forall\xi_i((([\kappa]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow([(QRL\ \kappa)]\chi_i))$

which by RL1 and RL2 is true. QED.

From RD1 and RD2 we may infer that every solution of the reflective equivalence of Quantified Default Logic is a solution of the equivalence for Quantified Reflective Logic:[8]

RD3: $(\kappa\equiv(QDL\ \kappa))\rightarrow(\kappa\equiv(QRL\ \kappa))$

It also follows that every solution to Quantified Reflective Logic entails (QDL κ).

RD4: $([\kappa](QRL\ \kappa))\rightarrow[\kappa](QDL\ \kappa)$

proof: Unfolding the definition of QDL gives:

$([\kappa](QRL\ \kappa))\rightarrow[\kappa]\exists p(p\wedge([p]\Gamma)\wedge\wedge_i\forall\xi_i((([p]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow([p]\chi_i)))$

Since p and κ are not in ξ_i and p is not free in Γ, α_i, β_{ij}, and χ_i, letting p be κ gives:

$([\kappa](QRL\ \kappa))\rightarrow[\kappa](\kappa\wedge([\kappa]\Gamma)\wedge\wedge_i\forall\xi_i((([\kappa]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow([\kappa]\chi_i)))$

Since κ is not in ξ_i, using the hypothesis to replace two occurrences of κ by (QRL κ) gives the generalization:

$([\kappa](QRL\ \kappa))\rightarrow[\kappa](\kappa\wedge([(QRL\ \kappa)]\Gamma)\wedge\wedge_i\forall\xi_i((([\kappa]\alpha_i)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow([(QRL\ \kappa)]\chi_i)))$

which is true by RL1 and RL2. QED

From RD3 and RD4 we may infer that the solutions to QDL are precisely those solutions to QRL which are entailed by (QDL κ):

RD5: $(\kappa\equiv(QDL\ \kappa))\leftrightarrow((\kappa\equiv(QRL\ \kappa))\wedge([(QDL\ \kappa)]\kappa))$

Likewise since $\kappa\equiv(QRL\ \kappa)$ in RD5 implies $([\kappa](QDL\ \kappa))$ by RD3 and since $([\kappa](QDL\ \kappa))$ implies that (QDL κ) is logically equivalent to (QDL* κ) by D5, it follows that:

RD6: $(\kappa\equiv(QDL\ \kappa))\leftrightarrow((\kappa\equiv(QRL\ \kappa))\wedge([(QDL^*\ \kappa)]\kappa))$

RD6 characterizes the relationship between QRL and QDL in terms of $[(QDL^*\ \kappa)]\kappa$. We now show that $([(QDL^*\ \kappa)]\kappa)$ is equivalent to the notion of being constructive, defined as follows: a Reflectivec solution κ is constructive iff it is not the case that there exists a proposition which satisfies the following four conditions: (1) κ entails that

---

[8] [Konolige 1987] previously proved at the syntactic level (i.e. using the set theoretic formulation rather than the Z modal logic formulation used herein) the analogous subcase of this theorem relating Autoeoistemic Logic and Default Logic where α, β_{ij}, and χ_i are closed sentences of First Order Logic and where Γ is a set of closed sentences of First Order Logic. By "closed" it is meant that no variable may occur free in these sentences. Thus that result does not explain the relationship when quantified variables cross modal scopes.

proposition, (2) the proposition does not entail κ, (3) the proposition entails Γ, and (4) for each i and for all ξ the proposition entails the conclusion χ_i of each default whose presupposition α_i is entailed by that proposition and whose β_{ij} formulas are possible with κ.

RD7: (Constructive $\kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i$) =df $\neg\exists p(([\kappa]p)\wedge(\neg([p]\kappa))\wedge([p]\Gamma)\wedge\wedge_i\forall\xi((([p]\alpha_i)\wedge(\wedge_{j=1,mi}(<\kappa>\beta_{ij})))\rightarrow([p]\chi_i)))$

RD8: $([(QDL^*\ \kappa)]\kappa)\leftrightarrow$(Constructive κ)

proof: Unfolding the $(QDL^*\ \kappa)$ in $([(QDL^*\ \kappa)]\kappa)$ gives:
$[\exists p(p\wedge([p]\Gamma)\wedge([\kappa]p)\wedge\wedge_i\forall\xi_i((([p]\alpha_i)\wedge(\wedge_{j=1,mi}(<\kappa>\beta_{ij})))\rightarrow([p]\chi_i)))]\kappa$

Pulling $\exists p$ out of the hypothesis of the entailment gives:

$\forall p((\wedge_i\forall\xi_i((([p]\alpha_i)\wedge(\wedge_{j=1,mi}(<\kappa>\beta_{ij})))\rightarrow([p]\chi_i))\wedge([p]\Gamma)\wedge([\kappa]p))\rightarrow([p]\kappa))$

Pushing a negation through the formula gives:

$\neg\exists p(([\kappa]p)\wedge(\neg([p]\kappa))\wedge([p]\Gamma)\ \wedge\wedge_i\forall\xi_i((([p]\alpha_i)\wedge(\wedge_{j=1,mi}(<\kappa>\beta_{ij})))\rightarrow([p]\chi_i)))$

which is the definition of being constructive. QED.

Being constructive isequivalent to the notion of being strongly grounded. A Quantified Reflective solution κ is strongly grounded iff it is not the case that there exists a proposition which satisfies the following four conditions: (1) κ entails that proposition, (2) the proposition does not entail κ, (3) the proposition entails Γ, and (4) for each i and all ξ the proposition entails the conclusion χ_i of each default whose β_{ij} formulas are also possible with κ in addition to being such that the default's presupposition α_i is entailed by that proposition and the default's β_{ij} formulas are possible with that proposition:[9]

RD9: (Strongly-grounded $\kappa\ \Gamma\ \alpha_i{:}\beta_{ij}/\chi_i$) =df

$\neg\exists p(([\kappa]p)\wedge(\neg([p]\kappa))\wedge([p]\Gamma)\ \wedge\wedge_i\forall\xi((\wedge_{j=1,mi}(<\kappa>\beta_{ij}))\rightarrow((([p]\alpha_i)\wedge(\wedge_{j=1,mi}(<p>\beta_{ij})))\rightarrow([p]\chi_i))))$

RD10: (Constructive κ)$\leftrightarrow$(Strongly-grounded κ)

proof: Unfolding Strongly-grounded gives:

$\neg\exists p(([\kappa]p)\wedge(\neg([p]\kappa))\wedge([p]\Gamma)\wedge\wedge_i\forall\xi_i((\wedge_{j=1,mi}(<\kappa>\beta_{ij}))\rightarrow((([p]\alpha_i)\wedge(\wedge_{j=1,mi}(<p>\beta_{ij})))\rightarrow([p]\chi_i))))$

Since $([\kappa]p)$, $(<\kappa>\beta_{ij})$ implies $(<p>\beta_{ij})$). Since p and κ do not occur in ξ, the above sentence is equivalent to:

$\neg\exists p(([\kappa]p)\wedge(\neg([p]\kappa))\wedge([p]\Gamma)\wedge\wedge_i\forall\xi_i((\wedge_{j=1,mi}(<\kappa>\beta_{ij}))\rightarrow((([p]\alpha_i)\wedge\#t)\rightarrow([p]\chi_i))))$

or rather: $\neg\exists p(([\kappa]p)\wedge(\neg([p]\kappa))\wedge([p]\Gamma)\wedge\wedge_i\forall\xi_i((([p]\alpha_i)\wedge(\wedge_{j=1,mi}(<\kappa>\beta_{ij})))\rightarrow([p]\chi_i)))$

which is the definition of being constructive. QED.

The above theorems give five characterizations of QDL in terms of QRL:[10]

RD11: All the following are equivalent:

(1) $\kappa\equiv$(QDL κ), (2) $(\kappa\equiv$(QRL κ))$\wedge(\kappa\equiv$(QDL κ)), (3) $(\kappa\equiv$(QRL κ))$\wedge$([(QDL κ)]κ), (4) $(\kappa\equiv$(QRL κ))$\wedge$([(QDL* κ)]κ), (5) $(\kappa\equiv$(QRL κ))$\wedge$(Constructive κ), (6) $(\kappa\equiv$(QRL κ))$\wedge$(Strongly-grounded κ)

---

[9] This notion of being strongly grounded is a generalization of that given in [Konolige 1987b] whereby variables may cross modal scopes. When no variables cross modal scopes as in [Konolige 1987b] this concept may be defined in terms of autoepistemic kernels in set theory as:

(strongly-grounded k) =d $\neg\exists p((k\supseteq p)\wedge(\neg(p\supseteq k))\wedge(p\supseteq(\text{folth } p))\wedge(p\supseteq\Gamma)\wedge\wedge_i((\wedge_{j=1,mi}((\neg\beta_{ij})\not\in k))\rightarrow(((\alpha_i\varepsilon p)\wedge\wedge_{j=1,mi}((\neg\beta_{ij})\not\in p))\rightarrow(\chi_i\varepsilon p))))$. The notion of being strongly grounded may also be defined in terms of the entire autoepistemic fixedpoint as was done in [Konolige 1987b] and [Antoniou 1997].

[10] [Konolige 1987b] previously proved at the syntactic level the equivalence of (1) and (6) relating Autoepistemic logic and Default Logic where α, β_{ij}, and χ_i are closed sentences of First Order Logic and where Γ is a set of closed sentences of First Order Logic. By "closed" it is meant that no variable may occur free in these sentences. Thus that result does not characterize the relationship when quantified variables cross modal scopes. A modern version of that result is described in [Antoniou 1997].

proof: The second formula follows from RD3, the third from RD5, the fourth from RD6, the fifth from RD8 and the sixth from RD10. QED.

Having shown that the Quantified Default solutions are the strongly grounded Quantified Reflective solutions, it is now shown that being strongly grounded essentially applies only to the defaults with entailment conditions since if there are essentially no entailment conditions in the defaults (i.e., α_i is #t for every ith default since #t is entailed by anything) then the Quantified Default solutions are precisely the Quantified Reflective solutions:

RD12: $(QDL\ \kappa\ \Gamma\ \#t{:}\beta_{ij}/\chi_i) \equiv (QRL\ \kappa\ \Gamma\ \#t{:}\beta_{ij}/\chi_i)$

proof: Unfolding $(QDL\ \kappa\ \Gamma\ \#t{:}\beta_{ij}/\chi_i)$ gives: $\exists p(p\wedge([p]\Gamma)\wedge\wedge_i\forall\xi_i(((([p]\#t)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow[p]\chi_i))$

which simplifies to: $\exists p(p\wedge([p]\Gamma)\wedge\wedge_i\forall\xi_i((\wedge_{j=1,m_i}(<\kappa>\beta_{ij}))\rightarrow([p]\chi_i)))$

Since p does not occur in ξ, the KU45 modal laws of [p] allow it to be pulled out giving:

$\exists p(p\wedge([p]\ (\Gamma\wedge\wedge_i\forall\xi((\wedge_{j=1,m_i}(<\kappa>\beta_{ij}))\rightarrow\chi_i))))$ which is: $\Gamma\wedge\wedge_i\forall\xi_i((\wedge_{j=1,m_i}(<\kappa>\beta_{ij}))\rightarrow\chi_i)$

which may be rewritten as: $\Gamma\wedge\wedge_i\forall\xi_i(((([\kappa]\#t)\wedge(\wedge_{j=1,m_i}(<\kappa>\beta_{ij})))\rightarrow\chi_i)$ which is: $(QRL\ \kappa\ \Gamma\ \#t{:}\beta_{ij}/\chi_i)$. QED.

6. Conclusion

Theorem RD11 shows that the solutions to Quantified Default Logic (i.e., QDL) are precisely the strongly grounded solutions to Quantified Reflective Logic (i.e., QRL). These results apply where variables cross modal scopes in any combination of the following two cases:

(1) where variables are universally quantified precisely over the scope of a default (or equivalently across the scope of all defaults and the initial theory Γ since they are connected by conjunction and since the universal quantifier commutes with conjunction),

(2) where variables are not quantified within the scope of the reflective equivalence in which case they are free within the scope of the theorem schemata proven herein and those schemata lie within the scope of any universal or existential quantification of such variables.

This paper does not address the important case where existential quantification occurs precisely over the scope of one or more defaults nor more complicated systems whereby quantifiers and modal symbols are nested in complex ways. (It is noted, however, that [Brown 1978] showed how an additional modal axiom allows modal scopes can be reduced to a depth of one even in the presence of quantifiers.)

This paper has not addressed automatic deduction systems for QDL and QRL, but there is the obvious point that theorems AD11 and AD12 suggest that a good deduction system for one logic may form the basis for a deduction system for the other logic. In particular, a deduction system that produced the QRL solutions could be used to produce the QDL solutions by checking which of those solutions satisfied a supporting condition (e.g. being strongly grounded) in AD11. The cost of checking a solution once it is produced would seem to be less than the cost of mechanically computing it.

Acknowledgements

This research was supported by National Science Foundation grants: #9818341 and #9972843.

Bibliography

[Anatoniou 1997] Antoniou, Grigoris 1997. *NonMonotonic Reasoning*, MIT Press.

[Bressan 1972] Bressan, Aldo 1972. *A General Interpreted Modal Calculus,* Yale University Press.

[Brown 1978] Brown,Frank M. 1978. "A Sequent Calculus for Modal Quantificational Logic", *3rd AISB/GI Conference Proceedings,* Hamburg, July 1978.

[Brown 1979] Brown, Frank M. 1979. "A Theorem Prover for Meta theory", *Proceedings Fourth Workshop on Automated Deduction*, Austin, Texas.

[Brown 1986] Brown, Frank M. 1986. "Reasoning in a Hierarchy of Deontic Defaults", *Proceedings of the Canadian Artificial Intelligence Conference CSCI 86*, Montreal, Canada, Morgan-Kaufmann, Los Altos.

[Brown 1987] Brown, Frank M. 1987. "The Modal Logic Z", In *The Frame Problem in AI"; Proc. of the 1987 AAAI Workshop*, Morgan Kaufmann, Los Altos, CA .
[Brown 1989] Brown, Frank M. 1989. "The Modal Quantificational Logic Z Applied to the Frame Problem", advanced paper *First International Workshop on Human & Machine Cognition*, May 1989 Pensacola, Florida. Abbreviated version published in *International Journal of Expert Systems Research and Applications, Special Issue: The Frame Problem. Part A.* eds. Keneth Ford and Pattrick Hayes, vol. 3 number 3, pp169-206 JAI Press 1990. Reprinted in *Reasoning Agents in a Dynamic World: The Frame problem*, editors: Kenneth M. Ford, Patrick J. Hayes, JAI Press 1991.
[Brown 2003a] Frank M, Brown, "Representing Reflective Logic in Modal Logic", **submitted to this conference**.
[Brown 2003b] Frank M, Brown, "Representing Default Logic in Modal Logic", **submitted to this conference**.
[Carnap 1946] Carnap, Rudolf 1946. "Modalities and Quantification" *Journal of Symbolic Logic*, vol. 11, number 2.
[Carnap 1956] Carnap, Rudolf 1956. *Meaning and Necessity: A Study in the Semantics of Modal Logic*, The University of Chicago Press.
[Cocchiarella 1984] Cocchiarella, N. B. 1984. "Philosophical Perspectives on Quantification in Tense and Modal Logic", *Handbook of Philosophical Logic*, D. Reidel.
[Fine 1970] Fine, K. 1970. "Propositional Quantifiers in Modal Logic" *Theoria* 36, p336--346.
[Hendry & Pokriefka 1985] Hendry, Herbert E. & Pokriefka, M. L. 1985. "Carnapian Extensions of S5", *Journal of Phil. Logic 14*.
[Hughes & Cresswell 1968] Hughes, G. E. & Creswell, M. J., 1968. *An Introduction to Modal Logic*, Methuen & Co. Ltd., London.
[Konolige 1987a] Konolige, Kurt 1987. "On the Relation Between Default Theories and Autoepistemic Logic", *IJCAI87 Proceedings of the Tenth International Joint Conference on Artificial Intelligence*, 1987.
[Konolige 1987b] Konolige, Kurt 1987b. "On the Relation Between Default Theories and Autoepistemic Logic", personally circulated new version of IJCAI87 paper correcting it to account for a counterexample found by Gelfond and Przymusinska.
[Lewis 1936] Lewis, C. I. 1936. Strict Implication, *Journal of Symbolic Logic*, vol I.
[Mendelson 1964] Mendelson, E. 1964. *Introduction to Mathematical Logic*, Van Norstrand, Reinhold Co., New York.
[Parks q976] Parks, Z. 1976. "An Investigation into Quantified Modal Logic", Studia Logica 35, p109-125.
[Reiter 1980] Reiter, R. 1980. "A Logic for Default Reasoning" *Artificial Intelligence*, 13.

Author information

Frank M. Brown- Artificial Intelligence Laboratory, University of Kansas, Lawrence, Kansas, 66045, e-mail: brown@ku.edu.

REPRESENTING AUTOEPISTEMIC LOGIC IN MODAL LOGIC

Frank M. Brown

***Abstract**: The nonmonotonic logic called Autoepistemic Logic is shown to be representable in a monotonic Modal Quantificational Logic whose modal laws are stronger than S5. Specifically, it is proven that a set of sentences of First Order Logic is a fixedpoint of the fixedpoint equation of Autoepistemic Logic with an initial set of axioms if and only if the meaning or rather disquotation of that set of sentences is logically equivalent to a particular modal functor of the meaning of that initial set of sentences. This result is important because the modal representation allows the use of powerful automatic deduction systems for Modal Logic and unlike the original Autoepistemic Logic, it is easily generalized to the case where quantified variables may be shared across the scope of modal expressions thus allowing the derivation of quantified consequences. Furthermore, this generalization properly treats such quantifiers since both the Barcan formula and its converse hold*

***Keywords**: Autoepistemic Logic, Modal Logic, Nonmonotonic Logic.*

1. Introduction

One of the most well known nonmonotonic logics [Antoniou 1997] which inherently deals with entailment conditions in addition to possibility conditions in its sentences is the so-called Autoepistemic Logic [Moore 1985][11]. The basic idea of Autoepistemic Logic is that there is a set of axioms Γ and for every closed sentence χ there are two non-logical "inference rules" of the forms:

$$\frac{\chi:}{L'\chi} \qquad \frac{:\neg\chi}{\neg L'\chi}$$

where the predicate symbol L intuitively means that its argument names a sentence which is inferable. The first rule suggests that L $'\chi$ may be inferred from χ and the second rule suggests that $\neg L'\chi$ may be inferred if $\neg\chi$ is not inferable. When L is in Γ such "inference rules" maybe circular in that determining if they are applicable depends on the inferability or noninferability of χ which in turn depends on what else was derivable. Thus, tentatively applying such inference rules by checking whether χ has been or has not yet been inferred produces consequences which may later have to be retracted. For this reason valid inferences in a nonmonotonic logic such as Autoepistemic Logic are essentially carried out not in the original nonmonotonic language, but rather in some (monotonic) metatheory in which that nonmonotonic logic is defined. [Moore 1985; Konolige 1987; Konolige 1987b] explicated the above intuition by defining Autoepistemic Logic in terms of the set theoretic proof theory metalanguage of a First Order Logic (i.e. FOL) object language with the fixed point equation:

$$'\kappa=(\text{ael } '\kappa\ '\Gamma\})$$

where ael is defined as: $(\text{ael } '\kappa\ '\Gamma)=\text{df}(\text{fol}('\Gamma\cup\{'(L\ '\chi_i):'\chi_i\varepsilon'\kappa\}\cup\{'(\neg(L\ '\chi_i)):'\chi_i\notin'\kappa\}))$

where $'\chi_i$ is the ith sentence of the FOL object language and where $'\kappa$ and $'\Gamma$ are sets of closed sentences of the FOL object language. A closed sentence is a sentence without any free variables. fol is a function which produces the set of theorems derivable in FOL from the set of sentences to which it is applied. The quotations appended to the front of these Greek letters indicate references in the metalanguage to the sentences of the FOL object language. Interpreted doxastically this fixed point equation states:

> the set of closed sentences which are believed is equal to
> the set of theorems derivable by the laws of FOL
> from the union of
> the set of closed sentences $'\Gamma$,
> the set of all closed sentences of the form: $'(L\ '\chi_i)$ for each i such that $'\chi_i$ is believed,
> and the set of all closed sentences of the form: $'(\neg(L\ '\chi_i))$ for each i such that $'\chi_i$ is not believed.

The purpose of this paper is to show that all this metatheoretic machinery including the formalized syntax of FOL, the proof theory of FOL, the axioms of a strong set theory, and the set theoretic fixedpoint equation is not needed and that the essence of Autoepistemic Logic is representable as a necessary equivalence in a (monotonic) Modal Quantificational Logic. Interpreted as a doxastic logic this equivalence states:

> that which is believed is equivalent to: Γ and for all i $(L\ '\chi_i)$ if and only if χ_i is believed.

thereby eliminating the metatheoretic machinery.[12]

The remainder of this paper proves that this modal representation is equivalent to Autoepistemic Logic. Section 2 describes a formalized syntax for a FOL object language. Section 3 describes the part of the proof theory of FOL needed herein (i.e. theorems FOL1-FOL4). Section 4 describes the Intensional Semantics of FOL which includes laws giving the meaning of FOL sentences: M0-M8, theorems giving the meaning of sets of sentences: MS1, MS2, MS3, and laws specifyingthe relationship of meaning and modality to the proof theory of FOL (i.e. the laws R0, A1, A2, and A3 and the theorems: C1, C2, C3, and C4). The modal version of Autoepistemic Logic is defined in section 5 and explicated with theorems MA1-MA6 and SS1-SS2. In section 6, this modal version is shown by

[11] Autoepistemic Logic may be viewed as an improved version of the systems described in [McDermott 1980; McDermott 1982].

[12] The occurrence of quotation in the argument to L may be replaced by using a new symbol *L* such that (*L*) replaces (L ').

theorems AEL1 and AEL2 to be equivalent to the set theoretic fixed point equation for Autoepistemic Logic. Figure 1 outlines the relationship of all these theorems in producing the final theorems AEL2, FOL4, and MA6. Finally, in section 7, some consequences of these results are discussed.

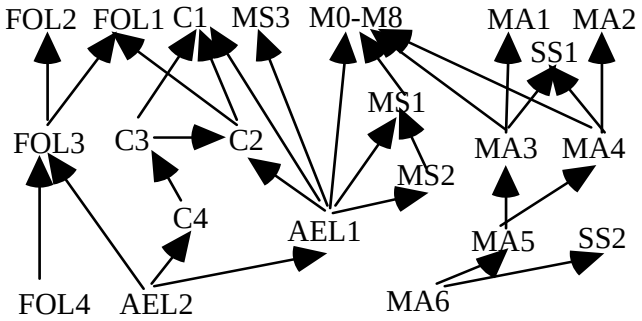


Figure 1: Dependencies among the Theorems

2. Formal Syntax of First Order Logic

We use a First Order Logic (i.e. FOL) defined as the six tuple: ($\rightarrow$, #f, $\forall$, *vars*, *predicates*, *functions*) where $\rightarrow$, #f, and $\forall$ are logical symbols, *vars* is a set of variable symbols, *predicates* is a set of predicate symbols each of which has an implicit arity specifying the number of associated terms, and *functions* is a set of function symbols each of which has an implicit arity specifying the number of associated terms. The sets of logical symbols, variables, predicate symbols, and function symbols are pairwise disjoint. Lower case Roman letters possibly indexed with digits are used as variables. Greek letters possibly indexed with digits or lower case roman letters are used as syntactic metavariables. γ, γ_1,...γ_n, range over the variables, ξ, ξ_1...ξ_n range over sequences of variables of an appropriate arity, π,π_1...π_n range over the predicate symbols, ϕ, ϕ_1...ϕ_n range over function symbols, δ, δ_1...δ_n, σ range over terms, and α, α_1...α_n, β, β_1...β_n,χ, χ_1...χ_n, Γ_1,...Γ_n,φ range over sentences. The terms are of the forms γ and (ϕ δ1...δn), and the sentences are of the forms ($\alpha\rightarrow\beta$), #f, ($\forall\gamma$ α), and (π δ_1...δ_n). A nullary predicate π or function ϕ is written as a sentence or a term without parentheses. $\varphi\{\pi/\lambda\xi\alpha\}$ represents the replacement of all occurrences of π in φ by $\lambda\xi\alpha$ followed by lambda conversion. The primitive symbols are shown in Figure 2 with their intuitive interpretations.

Symbol	Meaning
$\alpha\rightarrow\beta$	if α then β.
#f	falsity
$\forall\gamma\ \alpha$	for all γ, α.

Figure 2: Primitive Symbols of First Order Logic

The defined symbols are listed in Figure 3 with their definitions and intuitive interpretations.

Symbol	Definition	Meaning		Symbol	Definition	Meaning
$\neg\alpha$	$\alpha\rightarrow$ #f	not α		$\alpha\wedge\beta$	$\neg(\alpha\rightarrow\neg\beta)$	α and β
#t	$\neg$ #f	truth		$\alpha\leftrightarrow\beta$	$(\alpha\rightarrow\beta)\wedge(\beta\rightarrow\alpha)$	α if and only if β
$\alpha\vee\beta$	$(\neg\alpha)\rightarrow\beta$	α or β		$\exists\gamma\ \alpha$	$\neg\forall\gamma\neg\alpha$	for some γ , α

Figure 3: Defined Symbols of First Order Logic

The particular FOL used herein includes the predicate symbol L and a denumerably infinite number of 0-ary function symbols representing the names (i.e. '□) of the sentences (i.e. □) of this First Order Logic. The FOL object language expressions are referred in the metalanguage (which also includes a FOL syntax) by inserting a quote sign in front of the object language entity thereby making a structural descriptive name of that entity. In addition to referring to object language sentences, the formalized metalanguage also needs to refer to sets of sentences

of FOL. Generally, a set of sentences is represented as: {'Γ_i} which is defined as: {'Γ_i: #t} which in turn is defined as: {s: $\exists$i(s='Γ_i)} where i ranges over some range of numbers (which may be finite or non-infinite). With a slight abuse of notation we also write 'κ, 'Γ to refer to such sets.

3. Proof Theory of First Order Logic

First Order Logic (i.e. FOL) is axiomatized with a recursively enumerable set of theorems as the set of axioms is itself recursively enumerable and its inference rules are recursive. The axioms and inference rules of FOL [Mendelson 1964] are those given in Figure 4. They form a standard set of axioms and inference rules for FOL.

MA1: $\alpha \rightarrow (\beta \rightarrow \alpha)$ MR1: from α and $(\alpha \rightarrow \beta)$ infer β
MA2: $(\alpha \rightarrow (\beta \rightarrow \rho)) \rightarrow ((\alpha \rightarrow \beta) \rightarrow (\alpha \rightarrow \rho))$ MR2: from α infer $(\forall\gamma\ \alpha)$
MA3: $((\neg\ \alpha) \rightarrow (\neg\ \beta)) \rightarrow (((\neg\ \alpha) \rightarrow \beta) \rightarrow \alpha)$
MA4: $(\forall\gamma\ \alpha) \rightarrow \beta$ where β is the result of substituting an expression (which is free for the free positions of γ in α) for all the free occurrences of γ in α.
MA5: $((\forall\gamma(\alpha \rightarrow \beta)) \rightarrow (\alpha \rightarrow (\forall\gamma\ \beta)))$ where γ does not occur in α.

Figure 4: Inferences Rules and Axioms of FOL

In order to talk about sets of sentences we include in the metatheory set theory symbolism as developed along the lines of [Quine 1976]. This set theory includes the symbols ε, $\notin$, $\supseteq$, =, $\cup$ as is defined therein.

The derivation operation (i.e. fol) of any First Order Logic obeys the Inclusion (i.e. FOL1) and Idempotence (i.e. FOL2) properties:

FOL1: (fol 'κ)$\supseteq$'κ Inclusion

FOL2: (fol 'κ)$\supseteq$(fol(fol 'κ)) Idempotence

From these two properties we prove:

FOL3: (rl 'κ 'Γ 'α_i:'β_{ij}/'χ_i))=(fol(rl 'κ 'Γ 'α_i:'β_{ij}/'χ_i))

proof: FOL1 and FOL2 imply that (fol(fol 'κ))=(fol 'κ). Since rl begins with fol this implies: 'κ=(fol(rl 'κ)) QED.

FOL4: ('κ=(rl 'κ 'Γ 'α_i:'β_{ij}/'χ_i))$\rightarrow$('κ=(fol 'κ))

proof: From the hypothesis and FOL3: 'κ=(fol(rl 'κ 'Γ 'α_i:'β_{ij}/'χ_i)) is derived. Using the hypothesis to replace (rl 'κ 'Γ 'α_i:'β_{ij}/'χ_i) by 'κ in this result gives: 'κ=(fol 'κ). QED.

4. Intensional Semantics of FOL

The meaning (i.e. mg) [Brown 1978, Boyer&Moore 1981] or rather disquotation of a sentence of First Order Logic (i.e. FOL) is defined to satisfy the laws given in Figure 5 below mg is defined in terms of mgs which maps each FOL object language sentence and an association list into a meaning. Likewise, mgn maps a FOL object language term and an association list into a meanings. An association list is simply a list of pairs consisting of an object language variable and the meaning to which it is bound.

M0: (mg 'α) =df (mgs '($\forall\gamma$1...γn α)'()) where 'γ1...'γn are all the free variables in 'α
M1: (mgs '($\alpha \rightarrow \beta$)a) $\leftrightarrow$ ((mgs 'α a)$\rightarrow$(mgs 'β a))
M2: (mgs '#f a) $\leftrightarrow$ #f
M3: (mgs '($\forall\ \gamma\ \alpha$)a) $\leftrightarrow$ $\forall$x(mgs 'α(cons(cons 'γ x)a))
M4: (mgs '($\pi\ \delta_1...\delta_n$)a) $\leftrightarrow$ (π(mgn 'δ_1 a)...(mgn 'δ_n a)) for each predicate symbol 'π.
M5: (mgn '($\phi\ \delta_1...\delta_n$)a) = ($\phi$(mgn '$\delta_1$ a)...(mgn 'δ_n a)) for each function symbol 'ϕ.
M6: (mgn 'γ a) = (lookup 'γ a) where (lookup 'γ a) is the value associated wit 'γ in the association list a.

Figure 5: The Meaning of FOL Sentences

The meaning of a set of sentences is defined in terms of the meanings of the sentences in the set as:

(ms 'κ) =df ∀s((sε'κ)→(mg s))

MS1: (ms{'α: Γ}) ↔ ∀ξ(Γ→α) where ξ is the sequence of all the free variables in 'α and where Γ is any sentence of the intensional semantics.

proof: (ms{'α:Γ}) Unfolding ms and the set pattern abstraction symbol gives: ∀s((sε{s: ∃ξ((s='α)∧Γ)})→(mg s))

where ξ is a sequence of the free variables in 'a. This is equivalent to: ∀s((∃ξ((s='α)∧Γ)))→(mg s))

which is logically equivalent to: ∀s∀ξ (((s='κ)∧Γ)→(mg s)) which is equivalent to: ∀ξ(Γ→(mg 'α))

Unfolding mg using M0-M7 then gives: ∀ξ(Γ→α) QED

The meaning of the union of two sets of FOL sentences is the conjunction of their meanings (i.e. MS1) and the meaning of a set is the meaning of all the sentences in the set (i.e. MS2):

MS2: (ms{'Γi}) ↔ ∀i∀ξiΓi

proof: (ms{'Γi}) Unfolding the set notation gives: (ms{'Γi: #t})

By MS1 this is equivalent to: ∀i∀ξi(#t→Γi) which is equivalent to: ∀iΓi QED.

MS3: (ms('κ∪'Γ)) ↔ ((ms 'κ)∧(ms 'Γ))

proof: Unfolding ms and union in: (ms('κ∪'Γ)) gives: ∀s((sε{s: (sε'κ)∨(sε'Γ)})→(mg s)) or rather:

∀s(((sε'κ)∨(sε'Γ))→(mg s)) which is logically equivalent to: (∀α((sε'κ)→(mg s)))∧(∀s((sε'Γ)→(mg s)))

Folding ms twice then gives:((ms 'κ)∧(ms 'Γ)) QED.

The meaning operation may be used to develop an Intensional Semantics for a FOL object language by axiomatizing the modal concept of necessity so that it satisfies the theorem:

C1: ('αε(fol 'κ)) ↔ ([] ((ms 'κ)→(mg 'α)))

for every sentence 'α and every set of sentences 'κ of that FOL object language. The necessity symbol is represented by a box: [] herein constructed from two square brackets. C1 states that a sentence of FOL is a FOL-theorem (i.e. fol) of a set of sentences of FOL if and only if the meaning of that set of sentences necessarily implies the meaning of that sentence. One modal logic which satisfies C1 is the Z Modal Quantificational Logic described in [Brown 1987; Brown 1989] whose theorems are recursively enumerable and which extends the weaker possibility axioms used in [Lewis 1936; Bressan 1972; Hendry & Pokriefka 1985].[13] We note that Z includes all the laws of S5 modal Logic [Hughes & Cresswell 1968] whose modal axioms and inference rules are given in Figure 6. κ and Γ represent arbitrary sentences of the intentional semantics.

R0: from α infer ([] κ) A2: ([](κ→ Γ)) → (([]κ)→ ([]Γ))

A1: ([]κ) → κ A3: ([]κ) ∨ ([]¬[]κ)

Figure 6: The Laws of S5 Modal Logic

These S5 modal laws and the laws of FOL given in Figure 4 constitute an S5 Modal Quantificational Logic similar to [Carnap 1946; Carnap 1956], and a FOL version [Parks 1976] of [Bressan 1972] in which the Barcan formula: (∀γ([]κ))→([]∀γκ) and its converse hold. The R0 inference rule implies that anything derivable in the metatheory is necessary. Thus, in any logic with R0, contingent facts would never be asserted as additional axioms of the metatheory. For example, we would not assert ([](κ↔Γ)) as an axiom and then try to prove ([](κ→α)). Instead we would try to prove that ([](κ↔Γ))→([](κ→α)).

The defined Modal symbols used herein are listed in Figure 7 with their definitions and interpretations.

[13] An S5 modal logic which satisfies a metatheorem analogous C1 for Propositional Logic is the system S5c given in [Hendry and Pokriefka 1985] which has axiom schemes stating that every conjunction of distinct propositional constants is logically possible. This extends the trivial possibility axiom that some proposition is neither #t nor #f used in [Lewis 1936; Bressan 1972]. A modal logic which satisfies C1 for FOL is the Z Modal Quantificational Logic described in [Brown 1987; Brown 1989] whose theorems are recursively enumerable. This logic has the metatheorem: (<>Γ){π/λξα}→(<>Γ) where Γ is a sentence of FOL.

Symbol	Definition	Meaning		Symbol	Definition	Meaning
<>κ	¬ [] ¬κ	α is logically possible		[κ] Γ	[] (κ→Γ)	β entails α
κ≡ Γ	[] (κ↔Γ)	α is logically equivalent to β		<κ> Γ	<> (κ∧Γ)	α and β is logically possible

Figure 7: Defined Symbols of Modal Logic

For example, folding the definition of entailment, C1 may be rewritten more compactly as:

C1': ('αε(fol 'κ)) ↔ ([(ms 'κ)](mg 'α))

This compact notation for entailment is used hereafter.

From the laws of the Intensional Semantics we prove that the meaning of the set of FOL consequences of a set of sentences is the meaning of that set of sentences (C2), the FOL consequences of a set of sentences contain the FOL consequences of another set if and only if the meaning of the first set entails the meaning of the second set (C3), and the sets of FOL consequences of two sets of sentences are equal if and only if the meanings of the two sets are logically equivalent (C4):

C2: (ms(fol 'κ))≡(ms 'κ)

proof: The proof divides into two cases:

(1) [(ms 'κ)](ms(fol 'κ))) Unfolding the second ms gives: [(ms 'κ)]∀s((sε(fol 'κ))→(mg s))

By the soundness part of C1 this is equivalent to: [(ms 'κ)]∀s(([(ms 'κ)](mg s))→(mg s))

By the S5 laws this is equivalent to: ∀s(([(ms 'κ)](mg s))→ [(ms 'κ)](mg s)) which is a tautology.

(2) [(ms(fol 'κ))](ms 'κ) Unfolding ms twice gives: [∀s((sε(fol 'κ))→(mg s))]∀s((sε'κ)→(mg s))

which is: [∀s((sε(fol 'κ))→(mg s))]((sε'κ)→(mg s)) Backchaining on the hypothesis and then dropping it gives: (sε'κ)→(sε(fol 'κ)). Folding ⊇ gives an instance of FOL1. QED.

C3: (fol 'κ)⊇(fol 'Γ) ↔ ([(ms 'κ)](ms 'Γ))

proof: Unfolding ⊇ gives: ∀s((sε(fol 'Γ))→(sε(fol 'κ)))

By C1 twice this is equivalent to: ∀s(([(ms 'Γ)](mg s))→([(ms 'κ)](mg s)))

By the laws of S5 modal logic this is equivalent to: ([(ms 'κ)]∀s(([(ms 'Γ)](mg s)))→(mg s)))

By C1 this is equivalent to: [(ms 'κ)]∀s((sε(fol 'Γ))→(mg s)). Folding ms then gives: [(ms 'κ)](ms(fol 'Γ))

By C2 this is equivalent to: [(ms 'κ)](ms 'Γ). QED.

C4: ((fol 'κ)=(fol 'Γ)) ↔ ((ms 'κ)≡(ms 'Γ))

proof: This is equivalent to (((fol 'κ)⊇(fol 'Γ))∧((fol 'Γ)⊇(fol 'κ))) ↔ ([(ms 'κ)](ms 'Γ))∧([(ms 'Γ)](ms 'κ))

which follows by using C3 twice.

5. Autoepistemic Logic Represented in Modal Logic

The fixed point equation for Autoepistemic Logic may be expressed in S5 Modal Quantificational Logic by the necessary equivalence:

$$\kappa \equiv (\text{AEL } \kappa\ \Gamma)$$

where AEL is defined as follows: (AEL κ Γ) =df Γ∧∀i((L 'χi)↔([κ]χi))

where χi is the ith sentence of the FOL object language.

Given below are some simple properties of AEL used to prove the equivalence of the proof theoretic and modal representations of Autoepistemic Logic. The first two theorems state that AEL entails Γ and that AEL entails for all i, (L 'χi) if and only if χi holds in κ.

MA1: [(AEL κ Γ)]Γ

proof: By R0 it suffices to prove: (AEL κ Γ)$\rightarrow\Gamma$. Unfolding AEL gives: ($\Gamma\wedge\forall$i((L 'χ_i)$\leftrightarrow$([κ]χ_i)))$\rightarrow\Gamma$ which is a tautology. QED.

MA2: [(AEL κ Γ)]$\forall$i((L 'χ_i)$\leftrightarrow$([κ]χ_i))

proof: By R0 it suffices to prove: (AEL κ Γ)$\rightarrow\forall$i((L 'χ_i)$\leftrightarrow$([κ]χ_i))

Unfolding AEL gives: [$\Gamma\wedge\forall$i((L 'χ_i)$\leftrightarrow$([κ]χ_i)))$\rightarrow\forall$i((L 'χ_i)$\leftrightarrow$([κ]χ_i))) which is a tautology. QED.

The concept (i.e. ss) of the combined meaning of all the sentences of the FOL object language whose meanings are entailed by a proposition is defined as follows: (ss κ) =df $\forall$s(([κ](mg s))$\rightarrow$(mg s)). SS1 shows that a proposition entails the combined meaning of the FOL object language sentences that it entails. SS2 shows that if a proposition is necessarily equivalent to the combined meaning of all the FOL object language sentences that it entails, then there exists a set of FOL object language sentences whose meaning is necessarily equivalent to that proposition:

SS1: [κ](ss κ)

proof: By R0 it suffices to prove: $\kappa\rightarrow$(ss κ). Unfolding ss gives: $\kappa\rightarrow\forall$s(([κ](mg s))$\rightarrow$(mg s))

which is equivalent to:$\forall$s(([κ](mg s))$\rightarrow$($\kappa\rightarrow$(mg s))) which is an instance of A1. QED.

SS2: ($\kappa\equiv$(ss κ))$\rightarrow$ $\exists$s($\kappa\equiv$(ms s))

proof: Letting s be {s: ([κ](mg s)) gives: ($\kappa\equiv$(ss κ))$\rightarrow$ ($\kappa\equiv$(ms{s: ([κ](mg s))))

Unfolding ms and lambda conversion gives: ($\kappa\equiv$(ss κ))$\leftrightarrow$ ($\kappa\equiv\forall$s(([κ](mg s))$\rightarrow$(mg s)))

Folding ss gives a tautology. QED.

Theorems MA3 and MA4 are analogous to MA1 and MA2 except that AEL is replaced by the combined meaning of all of the sentences entailed by AEL.

MA3: [ss(AEL κ $\forall$iΓ_i)]$\forall$iΓ_i

proof: By R0 it suffices to prove: (ss(AEL κ $\forall$iΓ_i))$\rightarrow\forall$iΓ_i

Unfolding ss gives: ($\forall$s(([(AEL κ $\forall$iΓ_i)](mg s))$\rightarrow$(mg s)))$\rightarrow\forall$iΓ_i

which is equivalent to: ($\forall$s(([(AEL κ $\forall$iΓ_i)](mg s))$\rightarrow$(mg s)))$\rightarrow\Gamma_i$

which by the meaning laws is equivalent to: ($\forall$s(([(AEL κ $\forall$iΓ_i)](mg s))$\rightarrow$(mg s)))$\rightarrow$(mg 'Γ_i)

Backchaining on (mg 'Γ_i) with s in the hypothesis assigned to be 'Γ_i in the conclusion shows that it suffices to prove: ([(AEL κ $\forall$iΓ_i)](mg 'Γ_i)) which by the meaning laws is equivalent to: ([(AEL κ $\forall$iΓ_i)]Γ_i)

which by the laws of S5 Modal Logic is equivalent to: ([(AEL κ $\forall$iΓ_i)]$\forall$iΓ_i) which is an instance of MA1. QED.

MA4: [(ss(AEL κ Γ))]$\forall$i((L 'χ_i)$\leftrightarrow$([κ]χ_i)))

proof: By R0 it suffices to prove: (ss(AEL κ Γ))$\rightarrow\forall$i((L 'χ_i)$\leftrightarrow$([κ]χ_i)))

which is equivalent to: (ss(AEL κ Γ))$\rightarrow$((([κ]χ_i)$\rightarrow$(L 'χ_i))$\wedge$((¬([κ]χ_i))$\rightarrow$(¬(L 'χ_i)))

Unfolding ss gives: ($\forall$s(([(AEL κ Γ)](mg s))$\rightarrow$(mg s)))$\rightarrow$((([κ]χ_i)$\rightarrow$(L 'χ_i))$\wedge$((¬([κ]χ_i))$\rightarrow$(¬(L 'χ_i))))

Letting the quantified s in the hypothesis have the two instances: '(L 'χ_i) and '(¬(L 'χ_i)) and then dropping that hypothesis gives:

((([(AEL κ Γ)](mg '(L 'χ_i)))$\rightarrow$(mg '(L 'χ_i)))$\wedge$(([(AEL κ Γ)](mg '(¬(L 'χ_i))))$\rightarrow$(mg '(¬(L 'χ_i)))))

$\rightarrow$((([κ]χ_i)$\rightarrow$(L 'χ_i))$\wedge$((¬([κ]χ_i))$\rightarrow$(¬(L 'χ_i))))

By the meaning laws M0-M8 this is equivalent to:

((([(AEL κ Γ)](L 'χ_i))$\rightarrow$(L 'χ_i))$\wedge$(([(AEL κ Γ)](¬(L 'χ_i)))$\rightarrow$(¬(L 'χ_i))))$\rightarrow$((([κ]χ_i)$\rightarrow$(L 'χ_i))$\wedge$((¬([κ]χ_i))$\rightarrow$(¬(L 'χ_i))))

Using these instances of the hypothesis to backchain on (L 'χ_i) and ($\neg$(L 'χ_i)) in the conclusion, and then dropping these instances gives:

((([κ]χ_i)$\rightarrow$([(AEL κ Γ)](L 'χ_i)))$\wedge$(($\neg$([κ]χ_i)))$\rightarrow$([(AEL κ Γ)]($\neg$(L 'χ_i))))

Using the laws of S5 Modal Logic then gives: ([(AEL κ Γ)]((([κ]χ_i)$\rightarrow$(L 'χ_i))$\wedge$(($\neg$([κ]χ_i))$\rightarrow$($\neg$(L 'χ_i))))

which is equivalent to: [(AEL κ Γ)]((L 'χ_i)$\leftrightarrow$([κ]χ_i)) which holds by MA2. QED.

Finally MA5 and MA6 show that talking about the meanings of sets of FOL sentences in the modal representation of Autoepistemic Logic is equivalent to talking about propositions in general.

MA5: (ss(AEL κ $\forall i\Gamma_i$)) $\leftrightarrow$ (AEL κ $\forall i\Gamma_i$)

proof: In view of SS1, it suffices to prove: (ss(AEL κ $\forall i\Gamma_i$))$\rightarrow$(AEL κ $\forall i\Gamma_i$)

Unfolding the second occurrence of AEL gives: (ss(AEL κ $\forall i\Gamma_i$))$\rightarrow$($\forall i\Gamma_i\wedge\forall i$((L '$\chi_i$)$\leftrightarrow$([$\kappa$]$\chi_i$)))

which holds by theorems MA3 and MA4. QED.

MA6: ($\kappa\equiv$(AEL κ $\forall i\Gamma_i$)))$\rightarrow\exists$s($\kappa\equiv$(ms s))

proof: ($\kappa\equiv$(ss(AEL κ $\forall i$(mg 'Γ_i)))) is derived from the hypothesis and MA5. Using the hypothesis to replace (AEL κ $\forall i$(mg 'Γ_i))) by κ in this result gives: ($\kappa\equiv$(ss κ)). By SS2 this implies the conclusion. QED.

6. Conclusion: Autoepistemic Logic represented in Modal Logic

The relationship between the proof theoretic definition of Autoepistemic Logic [Moore 1985] and the modal representation is proven in two steps. First theorem AEL1 shows that the meaning of the set ael is the proposition AEL and then theorem AEL2 shows that a set of FOL sentences which contains its FOL theorems is a fixedpoint of the fixedpoint equation of Autoepistemic Logic with an initial set of axioms if and only if the meaning (or rather disquotation) of that set of sentences is logically equivalent to AEL of the meanings of that initial set of sentences.

AEL1: (ms(ael(fol 'κ){'Γ_i}))$\equiv$(AEL(ms 'κ)($\forall i\Gamma_i$))

proof: By R0 it suffices to prove: (ms(ael(fol 'κ){'Γ_i}))$\leftrightarrow$(AEL(ms 'κ)Γ). The left side is: ms(ael(fol 'κ){'Γ_i})

Unfolding the definition of ael gives: ms(fol({'Γ_i}$\cup${'(L 'χ_i): '$\chi_i\varepsilon$(fol 'κ)}$\cup${'($\neg$(L 'χ_i)): '$\chi_i\notin$(fol 'κ)}))

By C2 this is equivalent to: ms({'Γ_i}$\cup${'(L 'χ_i): '$\chi_i\varepsilon$(fol 'κ)}$\cup${'($\neg$(L 'χ_i)): '$\chi_i\notin$(fol 'κ)})

Using C1 twice gives: ms({'Γ_i}$\cup${'(L 'χ_i): ([ms 'κ]χ_i)}$\cup${'($\neg$(L 'χ_i)): $\neg$([ms 'κ]χ_i)})

Using MS3 twice gives: (ms{'Γ_i})$\wedge$(ms{'(L 'χ_i): ([ms 'κ]χ_i)})$\wedge$(ms{'($\neg$(L 'χ_i)): $\neg$([ms 'κ]χ_i)})

Using MS2 gives: ($\forall i\Gamma_i$)$\wedge$(ms{'(L 'χ_i): ([ms 'κ]χ_i)})$\wedge$(ms{'($\neg$(L 'χ_i)): $\neg$([ms 'κ]χ_i)})

Applying MS1 twice gives: ($\forall i\Gamma_i$)$\wedge\forall i$((([ms 'κ]χ_i)$\rightarrow$(L 'χ_i))$\wedge\forall i$(($\neg$([ms 'κ]χ_i))$\rightarrow$($\neg$(L 'χ_i)))

which is logically equivalent to: ($\forall i\Gamma_i$)$\wedge\forall i$((L 'χ_i)$\leftrightarrow$([ms 'κ]χ_i))

Folding the definition of AEL gives: (AEL(ms 'κ)($\forall i\Gamma_i$)) QED.

AEL2: ((fol 'κ)=(ael(fol 'κ){'Γ_i}))$\leftrightarrow$((ms 'κ)$\equiv$(AEL(ms 'κ)($\forall i\Gamma_i$)))

proof: (fol 'κ)=(ael(fol 'κ){'Γ_i}). By FOL3 this is equivalent to: (fol 'κ)=(fol(ael(fol 'κ){'Γ_i}))

By C4 this is equivalent to: (ms 'κ)$\equiv$(ms(ael(fol 'κ){'Γ_i})).

By AEL1 this is equivalent to:(ms 'κ)$\equiv$(AEL(ms 'κ)($\forall i\Gamma_i$)) QED.

Theorem AEL2 shows that the set of theorems: (fol 'κ) of a set 'κ is a fixedpoint of Autoepistemic Logic if and only if the meaning (ms 'κ) of 'κ is a solution to the necessary equivalence. Furthermore, by FOL4 there are no other fixedpoints (such as a set not containing all its theorems) and by MA6 there are no other solutions (such as

a proposition not representable as a sentence in the First Order Logic object language). Therefore the Modal representation of Autoepistemic Logic (i.e. AEL), faithfully represents the original set theoretic description of Autoepistemic Logic (i.e. ael). Finally, we note that (ms 'κ) and ∀iΓ¡ may be generalized to be arbitrary propositions κ and Γ giving the more general modal representation: κ≡(AEL κ Γ).

Acknowledgements

This research was supported by National Science Foundation grants: #9818341 and #9972843.

Bibliography

[Anatoniou 1997] Antoniou, Grigoris 1997. *NonMonotonic Reasoning*, MIT Press.

[Bressan 1972] Bressan, Aldo 1972. *A General Interpreted Modal Calculus,* Yale University Press.

[Boyer&Moore 1981] R. S. Boyer and J. Strother Moore, "Metafunctions: proving them correct and using them efficiently as new proof procedures," *The Correctness Problem in Computer Science,* R. S. Boyer and J. Strother Moore, eds., Academic Press, New York, 1981.

[Brown 1986] Brown, Frank M. 1986. "Reasoning in a Hierarchy of Deontic Defaults", *Proceedings of the Canadian Artificial Intelligence Conference CSCI 86*, Montreal, Canada, Morgan-Kaufmann, Los Altos, 1986.

[Brown 1987] Brown, Frank M. 1987. "The Modal Logic Z", In *The Frame Problem in AI"; Proc. of the 1987 AAAI Workshop*, Morgan Kaufmann, Los Altos, CA .

[Brown 1989] Brown, Frank M. 1989. "The Modal Quantificational Logic Z Applied to the Frame Problem", advanced paper *First International Workshop on Human & Machine Cognition*, May 1989 Pensacola, Florida. Abreviated version published in *International Journal of Expert Systems Research and Applications, Special Issue: The Frame Problem. Part A.* eds. Keneth Ford and Pattrick Hayes, vol. 3 number 3, pp169-206 JAI Press 1990. Reprinted in *Reasoning Agents in a Dynamic World: The Frame problem*, editors: Kenneth M. Ford, Patrick J. Hayes, JAI Press 1991.

[Carnap 1946] Carnap, Rudolf 1946. "Modalities and Quantification" *Journal of Symbolic Logic*, vol. 11, number 2, 1946.

[Carnap 1956] Carnap, Rudolf 1956. *Meaning and Necessity: A Study in the Semantics of Modal Logic*, The University of Chicago Press.

[Fine 1970] Fine, K. 1970. "Propositional Quantifiers in Modal Logic" *Theoria* 36, p336--346.

[Hendry & Pokriefka 1985] Hendry, Herbert E. and Pokriefka, M. L. 1985. "Carnapian Extensions of S5", *Journal of Phil. Logic 14*.

[Hughes & Cresswell 1968] Hughes, G. E. and Creswell, M. J., 1968. *An Introduction to Modal Logic*, Methuen & Co. Ltd., London

[Konolige 1987] Konolige, Kurt 1987. "On the Relation Between Default Theories and Autoepistemic Logic", *IJCAI87 Proceedings of the Tenth International Joint Conference on Artificial Intelligence*, 1987.

[Konolige 1987b] Konolige, Kurt 1987. "On the Relation Between Default Logic and Autoepistemic Theories", Artificial Intelligence 35(3):343-382.

[Konolige 1989] Konolige, Kurt 1989. "On the Relation between Autoepistemic Logic and Circumscription Preliminary Report", IJCAI89.

[Leasure 1993] Leasure, David E., The Modal Logic Z Applied to Lifschitz's Benchmark problems for Formal Nonmonotonic Reasoning, University of Kansas Dissertation, University of Kansas Library , 1993.

[Leasure & Brown] Leasure,David E. Brown,Frank M.., "AutoEpistemic Logic with Quantifiers", Proceedings Florida Artificial Intelligence Conference, 1995

[Lewis 1936] Lewis, C. I. 1936. Strict Implication, *Journal of Symbolic Logic*, vol I.

[McDermott 1980] McDermott, D. and Doyle, J. 1980. "Nonmonotonic Logic I" *Artificial Intelligence*, 13.

[McDermott 1982] McDermott, D. 1982. "Nonmonotonic Logic II: Nonmonotonic Modal Theories", JACM, vol. 29, No. 1.

[Mendelson 1964] Mendelson, E. 1964. *Introduction to Mathematical Logic*, Van Norstrand, Reinhold Co., New York.

[Moore 1985] Moore, R. C. 1985. "Semantical Considerations on Nonmonotonic Logic" *Artificial Intelligence*, 25.

[Parks 1976] Parks, Z. 1976. "An Investigation into Quantified Modal Logic", Studia Logica 35, p109-125.

[Quine 1969] Quine, W.V.O., Set Theory and Its Logic, revised edition, Oxford University Press, London, 1969.

Author information

Frank M. Brown- Artificial Intelligence Laboratory, University of Kansas, Lawrence, Kansas, 66045, e-mail: brown@ku.edu.

THE HOUGH TRANSFORM AND UNCERTAINTY

V.S.Donchenko

***Abstract**: The paper deals with the generalisations of the Hough Transform making it the mean for analysing uncertainty. Some results related Hough Transform for Euclidean spaces are represented. These latter use the powerful means of the Generalised Inverse for description the Transform by itself as well as its Accumulator Function.*

***Keywords**: Uncertainty, Hough Transform, Accumulator Function, Generalised Inverse .*

Introduction

This report is the attempt to represent Hough Transform (HT) [Hough 1962] as a tool for analysis of the uncertainty

Some results, besides, are represented for the vector observations in the scheme of the Hough Transform as well as for complex observations in this case.

The Hough Transform (HT) for well over forty years has been and continues to be an important tool in analysis of shape and pattern recognition. But potentially the Transform seems to be much more than engineering tool only. The idea of the Transform may be used for analyzing uncertainty in much more general cases than in its classical variant. In the paper [Donchenko 1994] the general concept of the Hough Transform within the Hough-pair of the spaces was proposed. Later, in [Donchenko, Kirichenko,2001] [Donchenko , Kirichenko 2002] the powerful Generalized Inverse apparatus [Nashed, Votruba 1976]], [Albert 1977], [Kirichenko 1997], [Kirichenko, Lepeha 2001] applied to describe HT and Accumulator Function (AF). In the work the results from [[Donchenko, Kirichenko,2001] and [Donchenko 1999] are extended on the vector case.

General concept of the Hough Transform – Hough-pare of spaces

General concept of the HT[Donchenko 1994] as a tool for analysis of the uncertainty may be built on the base of so called Hough-pare of spaces, which are virtually a pare of sets S_o, S_p enhanced by its subsets G_θ L_s: $G_\theta \subseteq S_o$, $L_s \subseteq S_p$, $\theta \in S_p$, $s \in S_o$, mutually indexed. One set (space) S_o is interpreted as a space of observations, another S_p – as a space of parameters. This space of parameters is interpreted as a variety of the variants for uncertainty which corresponds to observation s.

These subsets are agreed in the next sense: for any pare (s,θ) $\theta \in L_s \Rightarrow s \in G_\theta$ and conversely : $s \in G_\theta \Rightarrow \theta \in L_s$. Some additional conditions may be added to these: about type of interception for example.

The HT within the Hough-pare is determined as a transition from the observation s – or its sequence s_1, $s_2,...,s_N \in S_o$ – to subset L_S , correspondingly –sequence of subsets $L_{s_1} = L_1$, $L_{s_2} = L_2,...,$ $L_{s_N} = L_N$ – of another space, indexed by the observations.

Such determination permits the description of straight and inversed HT, Hough-estimator, Fast HT as a sequential HT and Fast HT as the HT of the complex observations.

Each of the observations is supposed to be taken by the choice of the parameter $\theta_1,...,$ θ_N and the consecutive choice of the elements s_1, $s_2,...,s_N \in S_o$ from correspondent subsets of G_θ -type. Besides, the observations may be disturbed in that or this way. In this case one says that the elements are observed with an error.

The parameters $\theta_1,...,$ θ_N –some of them may be equal - is said to be represented in s_1, $s_2,...,s_N \in S_o$.

The set of parameters represented in the sample is supposed to be “comparatively small” and the main target of the HT-based analysis is ascertainment – estimation - of that set. Properly, this is the task of the estimation in the theory of the HT.

HT may be applied to the sequence $s_1, s_2,...,s_N \in S_o$ taken without previous choice of , $\theta_1,..., \theta_N$. In that case HT-analysis is targeted to describe "comparatively small" set of the parameters, "concentrated" the observations. This task may be called the task of the clustering in HT.

As to complex observations S, then we say it to be the subset $S = S_{i_1,...,i_k} = \{ s_{i_1}, s_{i_2},..., s_{i_k} \}$, $\{i_1, i_2,..., i_k\} \subseteq \{1,\ldots,N\}$ of the initial observations. And by HT of the complex observation $S= S_{i_1,...,i_k}$ we will call the set $L_S= L_{i_1,...,i_k}$, determined by the relation:

$$L_S= L_{i_1,...,i_k} = \bigcap_{j=1}^{k} L_{i_j} .$$

Sometimes HT of the complex observations is called Fast HT. Fast HT can cut essentially the set of the parameters, pretended to be represented in the sample.

This variant of the Fast HT one ought to differ from another using, when the Accumulator Function (AF) of the HT consecutively calculated for some set C - rough approximation - and its consecutive – detailed - partitions.

AF is determined by the HT of the sample – original or complex observations – as a function of set C in the space of parameter in one of the next two senses: absolute A(C) or relative NA(C) –by the relations:

$$A(C) = \Sigma_{i=1}^{N} \delta(C \cap L_i), , \ C \subseteq S_p , \quad (1)$$

$$NA(C)= A(C)/N = N^{-1} \Sigma_{i=1}^{N} \delta(C \cap L_i),. C \subseteq S_p , \quad (2)$$

where $\delta(C)$, $C \subseteq S_p$, equal 1, if C is not empty and 0, if C – empty set.

The summing in (1), (2) for the complex observations is by the set of the complex observations under consideration.

Argument C in the AF depend on the concrete types of spaces. For the Euclidean spaces for parameters and observations set C may be: ball as in (10), (11) below; hyper-cube; compact and so on.

AF is the mean to estimate the set of parameter, which are represented in the sample or the "smallest" set of the parameters in the clustering task. Properly, such set (or sets) is the set of maximum for AF.

Hough Transform in the Euclidean spaces

In its original variant HT was determined for the case, when

- S_o, S_p are appropriate rectangles in R^2;
- parametric sets G_θ, $\theta=(\rho,\varphi) \in R^2$ is the set of the graphics of the straight lines in the normal representation: $G_\theta= G_{(\rho,\varphi)} =\{(x,y) \in R^2: \rho = x \cdot \cos\varphi + y \cdot \sin\varphi\}$;
- parametric set $L_s=L_{(x,y)}$, $s=(x,y) \in R^2$ is the set of parameters for which correspondent lines G_θ include observation $s=(x,y)$: $L_s=L_{(x,y)}= \{(\rho,\varphi) \in R^2: \rho = x \cdot \cos\varphi + y \cdot \sin\varphi\}$.;

Observations s=(x,y) may be with an error so without it. In the first case $y = \bar{y} + \varepsilon_{(x,y)}$, where $\varepsilon_{(x,y)}$- the error of an observation. Errors, which correspond to different observations, are supposed to be independent but not obligatory identically distributed.

One of the generalization of that original variant may be such one, in which the spaces in the Hough-pare are any Euclidean spaces or their appropriate subsets:

- $S_o=R^m$, $S_p =R^n$
- G_θ, $\theta \in S_p$,is determined the graphic of mappings $y=g(x,\theta)$, $\theta \in R^l$ from R^n in R^m . The sample $s_1, s_2,..., s_N$ consists of the pares $s_i=(x_i, y_i)$:

$$y_i = g(x_i,\theta_i) \in R^m, x_i \in R^n , \quad (3)$$

$$y_i = g(x_i,\theta_i) + \varepsilon_i \in R^m, x_i \in R^n , \ i=1,\ldots,N \quad (4)$$

Variant (4) represents the scheme of observations with an error, (3) – without it.

HT for such sample is the sequence $L_1, L_2,..., L_N$ of the subsets from R^l, where

$$L_i=\{ \theta \in R^l: y_i = g(x_i,\theta)\}, \ l=1,\ldots,N.$$

Particularly, if the set of mapping is of affine-type (linear + shift) from R^n in R^m, then the matrix $\theta = A \in R^{m\times(n+1)}$ of this map may be considered as the parameter, i.e. $l = m\times(n+1)$.

HT, AF and Hough-estimator are described on the sample (x_i,y_i), i=1,N, $x\in R^n$, $y\in R^m$ points of the graphics of the affine set of the mappings $y = A \begin{pmatrix} x \\ 1 \end{pmatrix}$, - $x\in R^n$, $y\in R^m$, A –$m\times(n+1)$) matrix, $\begin{pmatrix} x \\ 1 \end{pmatrix}$ – block vector-column from $x\in R^n$ and 1.

These observations may a be observed in the scheme without the error (3) or with it (4), correspondingly:

$$y_i = A_i \begin{pmatrix} x_i \\ 1 \end{pmatrix}, \tag{5}$$

$$y_i = A_i \begin{pmatrix} x_i \\ 1 \end{pmatrix} + \varepsilon_{x_i} \ , \ x_i \in R^n \ , \ y_i \in R^m \ , \ A_i \in R^{m\times(n+1)}, \ i = 1,\ldots,N. \tag{6}$$

As it was remarked earlier specific parameter $A_i \in R^{m\times(n+1)}$, i =1,…,N corresponds to each of the observations. Only scheme with the error (6) and independent errors will be considered below. The last means, that errors of the observations ε_{x_i} , i =1,…,N are independent. The distribution of ε_x will be denoted by P_x:

$$P_x(B^{(m)}) = P\{\varepsilon_x \in B^{(m)}\}, \tag{7}$$

$B^{(m)}$ – Borel set from R^m.

HT $L_{(x,y)}$ of an observation (x,y) is the set of affine transforms, mapping x in the observed y, which may be disturbed:

$$L_{(x,y)} = \{A \in R^{m\times(n+1)}: y = A \begin{pmatrix} x \\ 1 \end{pmatrix} \}. \tag{8}$$

$S_r(\theta)$ below denotes the r-ball with a center in the θ in the space of all $m\times(n+1)$ matrixes with the trace norm, induced by the trace scalar product:

$(A,B) = \mathrm{tr}\, A'B = \Sigma_i (A'B)_{ii} = \Sigma_{ij} a_{ij}b_{ij}$.

The trace norm, obviously, coincides with the Euclidean norm in $R^{m\times(n+1)}$.

AF in absolute or frequency variants will be defined for the balls $S_r(\theta)$ as for arguments and denoted correspondingly $A_r(B)$, $NA_r(B)$:

$$A_r(B) = A(S_r(B)) = \Sigma_{i=1}^N \delta(S_r(B)\cap L_i), \ , \tag{9}$$

$$NA_r(B) = A_r(B)/N = N^{-1} \Sigma_{i=1}^N \delta(S_r(B)\cap L_i), \ B \in R^{m\times(n+1)}. \tag{10}$$

Theorem 1. AF for the sample (x_i,y_i), i=1,N points of affine observations may be represented by next expression:

$$A_r(B) = \Sigma_{i=1}^N \delta(S_r(B)\cap L_i) = \sum_{i=1}^N \delta(\varepsilon_{x_i} \in S_{r\sqrt{1+||x_i||^2}}((B - A_i)\begin{pmatrix} x_i \\ 1 \end{pmatrix})). \tag{11}$$

Proof. Accordingly with the theorem 2 [2]

$$A_r(B) = \Sigma_{i=1}^N \delta(S_r(B)\cap L_i) = \Sigma_{i=1}^N \delta(||y_i - B\begin{pmatrix} x_i \\ 1 \end{pmatrix}||^2 \le r^2(1 + ||x_i||^2) \ , \ B \in R^{m.\times(n+1)}. \tag{12}$$

As for the each of observations

$$y_i = A_i \begin{pmatrix} x_i \\ 1 \end{pmatrix} + \varepsilon_{x_i} \ , \ i = 1,\ldots,N,$$

then condition

$$||y_i - B\begin{pmatrix} x_i \\ 1 \end{pmatrix}||^2 \le r^2(1 + ||x_i||^2)$$

in (12) is equivalent to the condition

$$\| A_i \begin{pmatrix} x_i \\ 1 \end{pmatrix} + \varepsilon_{x_i} - B \begin{pmatrix} x_i \\ 1 \end{pmatrix} \| \le r\sqrt{1+\| x_i \|^2} ,$$

that proves the theorem.

Remark 1. $S_r(B)$ in (12) is the r-ball in the trace norm in the matrix space with the center in a B, then the $S_{r\sqrt{1+\|x_i\|^2}}((B-A_i)\begin{pmatrix} x_i \\ 1 \end{pmatrix})$ is the $r\sqrt{1+\| x_i \|^2}$ -ball in R^m with the center in $(B-A_i)\begin{pmatrix} x_i \\ 1 \end{pmatrix}$, i=1,...,N.

Corollary 1. Obviously, $\delta(S_r(B) \cap L_i)$, i =1,...,N are Bernoulli-distributed random variables with the parameters, determined by the expressions

$$p_i = P\{\varepsilon_{x_i} \in S_{r\sqrt{1+\|x\|^2}}((B-A_i)\begin{pmatrix} x_i \\ 1 \end{pmatrix}\} , \text{ i=1,...,N.} \qquad (13)$$

Proof. The result is the consequence of taking 1 for each of $\delta(S_r(B) \cap L_i)$, i =1,...,N in (12).

$$\delta(\varepsilon_{x_i} \in S_{r\sqrt{1+\|x\|^2}}((B-A_0)\begin{pmatrix} x_i \\ 1 \end{pmatrix}) , \text{ i=1,...,N.}$$

Theorem 2.(0-1 Law). The limit value of the AF when with probability 1 is finite or infinite as $n \to \infty$. It is finite iff $\sum_{n=1}^{\infty} p_i = \sum_{n=1}^{\infty} P\{\varepsilon_{x_i} \in S_{r\sqrt{1+\|x\|^2}}((B-A_i)\begin{pmatrix} x_i \\ 1 \end{pmatrix}\} < \infty$

Proof. The proof repeats that one for scalar case in [8].

Theorem 3. The next limit take place with the probability 1:
$\lim_{N\to\infty}(N^{-1}\sum_{i=1}^{N}\delta(S_r(B)\cap L_i)) - N^{-1}\sum_{i=1}^{N} p_i) = \lim_{N\to\infty}(NA_r(B)) - N^{-1}\sum_{i=1}^{N} p_i) = 0$, where p_i, i=1,...,N are determined by (13).

Proof. As in was in previous case the proof repeats that one for scalar case in [8].

Remark 2. For the case under consideration – vector case – all the consequences from [8] for scalar observations are valid.

Remark 3. The statements of the theorems earlier are free from constraints on the distribution of an error. Besides, the distribution may depends on x.

Theorem 4. AF for the sequence of K complex observations may be represented by next expression:

$A_r(B) = \Sigma_{i=1}^{K} \delta(S_r(B) \cap Ł_i) = \Sigma_{i=1}^{K} \delta(\|(Y_i - BX_i)X_i^+\| \le r)$, $B \in R^{m\times(n+1)}$,

where:

- $Ł_i$, i=1,...,K – Hough transforms for the complex observations,
- A^+- General Inverse for A,
- Y_i, i=1,...,K – block-matrixes from the y-components of the original observations the complex observation consists of,
- X_i, i=1,...,K– block-matrixes from the x-components of the original observations the complex observation consists of,

Conclusion

The subject matter of the paper are generalizing the Hough Transform that convert it in mathematical tool with wide range of application in analyzing the “uncertainty”. The abstract form of the HT is represented within the framework of Hough-pare of spaces.

The HT for observations and parameters from Euclidean spaces has been represented and investigated for affine sets of transforms. The author would like to believe that the results represented are the only one step to promote HT to be the mean for uncertainty analyzing.

Bibliography

[Albert 1977]Albert A. Regression, pseudo inverse, recurrent estimation..–M.: Nauka, 1977 r., 305 p(in Russian).

[Donchenko 1994] Donchenko V.S. General Scheme of the Hough Transform and properties of the Hough estimates in special case of Discrete Spaces.//Statist. Res. Rep.–Umea, Sweden: University of Umea,1994.–1994-6, S-901 87. 10 p.

[Donchenko 1999] Donchenko V.S. Limit theorems for Accumulator Function in Hough Transform Scheme. // Bulletin of the University of Kiev. Series: Physics \& Mathematics. –1999, Vol..№ 1, cc.191-195.(In Ukrainian)

Donchenko, Kirichenko,2001] Donchenko V.S., Kirichenko M.F. Hough Transform and Generalized Inverse. // Bulletin of the University of Kiev, Series: Physics \& Mathematics.– 2001, vol. № 4, , pp.191-196. (in Ukrainian)

[Donchenko , Kirichenko 2002] Donchenko V.S., Kirichenko M.F. Fast Hough Transform and Generalized Inverse.// Journal of Automation and Information Sciences . – 2002, vol. №2, pp..115-125. (In Russian, also translated in English)

[Hough 1962] Hough P.V.C. Method and Means for Recognizing Complex Patterns. - U.S. Patent 3069354, 1962.

[Kirichenko 1997] Kirichenko M.F. Analytical representation for Generalized Inverse disturbances // Cybernetics and System Analysis. – 1997, №2, pp.98-107. (In Russian)

[Kirichenko, Lepeha 2001]. Kirichenko M.F., Lepeha M.P. Disturbances for pseudo-inverse and projective matrixes and their applications to linear and non-linear identification.// Journal of Automation and Information Sciences. - 2001, №1, c.6-23. (In Russian, also translated in English)

[Nashed, Votruba 1976] Nashed M. Zuhair ,Votruba G.F. A Unified Operator Theory of Generalized Inverse, Proceedings of an Advanced Seminar Sponsored by the Mathematical Research Center, The University of Wisconsin, Madison, October 8-10, 1973. – New York, Academic Press, 1976.

Author information

Volodymyr S. Donchenko – Kyiv National Taras Shevchenko Universiry, Professor.
2, Akademician Glushkov prospectus, building 6, Kyiv,03680, Ukraine; e-mail: vsdon@unicyb.kiev.ua

FRONTAL SOLUTIONS: AN INFORMATION TECHNOLOGY TRANSFER TO ABSTRACT MATHEMATICS

V. Jotsov

***Abstract**: The paper introduces a method for dependencies discovery during human-machine interaction. It is based on an analysis of numerical data sets in knowledge-poor environments. The driven procedures are independent and they interact on a competitive principle. The research focuses on seven of them. The application is in Number Theory.*

***Keywords**: knowledge discovery and data mining, modeling, Number Theory.*

1. Introduction

The offered research has begun since 1986 after the exploration of some of the early D. Lenat's papers [Lenat 1976, Lenat 1983]. They gave us the conviction, that the information technologies (IT) are suitable for applications in models which are bounded by Number Theory. The newest evolutionary programming (EP) [EAEA 1997, EA 1997, Nordin 1999] research confirms the possibilities for elaborating new formulas. The considered paper follows the line from our papers [Jotsov1 1999, Jotsov2 1999]. Compared with the works of Lenat [Lenat

1983], or with other sources in the references on informatics, the majority of our papers describe the mathematical results, not the method. The paper's scope is *interdisciplinary* and includes many significantly far research areas. To some extent the proposed method is a continuation of the Lenat's ideas and serves the same **purposes**: elicitation of new knowledge in the integer data processing, derivation of new formulas, and *whenever possible* generation of new mathematical theorems. At the same time it has some points in common with the Narin'yani's, Shvetsov's constraint programming [Narin'yani 2000,Shvetsov 1997] and reasoning in the Altshuller or Hadamard or Polya style [Altshuller 1979, Hadamard 1975, Polya 1963]. The approach is enriched from the most remote principles coming from both directions but it *uses no plausible reasoning*.

2. The FRONTAL Method and the Working Environment

The shortly described below FRONTAL method interacts with several other methods under the common control of a new type of an evolutionary metamethod. The metamethod avoids or *defeats* crossovers, phenotypes, mutations, etc. Below we choose the description in an analogous manner as the way to reduce the extra descriptions, because the general scheme of the chosen strategy is rather voluminous. The evolutionary metamethod swallows and controls the following methods:

I. FRONTAL method;

II. KALEIDOSCOPE method;

III. FUNNEL method;

IV. CROSSWORD method.

The KALEIDOSCOPE method is the *background* for the human-machine strategies for work. The machine forms and visualizes different mappings for the chosen groups of numbers or like, while the obtained results are estimated by the human. *The human* makes the necessary conclusions and undertakes the required steps. Analogically the kaleidoscope rotations form different images in a hazardous manner, and the spectator takes an *informal decision* whether the seen by him is nice, original etc.

Let's assume you have a *plastic funnel*. If you fix it vertically above the ground, you can direct a stream of water or of vaporous drops etc. If you change the funnel direction, then the stream targeting will be hampered. Fixing the funnel horizontally makes it practically useless. Analogically in the evolutionary method the general direction in numerical models is determined likewise. In other words this is a movement along the predefined gradient of the information. This term is proposed in a manner which *has some connection* to [Baldi 1995]. Just like in the case of the physical example in the beginning of the investigation there are lots of undirected hazardous steps towards conclusions and hypotheses. The FUNNEL method is based on inconsistency tests with known information.

Let us assume that the reader solves a problem with a complex sentence of 400 letters with vague for the reader explanations. Let the unknown sentence be horizontally located. The reader can't solve the problem in an arbitrary manner, because the number of combinations is increased exponentially. Now it is convenient to **facilitate** the solution by linking the well known to the reader information with the complex one from the same model. The reader tries to find vertical words that he is conscious about like the place of our conference KDS 2003 - Varna. The more the crosspoints are, the easier is the solution of the horizontal sentence. The approach for the CROSSWORD is *even easier*. Here both the easy meanings and the difficult ones are from one domain, therefore there exists an additional help to find the final solution.

For pity the paper length does not allow us to make more detailed descriptions of the mentioned above methods, and/or their connections, interactions, etc. We will turn exclusively to the considered FRONTAL method.

The trend in the investigation includes solutions of complex hypotheses and problems which require the usage of integer-number models. Great number of these problems have been unsolved for centuries; their decisions cannot be obtained *prima vista* or in a **frontal** manner. This is the reason for the development and application in mathematics of an evolutionary strategy. In it the **preproofs** are on the first place. In the process of solving oversophisticated problems the first draft solutions comprise only the first step in the marked by the FUNNEL direction. This direction is an approximate. This is due to the initial conditions and knowledge constraints. Fig. 1 depicts a similar general direction for research by the A-B line. The obtained intermediate solutions follow another route, A-C-D-B. The solution B is inaccessible from the node C or from any other node before D. The user can

change the direction according to her/his wish. D-E on Fig. 1 is a deviation from the line A-B. The new branch marks the process of solving another problem. Any of the intermediate solutions may contradict or doesn't correspond to the final solution (B). Together they form the set of preproofs for B. The mathematical proofs are formed in the process of evolution with no probabilities. In the evolutionary metamethod the preproofs are usually weak, with bottlenecks and/or incomplete. The preproofs in the considered domain are never so good as to be included in the "official" proof. Nevertheless they must not be easily rejected. They are weaker, but in our case they *are not* heuristical by nature, and they might assist the solution of other problems as well.

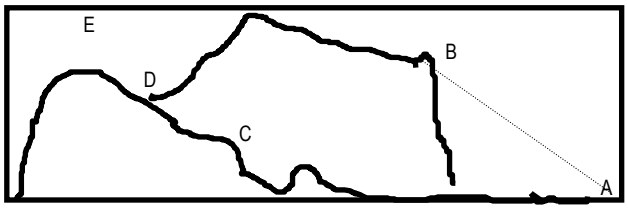


Fig. 1.

The presented evolutionary meta-method has the following features. The solution is evolved *step by step*. At every step it is possible to have a progress or a regress compared with the previous decision. The role of probabilities and other subjective estimations is played by interactive approaches for knowledge acquisition, data linkage, mappings and other processing of data and knowledge. The investigated FRONTAL method (I) includes the following procedures. Their short abbreviations are given in bold letters.

1. **MOC**: Mix Or Change (data/knowledge);
2. **BIND**: Connects the information (data sets/knowledge) during the automatic work or shows it to the user;
3. **WHY & HOW**: Forwards it (data sets/knowledge) to the user;
4. **CS**: Constraint Satisfaction (of knowledge), based on the weak negation ~;
5. **SPREAD** (knowledge);
6. **WHAT**: Explanation (of data/knowledge);
7. **EF**: Elimination Filter.

All the seven procedures can be modified together with the change of the different models. Now we introduce in short the FRONTAL method terminology. Let M be a set of such models M_i which contain sets of arithmetic progressions $\{a_i+b_ik\}^{\infty}_{k=0}$. At that:

(1) $$b_i = \prod_{p_j \in M} p_j; \quad p_j \in P.$$

where P denotes the prime numbers set. Every progression from M_i may be treated as a result after sieving out the set of positive integers, consisting of all p^i_M and such composite numbers that at least one of p^i_M divides them. To simplify the contents other models are not included, e.g. based on geometrical progressions. It is accepted that $(a_i,b_i)=1$; $a_i<b_i$.

Four operations are introduced in every model: {+,-,*,/}. Possibly every application of the algorithms based on the FRONTAL method leads to some change of different parameters inside the built-up algorithms whenever the model changes. This model changes serve as *an algorithm stability test*. This is the right place to use MOC. Denote $V=\{v_1,v_2,...v_z\}$ is a set of parameters. During our first investigations in the eighties we used V in a way similar to the genotype from Genetic Algorithms (GA). The user had the option to accept such v_i which deserved his attention and the system proceeded with the goal task. We offered that every task must begin with $V=\{\varnothing\}$. Thus the *released assumption* brings the user closer to data mining tasks.

The author proposes the following generalized MOC algorithm with an automatic mode set-up: **A**. Fixing of v_i in the current model; **B**. Case-based inclusion of v_i from previous solutions; **C**. The algorithm proceeds with review of $v_i=0$; **D**. An inverse mapping of (C.) is introduced or $v_i \rightarrow \max$; **E**. v_i is replaced by another parameter in V; **F**. The algorithm goes on with the WHAT procedure or with other procedures from the FORWARD method. The general MOC scheme is postulated with the formulas (2) and (3).

(2) $S(V \rightarrow V')$; $card(V) \neq card(V')$.

(3) $L(S(v_{i,k}))\rightarrow L(S(v_{i,j}))$; $S(v_{i,k})\neq S(v_{i,j})$.

Here S is a situation which has arisen as a result from the MOC activity changing the set V or its separate element v_i. L is the modal operator *possibility*.

For example, let v_5=2 means that all the numerical data are copied in a bidimensional array. This automatically inputs $v_6 = \vec{x}$ and $v_7= \vec{y}$ in V. During the activation of (**D**.) $v_8= \vec{z}$ is introduced, etc. When processing (**C**.), the bounded with v_5 parameters v_6=0 or v_7=0 are affected. In this way MOC acquires new knowledge from the data investigation. The next example is not so theoretical. Rather it is connected with numbers from eight arithmetic progressions.

The following denotations are introduced. $\{m+nk\}^{\infty}_{k=0}$ is an arithmetic progression (progression for short). In it m is the first member, and n is the step. $\boldsymbol{\pi(x)}$ is the total number of the primes which are elements of the set **P** ($p_i\in P$, $p_i\leq x$). $\boldsymbol{\pi_{n,m}(x)}$ is the number of primes $\leq x$ which are contained in the progression. $\mathbf{S_5}$ is an union of 8 progressions $\{y+nk\}^{\infty}_{k=0}$, $y\in Y$, Y={1,7,11,13,17,19,23,29}. Every of these progressions is represented as a column in Fig. 2 if the elements of S_5 are shown vertically. Fig. 3 shows the same environment in a slightly different manner. Every of the elements in S_5 is computed in the following way. The first number from the corresponding column - see line 1 - is added to the number from the same line and the leftmost column. For example $s_{14,2}$=7+390 is in line 14 and column 2. Composite numbers in S_5 are represented as products of prime numbers. The primes are the result of the decomposition of the composites. In Fig. 3 the primes are *omitted* while the particular cases $y\in Y$ are given in brackets. MOC has no logical inference. It simply finds and changes the scope parameters one by one while the rest of the parameters remain unchanged. The lines below show the cases when MOC pastes or cuts some of the elements in the interpretation. For example during the investigation of the operations addition and multiplication in S_5 the following parameters attract the attention: primes (with just a single divisor), composites with at least 2 divisors, 8 columns which are parallel to the vertical axis $\vec{y}$ and 15 lines which are parallel to $\vec{x}$. These 4 parameters can have other designations, which will have similar meanings. The names are not significant. The parameters are established by mere observations e.g. directly on the figures. The following transforms for the transition from Fig. 2 to Fig. 3 are used:

1	7	11	13	17	19	23	29
31	37	41	43	47	49	53	59
61	67	71	73	77	79	83	89
91	97	101	103	107	109	113	119
121	127	131	133	137	139	143	149
151	157	161	163	167	169	173	179
181	187	191	193	197	199	203	209
211	217	219	223	227	229	233	239
241	247	251	253	257	259	263	269
271	277	281	283	287	289	293	299
301	307	311	313	317	319	323	329
331	337	341	343	347	349	353	359
361	367	371	373	377	379	383	389
391	397	401	403	407	409	413	419
421	427	431	433	437	439	443	449

Fig. 2.

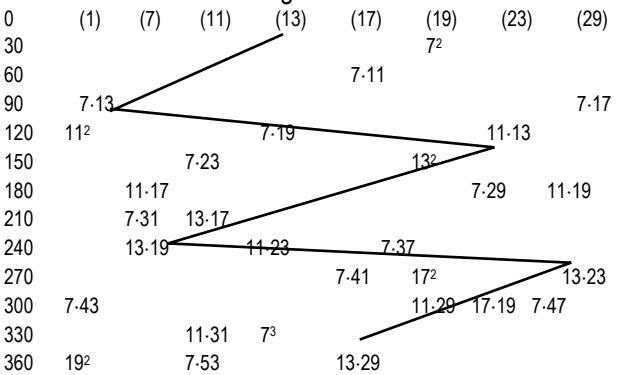


Fig. 3.

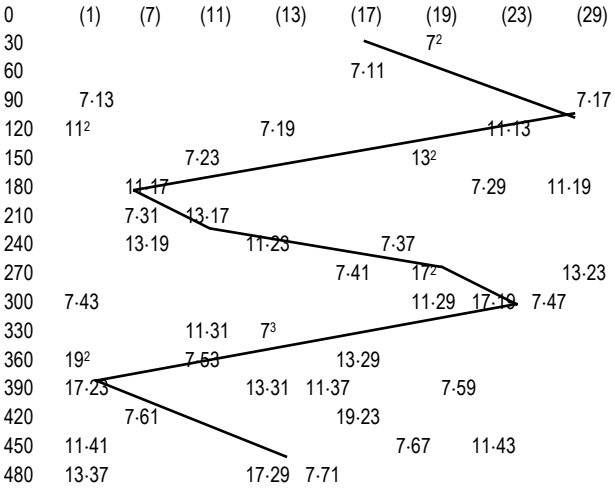


Fig. 4.

(T_1). The primes are determined but not shown from all the numbers in the fragment, see Fig. 2. The *very omission* introduces some new information. The figures below demonstrate the following versions of transformations in S_5.

(T_2). All the composites are presented as products of prime divisors.

(T_3). All the composites with the divisor of 13 are successively connected with straight lines.

(T_4). All the composites with the divisor of 17 are successively connected with straight lines. The result is shown in Fig. 4. The transformation itself is in the divisor replacement.

(T_5). Besides the graphical interpretations in Fig 3 and Fig. 4 must be added similar pictures for the "neighbors below" 43 and 47 or 13+30, 17+30. The result has the same succession of beat for the columns with periods 30 times 43 and 30 times 47. The illustrations resemble the Fig. 3 and Fig. 4 but they are more elongated due to the greater period.

(T_6). The parameter influence of $\overrightarrow{x}$ is **"**reduced"**.** So the attention is concentrated upon the **beat succession** for the columns S_5 and the lines are "compressed". The results are depicted in Fig. 5 and Fig. 6.

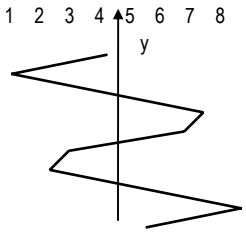


Fig. 5.

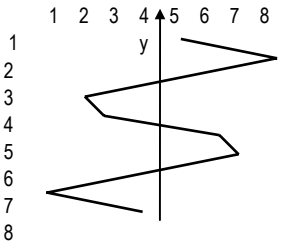


Fig. 6.

The discussed six relatively simple transformations show plainly and unambiguously that the cited in Fig. 5 way to beat the columns is one and the same for all the elements in column 4 in S_5: 13, 43... The result is in relation with the transition from a piece of S_5 to the whole S_5 or v.v. It is specially discussed in the SPREAD presentation. Fig. 6 presents the situation with the elements in column 5 (17,47...) which is analogous.

The two numerical sets in Fig. 5 and in Fig. 6 interpret the same cycles as those in Fig. 3 and Fig. 4. These cycles have common "similarity centers" on $\overrightarrow{y}$. Moreover the two figures coincide if one of them is rotated 180 degrees around $\overrightarrow{y}$(T_7). The revealed dependency is valid only for numbers of the type n and 30k-n for every positive integer k. If the beat cycle for the columns in Fig. 5 is in a column starting with the element m, then the analogical cycle in Fig. 6 is in a column starting with 30-m. The constantly repeated number 30 leads to (T_8): 30=2·3·5. *The act of mathematical creation* for Fig. 2-Fig. 3 is unambiguously simple when mapping Fig. 5 to Fig.

6. The revealing of different numerical properties takes place in the described above MOC procedure. Other transformations can be pointed like (T_9): the discovery of numbers which *can't be divisors* of any integer number. Zero which is not an element in S_5, but being a similarity center for the positive and negative parts in S_5, is set in this manner. The interpretation of any prime cycle as on Fig. 7 is unified by the total discrimination of the influence of $\overrightarrow{y}$; (T_{10}) is a suitable example as an illustration vs. (T_7).

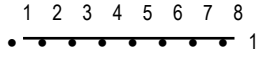


Fig. 7.

All discussed transformations are just consequences of observations based on the model. They give no answers to questions like WHY or HOW the presented results are obtained. The body of the preproofs is formed on the basis of such conclusions.

The achieved with MOC results may be related and compared. This is the purpose of the BIND procedure. The extracted information is analyzed by BIND on the basis of juxtapositions. BIND is based on the above function mapping $S_{x1}(v_1,...v_z)=S_{x2}(v_1,...v_z)$ or $S_{x1}(v_1,...v_z)\neq S_{x2}(v_1,...v_z)$ where x_i are different objects or data groups. The detailed BIND overview exceeds consideration line in the paper. The obtained results most of all lack of proving power and the inference obtained is nonmonotonous. Therefore after determining the regularities it is possible to formulate prompting queries to the user which are decorated in the well known form **WHY** and **HOW.** The system forms the basis for the general solution, and the details are an object for a manual or an interactive work. In this way, the investigation evolves itself. Using the WHY&HOW procedure, a new set is built from mutually related formulas and knowledge from the same domain.

The CS procedure is formalized in a manner similar to the one in [Narin'yani 2000]. An outstanding feature of the presented variant of CS is that the bounds of the domain are not restrictive in the case of a weak negation ~. After the contradictory resolution these bounds are overcome. The contradiction concentrates the attention to the incompleteness in the scope. The goal-forming scenario in the constraint satisfaction paradigm is formulated as follows. Let the variables x_1, x_2, ... x_n be the mapped sets of their value spaces X_1, X_2, ... X_n. The constraints $C_j(x_1,x_2,...,x_n)$, j=1,...k are valid for the same X_j. It is necessary to find such sets $<a_1, a_2, ... a_n>$ such that $a_i \in X_i$ and they satisfy all C_j simultaneously.

Denote M* is a *subdefinite model* or - roughly speaking - an incomplete model. Let $C_i(x_1,x_2,...,x_n)$ is one from the investigated constraints, and $N(x_1,x_2,...,x_n)$ be such that:

(4) $$N(x_1,x_2,...,x_n) \rightarrow \sim C_i(x_1,x_2,...,x_n).$$

This means that the constraint is violated because (4) contains the weak nonclassical negation ~. The ~ based inconsistencies may be solved after the complementation of M* with new knowledge/data. The augmented model is denoted with M'. In it the examined constraint takes the form $C'_i(x_1,x_2,...,x_n)$, z=n or z≠n, where:

(5) $$C'_i(x_1,x_2,...,x_n) \rightarrow \sim C_i(x_1,x_2,...,x_n).$$

There are other possible ways for the transition M*-M' besides $N(x_1,x_2,...,x_n)$. One of them is to include a new parameter v' in M*. Another approach is possible in the case when the system of constraints has no solution. Often in such cases there exists an information which admits the re-examination of $C_i(x_1,x_2,...,x_n)$. For example, let us examine the numbers x≥11. Then we may come to the conclusion that:

(6) $$\pi(x) > \frac{x}{\ln(x)}.$$

Here M* has no constraints and $C_i=\{\varnothing\}$. The result can be monotonously generalized to the whole interval $[0,\infty]$. The case when x=8 violates the formula (6). This contradicts the assumptions especially the case $C_i=\{\varnothing\}$. The introduction of C'_1: x≥11 leads to the result:

(7) $$\pi(x) \underset{x\geq 11}{>} \frac{x}{\ln(x)}.$$

The last three procedures do not contain substantially new theoretical ideas. SPREAD is based on the well known concept of mathematical induction. WHAT is designed to communicate with humans, because the internal

representation of the solutions is obscure. EP serves as a barrier against knowledge duplications or a surplus knowledge.

The interaction between the first five procedures is on a competitive basis according to the JUNGLE principle. In some cases they act in the role of *demons*. In the rest of the cases the top priority is assigned to the procedure from the previous iteration or this one which has generated the most effective solutions. The following formalization is aimed to derive this simple estimates and agreements. JUNGLE is based upon estimates $0 \le f(Q_i) \le 1$ for every procedure of the FRONTAL-based set $Q=\{Q_1,...,Q_7\}$. In this case it is preferable to compare the described JUNGLE strategy with the one from GA "the fittest wins" ([EAEA 1997], p.3). We use it in the form "the winner is best estimated". If $f(Q_i)=1$, then the procedure interacts with EF and the user. If $0.25 \le f(Q_i)<1$, then the display contains this value, and the corresponding solutions are considered only on the user request. The user may interfere in the automatic process of the estimation. The threshold value $f(Q_i)=1$ is achieved in the following situations:

(8) $S(Q_j) \rightarrow G(Q_i)$; $j \neq i$; $i,j=1...7$; $G(Q_i) \rightarrow f(Q_i)=1$.

where G is the modal operator *necessity*, $S(Q_j)$ is a scenario in Q_j leading to $G(Q_i)$. An example of (8) is presented above after ($\mathbf{T_6}$) thus activating SPREAD by MOC.

(9) $U \rightarrow f(Q_i)=1$.

Here U means *user-defined activation*. The user defines the necessary parameters for Q_i.

(10) $S(Q_3)=c \rightarrow f(Q_i)=1$.

where $S(Q_3)$ is the BIND output. The meaning of **c** (for short from **convergence**) is that *the results from the two independent research lines coincide*. Fig. 7 depicts an example leading to $S(Q_3)=c$. In the future JUNGLE may incorporate Machine Learning (ML) approaches. At that:

(11) $S(Q_3)=a \rightarrow G(Q_i)$.

(12) $S(Q_3)=e \rightarrow G(Q_i)$.

where **a** means "the memorized logical inference is abbreviated"; **e** means *an explanation of the obtained earlier results*. $f(Q_i)<1$ is obtained in the following cases:

(13) $f^p(Q_i)=\max_j(f^p_j(Q_i)),j=1,...7 \rightarrow f(Q_i)=0.5\, f^p(Q_i)$.

where $f^p(Q_i)$ are all the memorized evaluations in MOC.

(14) $f^p(Q_i)=\max_t(f^p_j(Q_i,t)),j=1,...7 \rightarrow f(Q_i)=0.7\, f^p(Q_i)$.

Only the last remembered value for the corresponding $f(Q_i)$ is taken into account in (14). Some of the above presented procedures are included not only in the FRONTAL, but also in the neighboring methods. The set of all those methods uses the same JUNGLE principle.

The goal function *is easy to change* (see Fig. 1), so the procedures from 1 up to 7 may operate not only with data, but also with goals. E.g. BIND can operate with hypothesis I with hypothesis J in S_5, etc.

3. Experimental Studies and Some of Theoretical Results

The software for the research includes more than 20 programs written in Visual Basic and more than 200 MB Excel data. The assistant and defensive software consists of more than 20 programs in C and C++.

The introduced method generated new results even during the first investigations in 1986. The following strategy was formulated later. The target is to find dependencies in the arrangements of different sets of numbers, e.g. which are multiples of 17. (For example see Fig. 4 and the multiplication cycle 17). One can say that the start is with *zero information*. We introduce descriptions of well known hypotheses, e.g. the twin primes hypothesis, Goldbach's conjecture etc. in the same model. Finally we obtain new mathematical dependencies and formulas. In practice this approach starts with a research of the twin primes hypothesis with a difference of 2: these are couples of prime numbers 5 and 7, 11 and 13 etc. The hypothesis is based on the suggestion that there exist an infinite number of such similar pairs. The hypothesis formalization *must not be mistaken* with the goal function. It is simply a model inside the given sets of progressions. The research of the multiplication operations with prime numbers in different numerical models, e.g. in S_5 leads to the conclusion that the principle properties of different composite numerical unions are also prime number functions (15), (16)! This result at a first glance is very remote

from the twin primes hypothesis. This result relates to the proof of Theorem 1 which was not a target in the research. Nevertheless it may assist in the process of solving for many different **goals.** The famous Dirichlet's theorem is a corollary from this Theorem 1.

$$c_{K,6,1}(x) = \sum_{p=7}^{p_{z7}} c_{K-1,6,1}(\frac{x}{p}) + \sum_{p=5}^{p_{z5}} c_{K-1,6,5}(\frac{x}{p}). \quad (15)$$

$$c_{K,6,5}(x) = \sum_{p=5}^{p_{z5}} c_{K-1,6,1}(\frac{x}{p}) + \sum_{p=7}^{p_{z7}} c_{K-1,6,5}(\frac{x}{p}). \quad (16)$$

where $p_{za} \in \{a+6k\}^{\infty}_{k=0}$, $C_{k,6,a}(x)$ are all the composites $\leq x$ from $\{a+6k\}^{\infty}_{k=0}$ which contain k prime divisors.

Theorem 1.

We have the interval [0,x]. In it we have two progressions $\{m_1+nk\}^{\infty}_{k=0}$ and $\{m_2+nk\}^{\infty}_{k=0}$ and the relevant numbers are mutually prime: $(m_1,n)=1$, $(m_2,n)=1$. Denote $\Delta\pi_{n,mi}(x)$, i=1,2. The denotation introduces the difference (delta) in the number of the primes $\leq x$ included in both progressions. This difference may not be greater then the number of the primes in the range $[0, \sqrt{x}]$, which is signed as follows: $\Delta\pi_{n,mi}(x) \leq \pi(\sqrt{x})$.

The Theorem 1 proof is given in [Jotsov2 1999]. Theorem 1 is the basic tool for the derivation of the twin primes formula:

$$P_x(p, p+2) \sim 1.320323632 \frac{(\pi(x))^2}{x}. \quad (17)$$

where $P_x(p,p+2)$ is the number of twin prime couples $\leq x$, ~ means "asymptotically equal". The solutions below are related to the well known Hardy-Littlewood's hypothesis, the formalization of which is introduced in (19). The formalization check of it revealed a series of inconsistencies, so the hypothesis was transformed in (18). Finally the FRONTAL method has lead to a new hypothesis 1 which is stronger than the Hardy-Littlewood's.

$$P_x(p, p+d_1, \ldots p+d_{z-1}) \geq K_z \frac{(\pi(x))^z}{x^{z-1}}. \quad (18)$$

where P_x is the number of z-tuples $\leq x$. They have different admittable differences between, and K_z are the corresponding coefficients [Riesel 1985].

$$P_x(p, p+d_1, \ldots p+d_{z-1}) \sim K_z \frac{x}{(\ln x)^z}. \quad (19)$$

Hypothesis 1.

Denote **z** the arithmetic progressions $\{a_1+b_1k\}^{\infty}_{k=0}$ ... $\{a_z+b_zk\}^{\infty}_{k=0}$ with a *noncoinciding* step of progressions. Let all the corresponding (a,b)=1. If **z**-tuples of positive integers $(c_i,d_i,...z_i)$ are compared; all of them are positive integer numbers; $c_i=a_2-a_1+(b_2-b_1)(i-1+w_1)$ ... $z_i-c_i=a_z-a_1+(b_z-b_1)(i-1+w_z)$, and **w** are positive integers, then there exist infinitely many such **z**-tuples $(c_i, d_i, \ldots z_i)$ in which all the numbers are primes $c_i \in P$, $d_i \in P$, ... $z_i \in P$.

Hypothesis 1 is formulated as a result of the application of Theorem 1 to the formula (18). Finally the MOC procedure was applied to the model of the Hardy-Littlewood's hypothesis in S_5. At the end we shall reveal an indicative fact. The paper containing the draft with the Theorem 1 proof is one page long. The initial version of the theorem comprised more than 30 pages with several bottlenecks. The author improved the proof using manually the FRONTAL method and the CROSSWORD method. The obtained by now results confirm the effect in cases with *infinite* sets of integers and they reveal possibilities for solving problems with *higher complexity*.

4. Some of the Advantages

The greater part of the seven procedures and their interaction inside the FRONTAL method are completely original. This method operates in the *environment of other methods* which are also proposed by the same author. The usage of this method in Number Theory leads to new mathematical results which are widely discussed and

acknowledged as original. Part of them are accepted for a publication in Australia. Another fraction is under consideration in AMS. The results from section 3 after Theorem 1 are only partially issued in the math periodicals. They are presented as an illustration of the method for the way in which a **front** of mutually related solutions can be formed. It is possible to set a way for applications of contemporary IT in computational mathematics, residing on the presented method.

5. Conclusions

A new IT method is proposed for the interactive construction of formulas and proofs in Number Theory. It follows from the consideration that *even a non-specialist* can make easy explainable solutions if she/he uses the present work with the described method. The method is multi-target oriented and its main part is domain independent.

Bibliography

[Altshuller 1979] G. S. Altshuller, *Creation as an Exact Science*, Moscow, Sov. Radio, 1979.

[Baldi 1995] P. Baldi, "Gradient learning algorithm overview: A general dynamical systems perspective," *IEEE Trans. on Neural Networks*, Vol. 6, pp. 182-195, Jan. 1995.

[EAEA 1997] *Evolutionary Algorithms in Engineering Applications*, D. Dasqupta and Z. Michalewicz (Eds.), Springer, Berlin etc., 1997.

[EA 1997] *Evolutionary Algorithms*, L. D. Davis et al. (Eds.), Springer - Verlag, New York etc., 1999.

[Hadamard 1975] J. Hadamard, Essai sue la Psychologie de l'Invention dans le Domaine Mathematique, Paris, Gautier-Villas, 1975.

[Jotsov1 1999] V. Jotsov, "On one approach in proof search and refinement," *J. Artif. Intell*. (National Academy of Sciences - Ukraine), No. 2, pp. 97-103, 1999.

[Jotsov2 1999] V. Jotsov, "On the usage of discovering systems' approaches in Number Theory," *IIT Working Paper* WP/83B, Bulg. Acad. of Sciences, 14pp., December 1999.

[Lenat 1976] B. Lenat, "AM: an artificial intelligence approach to discovery in mathematics as heuristic search," *Memo* AIM-286, Stanford University, CA, 1976.

[Lenat 1983] D. B. Lenat, "Why AM and EURISKO appear to work," *Artif. Intell*., Vol. 21, pp. 61-98, 1983.

[Narin'yani 2000] A. S. Narin'yani et. al., "Constraint programming and subdefinite models," preliminary version.

[Nordin 1999] P.Nordin and A. Eriksson, M. Nordahl, "Genetic reasoning: evolutionary induction of mathematical proofs," in R. Poli, P. Nordin, W. B. Langton, T. C. Fogarty (Eds.), Genetic Programming, *LNCS* 1598, Springer, Berlin etc., pp. 221-231, 1999.

[Polya 1963] G. Polya, *How to Solve It*, Princeton University Press, N.J., 1963.

[Riesel 1985] H. Riesel, Prime Numbers and Computer Methods for Factorization, Birkhauser, Boston, 1985.

[Shvetsov 1997] I. E.Shvetsov, V. V. Telerman, and D. M. Ushakov, "NeMo+: object-oriented constraint programming environment based on subdefinite models," Proc. Third Intern. Conf. on Principles and Practice of Constraint Programming (CP'97), *LNCS* 1330, Springer, Berlin etc., pp. 373-385, 1997.

Author information

Vladimir Jotsov – College of Library Science Education; Intstitute of Information Technologies – Bulgarian Academy of Sciences; 1113 Sofia, P.O. Box 161, Bulgaria; e-mail: jotsov@ieee.org

DISTANCES BETWEEN PREDICATES IN BY-ANALOGY REASONING SYSTEMS

V. Koval, Yu. Kuk

***Abstract**: The purpose is to develop expert systems where by-analogy reasoning is used. Knowledge "closeness" problems are known to frequently emerge in such systems if knowledge is represented by different production rules. To determine a degree of closeness for production rules a distance between predicates is introduced. Different types of distances between two predicate value distribution functions are considered when predicates are "true". Asymptotic features and interrelations of distances are studied. Predicate value distribution functions are found by empirical distribution functions, and a procedure is proposed for this purpose. An adequacy of obtained distribution functions is tested on the basis of the statistical χ^2 –criterion and a testing mechanism is discussed. A theorem, by which a simple procedure of measurement of Euclidean distances between distribution function parameters is substituted for a predicate closeness determination one, is proved for parametric distribution function families. The proposed distance measurement apparatus may be applied in expert systems when reasoning is created by analogy.*

***Keywords**: expert systems, production rules, predicates, distances between predicates, by-analogy reasoning.*

Introduction

Partnership systems are known to be the ones [1] able not only to use experts' knowledge, but also to derive themselves new knowledge from data accumulated in memory. They have means used to derive knowledge from data represented as statistical or empirical "object-feature-time"-type tables [2]. While inferences are obtained in traditional expert systems only deductively, partnership systems use additionally inductive inference features, by-analogy reasoning construction facilities and non-monotone reasonings [1]. The by-analogy reasoning creation basis is the rule that resembling conditions entail resembling effects in immediate proximity to known productions. Therefore, to construct a by-analogy reasoning mechanism, one should be able to compare a condition and an effect resemblance degree. A knowledge in expert systems is usually represented as "if $X_1 \,\&\, X_2 \,\&\, \ldots \,\&\, X_m$, then A"–type productions. Compare two productions, for instance, by some PROLOG language features, and left and right sides of both productions are compared. Productions coincide if compared predicates fully coincide. If productions do not coincide, then partnership systems take a non-coincidence degree into account. For this purpose, a distance between predicates is introduced in such systems, and it becomes possible to measure a degree to which one production resembles another. Thus, it is also possible to construct a by-analogy reasoning inference mechanism. By-analogy reasonings may be illustrated by the following example. Assume that it is necessary to check whether conditions $X_1 \,\&\, X_2 \,\&\, \ldots \,\&\, X_m$ lead to an effect A. An inference system detects that a knowledge base (KB) contains a resembling knowledge, i.e. "if $Y_1 \,\&\, Y_2 \,\&\, X_3 \,\&\, \ldots \,\&\, X_m$, then A ", a truth of which is equal to P. The conditions Y_1 and Y_2 do not coincide with X_1 and X_2 in this knowledge. Their non-coincidence degree is calculated. Hence, find a distance $d(X,Y)$ between the predicates $X = X_1 \,\&\, X_2$ and $Y = Y_1 \,\&\, Y_2$, and, if it does not exceed a threshold η, the conclusion is that A is probable. The truth of this inference is $P' < P$. A truth lowering value depends on a length of a distance between X and Y. The by–analogy inference rule scheme may be represented as

$$\frac{B', B \to A, d(B', B) < \eta}{A} \qquad (1)$$

Example 1. Let a predicate subject domain be a set of real functions $f(x)$ with one variable. Consider three predicates: 1) predicate B', i.e. "to be function $\frac{\sin(x)}{x}$"; 2) predicate B, i.e. "to be polynomial $f_n(x)$ with power exponent *n*=2*m*", where

$$f_n(x) = 1 - \frac{x^2}{2\cdot 3} + \frac{x^4}{2\cdot 3\cdot 4\cdot 5} - \frac{x^6}{2\cdot 3\cdot 4\cdot 5\cdot 6\cdot 7} + \cdots + (-1)^m \frac{x^{2m}}{2\cdot 3\cdot 4\cdot\cdots\cdot(2m+1)}\text{»;} \qquad (2)$$

and 3) predicate A, i.e. "to be represented as product of linear co-factors $f(x) = \prod_{i=1}^{n}(x-\alpha_i)$, where $\alpha_i, i = 1,\ldots,n$ are roots of equation $f(x) = 0$". The expression $B \to A$ is known [3]. Calculate the distance between B and B' by the following formula: $d(B', B) = \sup_x |g(x) - f_n(x)|$.

When n is chosen, this distance can be made shorter than any number η that is as low as possible: $d(B', B) \le \eta$. This fact be proved, if the function $g(x) = \frac{\sin(x)}{x}$ is expanded into Taylor series:

$$g(x) = \frac{\sin(x)}{x} = 1 - \frac{x^2}{2\cdot 3} + \frac{x^4}{2\cdot 3\cdot 4\cdot 5} - \frac{x^6}{2\cdot 3\cdot 4\cdot 5\cdot 6\cdot 7} + \cdots + (-1)^m \frac{x^{2m}}{2\cdot 3\cdot 4\cdot\cdots\cdot(2m+1)} + \ldots. \qquad (3)$$

Since $d(B', B) \le \eta$, then $B' \to A$ is the by-analogy inference (expression (1)), i.e. the function $g(x) = \frac{\sin(x)}{x}$ can also be expanded into linear co-factors. Since the roots of the equation $g(x) = \frac{\sin(x)}{x} = 0$ are $\pi, -\pi, 2\pi, -2\pi, \ldots$, then, when obtained by analogy, expansion (3) has the following form: $\frac{\sin x}{x} = (1 - \frac{x^2}{\pi^2})(1 - \frac{x^2}{4\pi^2})\cdots(1 - \frac{x^2}{n^2\pi^2})\cdots$. Pursuant to this formula, it is possible to determine the factor under x^2, i.e. $-(\frac{1}{\pi^2} + \frac{1}{4\pi^2} + \frac{1}{9\pi^2} + \cdots)$, and to make the latter equal to the factor under x^2, i.e. to $-\frac{1}{2\cdot 3}$, in expansion (3). And $\frac{1}{2\cdot 3} = \frac{1}{\pi^2} + \frac{1}{4\pi^2} + \frac{1}{9\pi^2} + \cdots$ is the result, from which the famous Euler formula follows:

$$1 + \frac{1}{4} + \frac{1}{9} + \frac{1}{16}\cdots = \frac{\pi^2}{6}.$$

1. Distances Between Predicates

1.1 Empirical Predicate Distribution Functions. An m-ary predicate $X = X(y_1,\ldots,y_m)$ is understood as a function, values of which are statements about *m* objects. Such objects are predicate argument values. A predicate is an object "feature" under $m = 1$ and it is a "relation" between *m* objects under $m > 1$.

Introduce the notions of empirical frequencies and of predicate value distribution functions needed in order to compare two "resembling" predicates X and Y. For this purpose, a point from Euclidean space R_m is brought in correspondence with each true statement. Consider the following cases.

1) $m = 1$ and a number of different true statements about an object feature is finite and equal to K. Bring an integer number, respectively, $1,2,\ldots K$ in correspondence with each such statement. Let there be n objects from some subject domain and, respectively, n true statements about a single feature of every such object. Define an empirical frequency for an i-th statement as $p_i = \frac{k_i}{n}$, where k_i is a number of i-th statements from among a whole number of n true statements. Pursuant to these frequencies, define an empirical distribution function $F_n^*(x)$ as a step function of a real variable x. This function is equal to zero under $x \le 1$, to p_1 under $1 \le x < 2$, to $p_1 + p_2$ under $2 \le x < 3$, … , and is to 1 under $x \ge K$. The derived empirical frequencies p_i, $i = 1,\ldots,K$, $\sum_{i=1}^{n} p_i = 1$, and $F_n^*(x)$ characterize this predicate well enough.

2) The second case is more complicated: $m=1$ and a number of different true statements have a power of a continuum. Construct $F_n^*(x)$. Take n objects from some subject domain and n true statements about a feature of every object. Bring a real number from the space R_1 in correspondence with each such statement. The result is that there are n numbers $x_1',\ldots,x_n'$ on the straight line R_1. Arrange these numbers in the ascending order, i.e. the variational series $x_{(1)} \le \ldots \le x_{(n)}$ is formed. Define $F_n^*(x)$ as a step function with the steps equal to $1/n$. It is the function of a real variable x, and it is equal to zero under $x \le x_{(1)}$, to k/n under $x_{(k)} \le x < x_{(k+1)}$, $k=1,\ldots,n-1$, and to 1 under $x \ge x_{(n)}$.

3) And now here is the general case when $m>1$ and a number of different true statements has a power of a continuum. Let there be n selections that have *m* objects from some subject domain and n true statements about relations between m objects from each selection. To reflect real relations between m objects, bring the m-dimensional vector from the space R_m in correspondence with each such statement. The result is that there are n vectors $x_1',\ldots,x_n'$ from R_m, where $x_i'=(x_{i,1}',\ldots,x_{i,m}')$. Define $F_n^*(x)$, where $x=(x_1,\ldots,x_m)\in R_m$, as follows. Consider a set $B_x=\{y\in R_m : y_i<x_i,\ i=1,\ldots,m\}$. Denote a number of $x_1',\ldots,x_n'$ by $\nu(B_x)$ as for the vectors that got into B_x. Assume the following: $F_n^*(x)=\nu(B_x)/n$, $x\in R_m$.

The $X=X_1\ \&\ X_2\ \&\ldots\&\ X_u$-type predicates are usually used in by-analogy reasoning systems. Let a predicate X_i be m_i-ary. Then, bring the values of X in correspondence with the points of the $m=\sum_{i=1}^{u} m_i$-dimensional space R_m. $F_n^*(x)$ is constructed in R_m in much the same way as in Case 3. It can be shown by analogy with Glivenko-Cantelli theorem [4] that the following assertion is valid for empirical distribution functions: $F_n^*(x)$ converges under $n\to\infty$ to some single limited predicate distribution function $G(x)$.

Example 2. Let the predicate subject domain be ceramics that belongs to different cultures [5]. Consider a predicate $X=X_1\ \&\ X_2\ \&\ldots\&\ X_8$, where predicate X_1 is Chernyakhovsky culture ceramics colour; X_2 is a contents of large-size ferrous inclusions in ceramics; X_3 is a contents of small-size admixture fragments in ceramics in per cent; X_4 is a contents of large-size admixture fragments in ceramics in per cent; X_5 is a size of admixture fragments; X_6 is a size of a dominating fraction; X_7 is a structure uniformity degree; and X_8 is a contents of quartz. On the basis of the available samples, it is possible to construct the function $F_n^*(x)$ for this predicate. Consider a "resembling" predicate $Y=Y_1\ \&\ Y_2\ \&\ldots\&\ Y_8$, where predicates X_i and Y_i, $i=1,\ldots,8$, are the same, but ceramics already belongs to some unknown culture. If the available samples of such an unknown culture are taken, it is also possible to construct the empirical distribution function $Q_n^*(x)$ for Y. If these functions turn out to be close to each other, there may be the by-analogy inference that an unknown ceramics belongs to Chernyakhovsky culture.

1.2. Calculating a Distance between Predicates. Differences in functions of distribution of two predicates can be used in by-analogy reasoning systems in order to compare two "resembling" predicates. Assume that predicates X and Y have initially the same arity m. Let $G(x)$ and $Q(x)$, $x\in R_m$, be predicate value probability distribution functions, respectively, for X, the first predicate, and for Y, the second predicate. In practice, empirical distribution functions or distribution function estimates are used as the former ones. They are selected from appropriate standard parametric distribution function families and tested for adequacy. The distribution function estimate derivation methodology is considered below.

Definition 1. *A distance $d(X,Y)$ between predicates X and Y is a distance $d(G,Q)$ between two value distribution functions $G(x)$ and $Q(x)$ when these predicates are true under their values.*

Consider the case in the by-analogy reasoning systems when X and Y have different arities, i.e., respectively, l and m under $l<m$. Assume that an effect A follows from X. Since the case $l<m$ is present, then Y describes not only a relation with objects described by X, but also with additional objects. Therefore, it is possible to narrow this relation to relations of l objects considered by X. For this purpose, substitute the

marginal distribution function $\hat{Q}(x_1,\ldots,x_l)=Q(x_1,\ldots,x_l,\infty,\ldots,\infty)$ of Y for its distribution function $Q(x)=Q(x_1,\ldots,x_m)$ and determine the distance between the predicates as the one between $G(x)$ and $\hat{Q}(x)$, where the vectors *x* are already of the same dimensionality. If this distance is small, then there is the by-analogy inference: A follows from Y. Evidently, such conclusion cannot be made under $l>m$.

Consider the distance d between two predicates $X=X_1\,\&\,X_2\,\&\ldots\&\,X_u$ and $Y=Y_1\,\&\,Y_2\,\&\ldots\&\,Y_w$ for the case when a "feature" or a "relation", described by each separate predicate, are by no means associated with "features" or "relations" described by other predicates. Let $G_{X_1},G_{X_2},\ldots,G_{X_u}$ and $Q_{Y_1},Q_{Y_2},\ldots,Q_{Y_w}$ be the distribution functions, respectively, for $X_1,X_2,\ldots,X_u$ and $Y_1,Y_2,\ldots,Y_w$, and the predicates have the same indices and arities. Then, d between $X=X_1\,\&\,X_2\,\&\ldots\&\,X_u$ and $Y=Y_1\,\&\,Y_2\,\&\ldots\&\,Y_w$ is equal to the distance between two products of the respective distribution functions $G_X=G_{X_1}\cdot G_{X_2}\cdot\ldots\cdot G_{X_u}$ and $Q_Y=Q_{Y_1}\cdot Q_{Y_2}\cdot\ldots\cdot Q_{Y_w}$. The requirement as for the arities can be weakened in by-analogy reasoning systems. Let an effect A follows from $Y=Y_1\,\&\,Y_2\,\&\ldots\&\,Y_w$ Then, to make A follow from $X=X_1\,\&\,X_2\,\&\ldots\&\,X_u$, it is necessary to make the arities of $X_1,X_2,\ldots,X_u$ not lower than the ones of the corresponding $Y_1,Y_2,\ldots,Y_w$. Besides this, when a distance between X and Y is calculated, the corresponding distribution functions $\hat{G}_{X_1},\hat{G}_{X_2},\ldots,\hat{G}_{X_m}$ must be substituted for the functions $G_{X_1},G_{X_2},\ldots,G_{X_u}$, and such distance itself must not exceed η. X may consist of such additional predicates that are not contained by Y. The distribution functions for these predicates are replaced by 1.

If a formula for d between predicates is chosen correctly, it is possible to use further on "good" features of this distance, for instance, the distance calculation procedure itself may be simplified. Consider various formulas used to calculate distances between X and Y. The distance

$$d(X,Y)=d(G,Q)=\sup_x\left|G(x)-Q(x)\right| \tag{4}$$

means an absolute deviation of values for one distribution function with respect to another distribution function at each point and the distance

$$d(X,Y)=d(G,Q)=\int(G(x)-Q(x))^2\,dQ(x) \tag{5}$$

takes a root mean square deviation of these values into account.

Example 3. Calculate the distance d between the predicates B' and B from Example 1. For every x, the real value for B' is equal to $g(x)=\dfrac{\sin(x)}{x}$. Therefore, the distribution function for this predicate is equal to zero under $x<g(x)$ and to 1 under $x\geq g(x)$. The values of B correspond to the values of the polynomial $f_n(x)$ that, under different and sufficiently large n, are arranged in a certain way within the interval Δ of the following form: $g(x)-\eta\leq f_n(x)\leq g(x)+\eta$. However, when n increases, the points $f_n(x)$ approach the point $g(x)$ because of $f_n(x)\to g(x)$. The distribution functions $Q(x)$ for these points are not found, since only the upper estimate for d between B' and B is important. The following is made: move each of these points away from $g(x)$ in such a way that they fill in the interval Δ uniformly. The result is that the distribution function $\tilde{Q}(x)$ in its new position becomes uniform, but the distance between $G(x)$ and the new $\tilde{Q}(x)$ increases here in comparison with the previous one between $G(x)$ and $Q(x)$ Therefore: $d(B,B')=d(G,Q)<d(G,\tilde{Q})$. Since $\tilde{Q}(x)$ is equal to zero under $x<g(x)-\eta$, to $-g(x)+\eta+x\dfrac{1+g(x)-\eta}{g(x)+\eta}$ under $x\in\Delta$ and to 1 under $x\geq g(x)+\eta$, then, if formula (5) is used, the following expression takes place:

$d(B,B') < d(G,\tilde{Q}) = 2\int_0^{\eta} \frac{x^2}{4\eta^2} dx = \frac{\eta}{6}$. Hence, the upper estimate is derived for $d(B,B')$. Thus, if $\sup_x |g(x) - f_n(x)| \le \eta$, then $d(B,B') < \eta/6$. Therefore, these formulas for the distances are equivalent.

1.3 Kulbak–Leibler Distance, χ^2–Distance, Hellinger Distance. Consider now different types of distances between two predicates X and Y for the case when their distribution functions Q and G have, respectively, the densities $q(x)$ and $g(x)$ as for a measure μ. The Lesbegue measure may be used for one group of distribution functions (absolutely continuous distributions) and a counting measure may be taken for another group (discrete distributions) as μ. Let N_Q be a carrier of Q ($N_Q = \{x : q(x) > 0\}$), and let G be denoted by N_G ($N_G = \{x : g(x) > 0\}$). The Kulbak–Leibler distance between X and Y is calculated in the following way:

$$r_1(X,Y) = r_1(\mathrm{G},Q) = \int_{N_G} \ln \frac{g(x)}{q(x)} g(x)\mu(dx).$$

The χ^2–distance between X and Y is

$$r_2(X,Y) = r_2(\mathrm{G},Q) = \int_{N_Q \cup N_G} \frac{(q(x) - g(x))^2}{g(x)} \mu(dx).$$

The values of $r_1(X,Y)$ and $r_2(X,Y)$ are more than or equal to zero. However, the equalities $r_1(X,Y) = 0$ and $r_2(X,Y) = 0$ are possible only under $Q = G$. Since $r_1(X,Y)$ and $r_2(X,Y)$ are not the symmetric functions of Q and G, then $r_1(X,Y)$ and $r_2(X,Y)$ are not the distances in the general case because of $r_1(X,Y) \ne r_1(Y,X)$ and $r_2(X,Y) \ne r_2(Y,X)$. Nevertheless, essentially speaking and from the statistical point of view, $r_1(X,Y)$ and $r_2(X,Y)$ characterize a deviation of Q from G.

The Hellinger distance between X and Y is

$$r_3(X,Y) = r_3(\mathrm{G},Q) = \int_{N_Q \cup N_G} \left(\sqrt{g(x)} - \sqrt{q(x)}^2\right) \mu(dx).$$

and it is already the symmetric function for X and Y. The value $\sqrt{r_3(Q,G)}$ possesses all the metric characteristics between the functions $\sqrt{q(x)}$ and $\sqrt{g(x)}$ in the metrical space L_2.

Consider the features of these distances, important when a predicate resemblance threshold is chosen. If a predicate closeness degree is characterized by such distances when $q(x)/g(x)$ is close to 1, then the following result turns out to take place: $r_1(Q,G) \approx \frac{1}{2} r_2(Q,G) \approx 2 r_3(Q,G)$.

Asymptotically, all the distances behave in the same way. To study this asymptotic feature, assume that G and Q for X and Y are taken from one and the same parametric family and defined, respectively, by the parameters θ and $\theta + \Delta$. Then, the rate of the convergence to zero for the distance between X and Y is equal to $O(\Delta^2)$ under $\Delta \to 0$. This fact follows from the asymptotic equality $r_3(\Delta) \approx \frac{I(\theta)}{4}\Delta^2$, where $I(\theta)$ is the Fisher information found by the formula

$$I(\theta) = \int \frac{(g'_\theta(x))^2}{g_\theta(x)} \mu(dx).$$

1.4 Predicate Comparison Procedure Simplification Theorem. The predicate resemblance determination procedure falls into two stages: 1) calculate a distance between predicates; and 2) compare a calculated distance with a threshold η. Let G and Q be distribution functions for predicates X and Y that belong to the same parametric family $\Psi = (G_\theta | \theta \in \Theta)$ and differ only in their parameters. Assume that G_{θ_1} and G_{θ_2} are,

respectively, the predicate value distribution functions for X and Y. Consider the Kulbak-Leibler, χ^2- and Hellinger distances as the ones between predicates: $\rho_i(\theta_1,\theta_2)$, $i=1,2,3$. The following theorem is true.

Theorem 1. *Assume that value distribution functions for predicates X and Y belong to a parametric distribution function family $\Psi=(G_\theta/\theta\in\Theta)$. Let the following conditions be met: 1) a parametric set Θ is compact; 2) $G_{\theta_1}\neq G_{\theta_2}$ under $\theta_1\neq\theta_2$; 3) for every $\theta\in\Theta$, Fisher information is restricted: $0<I(\theta)\le 4b<\infty$. Then, $\rho_i(\theta_1,\theta_2)\le\delta$, $i=1,2,3$ is equivalent to $(\theta_1-\theta_2)^2\le\delta/b_i$, where b_i, $i=1,2,3$ are constant, $b_1=2b$, $b_2=4b$, $b_3=b$.*

This theorem reduces the predicate resemblance determination procedure to the simple procedure by which a Euclidean distance between distribution function parameters is determined. The Θ-set compactness condition is not assumed to be restricting and it means that Θ is restricted. The second condition means that $\rho_i(\theta_1,\theta_2)>0$ takes place under $\theta_1\neq\theta_2$.

1.5. Distribution Function Estimates. As a rule, a predicate distribution function is not known. It is not very convenient to deal with empirical distribution functions. Therefore, the already known classes of distributions are used and estimates for $G(x)$ are created. Assume that an unknown estimate of $G(x)$ for a predicate X belongs to $\Psi=(G_\theta/\theta\in\Theta)$. Construct an empirical function G_n^* for X. Let $G*$ be a function from Ψ that is closest to G_n^* as for a distance d, i.e. $d(G*,G_n^*)=\min\limits_{\Pi\in\Psi}d(\Pi,G_n^*)$. $G*$ with the parameter $\theta*$ is an estimate for $G(x)$ as for a minimum of d.

Consider the practical methods used to create the estimates for distribution functions. First of all, describe the χ^2–procedure that helps to find estimates. In this case, the distance $d(G,Q)=\sum\limits_{i=1}^{r}\frac{(P_G(\Delta_i)-P_Q(\Delta_i))^2}{P_G(\Delta_i)}$ is used as d; $\Delta_1,\ldots,\Delta_r$ are non-intersecting sets of a predicate value space R and their union is equal to R; $P_G(\Delta_i)=\int\limits_{\Delta_i}dG(x)$, $P_Q(\Delta_i)=\int\limits_{\Delta_i}dQ(x)$, $i=1,\ldots,r$. Take $G_n^*(x)$ as $Q(x)$. The estimate $\theta*$ as for the given minimum distance is a value of θ, and

$$d(G_\theta,G_n^*)=n\sum_{i=1}^{r}\frac{\left(P_\theta(\Delta_i)-\frac{\nu_i}{n}\right)^2}{P_\theta(\Delta_i)}=\sum_{i=1}^{r}\frac{(nP_\theta(\Delta_i)-\nu_i)^2}{nP_\theta(\Delta_i)}. \qquad (7)$$

is minimized under this distance; in the present case, $\nu_i=nG_n^*(\Delta_i)$ is a number of predicate values that got into the set Δ_i and under which a predicate is “true”. Differentiate expression (7) with respect to the parameters, the components of which make up the vector θ, make the derivatives equal to zero, and the equation system is derived relative to unknown parameters. Solve this system and find the estimates for the parameters. The obtained $G*(x)$ is then tested for adequacy. If a test result shows that $G*(x)$ is not adequate to the data, then an initial distribution function family should be changed.

Consider the practically important maximum likelihood method also used to derive the estimates. To create a maximally likely estimate means to define one more important distance between an arbitrary Q and G_θ from $\Psi=(G_\theta/\theta\in\Theta)$. It is assumed that G_θ possesses a density $g_\theta(x)$ with respect to a measure μ. Such a distance is expressed by the formula $\rho(G_\theta,Q)=-\int\ln g_\theta(x)Q(dx)$. If an empirical G_n^* is taken as Q, then the estimate for θ is called the maximum likelihood estimate and it minimizes the distance $\rho(G_\theta,G_n^*)$. The yielded function estimates have the “good” features, i.e. they are efficient and asymptotically not biased.

1.6. Testing for Adequacy. Obtained distribution function estimates are tested for adequacy before a distance between predicates is found by means of them. An adequacy of a found function is tested for by the χ^2–statistics. The testing mechanism is as follows. Consider the hypothesis that, when a predicate is “true”, probable predicate

values are distributed by $G^*(x)$. Divide a predicate value space into a finite number of sets $\Delta_1,\ldots,\Delta_r$ without common points. Calculate the values of $p_i = P_G(\Delta_i)$. Determine the frequencies ν_i, i.e. a number of predicate values under which it is "true" and that got into a set Δ_i. Calculate the statistics $\chi^2 = \sum_{i=1}^{r} \frac{(np_i - \nu_i)^2}{np_i}$. It is possible to show by analogy with [5] that the χ^2-statistics distribution function does not depend on an initial predicate value distribution function at all under $n \to \infty$. The former function is expressed by the formula $w_{r-1}(x) = 2^{\frac{1-r}{2}} \Gamma^{-1}\left(\frac{r-1}{2}\right) x^{\frac{r-3}{2}} e^{-\frac{x}{2}}$, $x > 0$ and helps to find the point $x_{0.05}$ for which the expression $\int_{x_{0.05}}^{\infty} w_{r-1}(x)dx = 0.05$ takes place. If $\chi^2 > x_{0.05}$, then the choice of a distribution function is wrong.

2. A By-Analogy Reasoning System Flowchart

Figure 1 depicts a by-analogy reasoning system flowchart. Let the following request be received by the system: "Is effect A possible when conditions $X_1 \,\&\, X_2 \,\&\, \ldots \,\&\, X_m$ are met?"

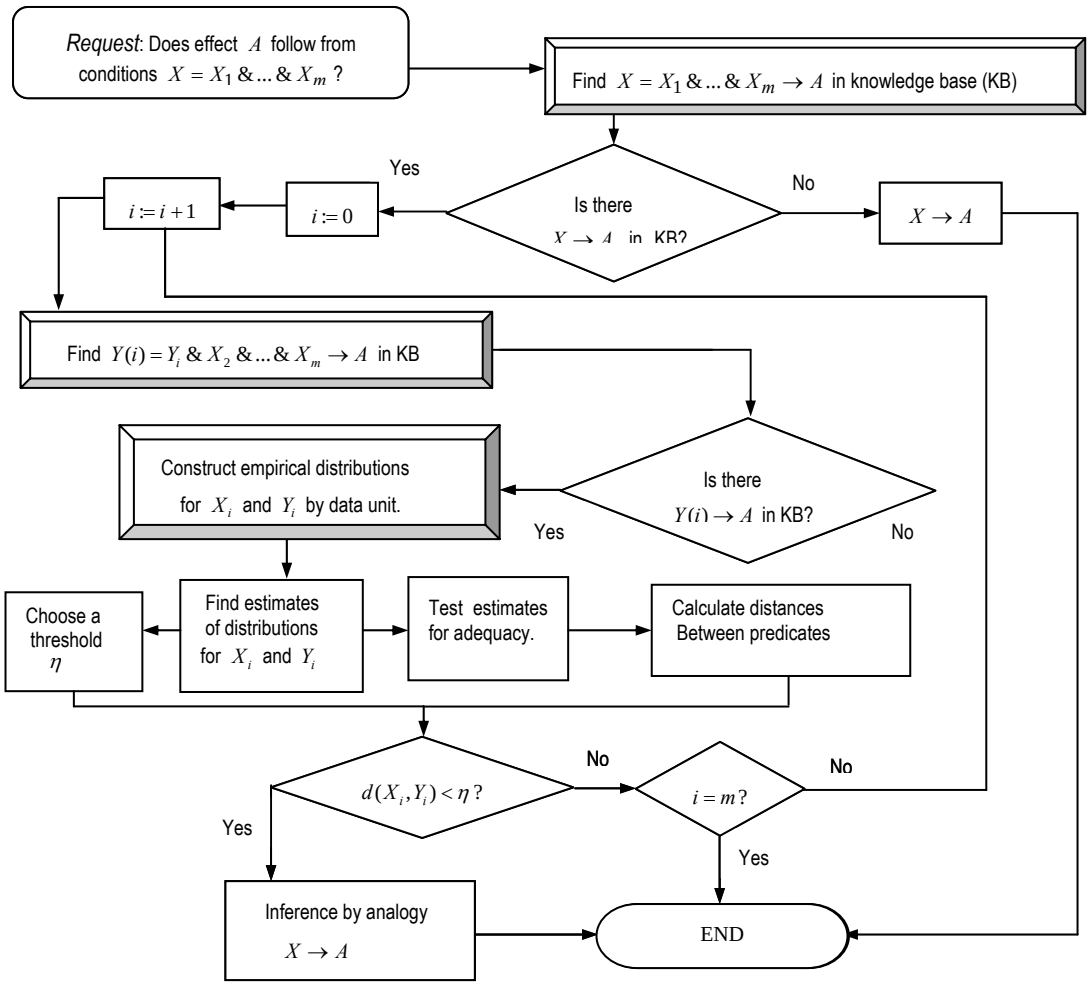


Figure 1. By-Analogy Reasoning System: A Flowchart

The production rule "if $X_1 \& X_2 \& \ldots \& X_m$ then A" is sought for in the KB. If it is found, the answer is positive. If it is absent, the production rule "if $Y_1 \& X_2 \& \ldots \& X_m$, then A" is sought for, the conditions of which contain the same predicate names as in the request conditions, but the first predicate differs from first predicate in the request. If this rule is not found in the KB, then such a production rule is sought for, the conditions of which differ from the request conditions already in the second predicate: "if $X_1 \& Y_2 \& \ldots \& X_m$ then A". And so on. For more certainty, let the procedure result be that the desired production rule "if $X_1 \& X_2 \& ... ... \& X_{i-1} \& Y_i \& X_{i+1} \& ... \& X_m$ then A" is found in the KB at the i-th step. However, the predicate Y_i does not coincide in this rule with the predicate X_i. Therefore, the procedure is started up that determines a closeness of predicates that do not coincide. The data about distribution of the values of X_i and Y_i, under which they are "true", are extracted from the database. Pursuant to these data, the empirical $F_n^*(x)$ and $G_n^*(x)$ are constructed for the values under which they are "true". In accordance with $F_n^*(x)$ and $G_n^*(x)$, the estimates of $F^*(x)$ and $G^*(x)$ are found as for X_i and Y_i. To find such estimates, introduce a distance between the distribution functions. To derive the estimates, find the distribution functions from the specified families that are closest to the found empirical functions in the sense of an introduced distance metrics. The obtained estimates for $F^*(x)$ and $G^*(x)$ are then tested for their adequacy as for the available empirical data by the χ^2–criterion. If the estimates for $F^*(x)$ and $G^*(x)$ do not fit available empirical data, choose another family where the same estimates are sought for again. The adequate estimates of $F^*(x)$ and $G^*(x)$ are yielded, and a distance d between the considered predicates is calculated by means of them. This distance determines a degree of "resemblance" or "closeness" for X_i and Y_i. Predicates are close if a distance between them does not exceed some threshold. As a threshold, a sufficiently small positive number η is chosen, and a value of this number states a by-analogy inference truth. Under $d(X_i, Y_i) \le \eta$, there is the following by-analogy inference: "if $X_1 \& X_2 \& \ldots \& X_m$ then A". If $d(X_i, Y_i) > \eta$ takes place, then a found production rule is rejected, and a new production rule is sought for that differs from a required one in a next-coming $(i+1)$-th predicate.

Conclusion

The paper considers different-type distances between predicates. They are the distances between predicate value distribution functions under which predicates are "true". The asymptotic features of such distances and the interrelation between the latter are studied. The paper proposes the procedure used to find distributions of predicate values for the case when predicates are true. The distribution functions are found by the empirical distribution ones. The paper also deals with the mechanism that tests an adequacy of a yielded distribution function on the basis of the χ^2–criterion. The predicate resemblance determination procedure is replaced by the simple procedure that determines Euclidean distances between distribution function parameters. The replacement theorem is proved for the parametric families. The proposed distances can be used in expert systems in order to construct by-analogy reasonings.

Bibliography

[1] N.G. Zagoruyko. Application Methods for Data and Knowledge Analysis. Novosibirsk, 1999, 269p.

[2] V.N. Koval, Yu.V. Kuk. Finding Unknown Rules of an Environment by Intelligent Goal-Oriented Systems, "Information Theories and Applications", International Journal, vol. 17, N 3, p. 127-138, Sofia, 2001.

[3] G. Polya. Mathematics and Plausible Reasoning. Prinston, New Jersey. 1954, 464p.

[4] A.A. Borovkov. Mathematical Statistics. Moscow, Nauka, 1984, 472p.

[5] Krug G.M., Krug O.Yu. A Mathematical Method for Classification of Ancient Ceramics. Proc. Institute of Archeology, Academy of Sciences of the USSR. Moscow, Nauka, 1965, p. 317-323.

Author information

Valeriy Koval – Institute of Cybernetics, Head of Department, address: 03680, Kiev, Prospect Glushkova, 40, Ukraine; e-mail: icdepval@ln.ua
Yuriy Kuk - Institute of Cybernetics, senior scientific researcher, Ukraine; Kiev, e-mail: vkyk@svitonline.com .

ADMISSIBLE SUBSTITUTIONS IN SEQUENT CALCULI AUTHORS

A. V. Lyaletski

Abstract: *For first-order classical logic a new notion of admissible substitution is defined. This notion allows optimizing the procedure of the application of quantifier rules when logical inference search is made in sequent calculi. Our objective is to show that such a computer-oriented sequent technique may be created that does not require a preliminary skolemization of initial formulas and that is efficiently comparable with methods exploiting the skolemization. Some results on its soundness and completeness are given.*

Keywords: completeness, first-order logic, quantifier rule, sequent calculus, skolemization, soundness

Introduction

Investigations in computer-oriented reasoning gave rise to the appearance of various methods for the proof search in the classical 1st order logic. Particularly, sequent calculi were suggested by Gentzen [1]. But their practical application as a logical technique (without preliminary skolemization) of the intelligent systems has not received wide use: preference is usually given to the resolution-type methods. This is explained by higher efficiency of the resolution-type methods as compared to sequent calculi, which is mainly connected with different possible orders of the quantifier rule applications in sequent calculi while resolution-type methods, due to skolemization, are free from this deficiency.

In its turn, the deduction process in sequent calculi reflects sufficiently well natural theorem-proving methods which, as a rule, do not include preliminary formula skolemization so that reasonings are performed within the scope of the signature of the initial theory. This feature of sequent calculi becomes important when some interactive mode of proof is developed since it is preferable to present the output information concerning the proof search in the form usual for man. That is now the problem of the efficient quantifier manipulation makes its appearance.

When quantifier rules are applied, some substitution of selected terms for variables is made. To do this step of deduction sound, certain restrictions are put on the substitution. The substitution, satisfying these restrictions, is said to be admissible. Here we investigate the classical notion of admissible substitution and show how it can be modified so that efficient sequent calculi can be finally obtained. We use the calculus G [2] for the demonstration of the way of the construction of such a modification denoted by mG here. Note that when constructing mG, we don't touch upon any procedure of selection of propositional rules and terms substituted, focussing our attention on quantifier handling only.

Genzen's Notion of Admissible Substitutions

Classical quantifier rules, substituting arbitrary structure terms when applied "from bottom to top", are usually of the following form [2]:

$\Gamma_1, A[t/x], \forall xA, \Gamma_2 \rightarrow \Gamma_3$ ($\forall$: left)

$\Gamma_1, \forall xA, \Gamma_2 \rightarrow \Gamma_3$

$\Gamma_1 \rightarrow \Gamma_2, A[t/x], \exists xA, \Gamma_3$ ($\exists$: right)

$\Gamma_1 \rightarrow \Gamma_2, \exists xA, \Gamma_3$

where the term t is required to be free for the variable x in the formula A. This restriction of the substitution of t for x gives Gentzen's (classical) notion of an admissible substitution, which proves to be sufficient for the needs of the proof theory. But it becomes useless from the point of view of efficiency of computer-oriented theorem-proving methods. It is clear from the following example.

Consider a sequent $A_1, A_2 \rightarrow B$, where A_1 is $\forall x_1 \exists y_1 (R_1(x_1) \vee R_2(y_1))$, A_2 is $\forall x_2 \exists y_2 (R_1(y_2) \vee R_2(x_2))$, and B is $\exists x_3 \forall y_3 (R_2(x_3) \vee R_3(y_3))$. The provability of this sequent in calculus G will is established below, while here we notice that quantifier rules must be applied to all the quantifiers occurring in A_1, A_2, and B. Therefore, classical notion of admissible substitution yields 90 (= 6!/(2!*2!*2!)) different orders of the quantifier rule applications ("from bottom to top") to the sequent $A_1, A_2 \rightarrow B$. It is clear that resolution type methods allow avoiding this redundant work.

Kanger's Notion of the Admissible Substitutions

To optimize procedure of the applications of quantifier rules, S.Kanger suggested in [2] his calculus of Gentzen type, denoted here by K. In calculus K a "pattern" of a deduction tree is first constructed with the help of special variables, the so called parameters and dummies. At some times an attempt is made to convert a "pattern" into proof tree to complete the deduction process. In case of failure, the process is continued.

The main difference between K and G consists in a special modification of the above quantifier rules and in a certain splitting (in K) of the process of the "pattern" construction into stages. In K the rules ($\forall$: left) and ($\exists$: right) are of the following form:

$\Gamma_1, A[d/x], \forall xA, \Gamma_2 \rightarrow \Gamma_3$

$\Gamma_1, \forall xA, \Gamma_2 \rightarrow \Gamma_3$ $\quad d/t_1,...,t_n$

$\Gamma_1 \rightarrow \Gamma_2, A[d/x], \exists xA, \Gamma_3$

$\Gamma_1 \rightarrow \Gamma_2, \exists xA, \Gamma_3$ $\quad d/t_1,...,t_n$

where $t_1,...,t_n$ are the terms occurring in the conclusion of the rules, d is the dummy, and $d/t_1,...,t_n$ denotes that when an attempt is made to convert "pattern" into proof tree, the dummy d must be replaced by one of the terms $t_1,...,t_n$. The replacement of dummies by terms is made in the end of every stage, and at every stage the rules are applied in a certain order.

This scheme of the deduction construction in calculus K leads to a notion of the Kanger-admissible substitution, which is more efficient than the classical one. Thus in the above example it yields only 6 (=3!) variants of different possible orders of the quantifier rule applications (but none of these variants is preferable). Despite this, the Kanger-admissible substitutions still did not allow to attain the efficiency comparable with that when the skolemization is made. It is due to the fact that, as in case of the classical admissible substitution, it is required to select a certain order of the quantifier rule applications when an input sequent is deduced, and, if it proves to be unsuccessful, the other order of applications is tried, and so on.

New Notion of Admissible Substitutions

For constructing the modification mG of calculus G from [2], let us introduce a new notion of admissible substitutions in order to get rid of the dependence of the deduction efficiency in sequent calculi on different possible orders of quantifier rule applications. The main idea is to determine, proceeding from quantifier

structures of formulas of an input sequent and a substitution under consideration, would there exists a sequence of desired quantifier rule application. (This notion was used in slightly modified form in [3].)

Substitution is defined as a finite (maybe, empty) set of ordered pairs, every of which contains a variable and a term and is written in the form t/x, where x is the variable and t is the term of substitution [4].

We assume that besides usual variables there are two countable sets of special variables, namely of parameters and dummies.

Let P be a set of sequences of parameters and dummies, and s be a substitution. Put T(P,s) = {<z,t,p>: z is the variable of s, t is the term of s,p P, and z lies in p to the left of some parameter from t}. The substitution s is said to be admissible for P if and only if (1) the variables of s are only dummies and (2) in T(P,s) there are no elements $<z_1,t_1,p_1>$,..., $<z_n,t_n,p_n>$ such that $t_2/z_1 \in s,...,t_n/z_{(n-1)} \in s$, $t_1/z_n \in s$ (n>0).

Calculus mG

As in the case of calculus G, its modification mG deals with formulas, except that in mG every formula from a sequent has a certain sequence of parameters and dummies. Therefore, it is convenient to define calculus mG by means of the pairs <p,A>, where A is the formula and p - the sequence (word) of parameters and dummies. Also, it will be assumed that the empty sequence is always added to all formulas from the input sequent (that is, from the sequent to be proved).

The rules of the calculus mG are the following.

Propositional rules:

$\Gamma_1,<p,A>,<p,B>,\Gamma_2 \to \Gamma_3$ $\quad$ $\Gamma_1 \to \Gamma_2,<p,A>,\Gamma_3$ $\Gamma_1 \to \Gamma_2,<p,B>,\Gamma_3$

$\Gamma_1,<p,A \wedge B>,\Gamma_2 \to \Gamma_3$ $\quad$ $\Gamma_1 \to \Gamma_2,<p,A \wedge B>,\Gamma_3$

$\Gamma_1,<p,A>, \Gamma_2 \to \Gamma_3$ $\Gamma_1 \to \Gamma_2,<p,B>,\Gamma_3$ $\quad$ $\Gamma_1 \to \Gamma_3,<p,A>,<p,B>,\Gamma_2$

$\Gamma_1,<p,A \vee B>,\Gamma_2 \to \Gamma_3$ $\quad$ $\Gamma_1 \to \Gamma_2,<p,A \vee B>,\Gamma_3$

$\Gamma_1, \Gamma_2 \to <p,A>$ $\quad$ $<p,B>,\Gamma_1,\Gamma_2 \to \Gamma_3$ $\quad$ $<p,A>,\Gamma_1 \to \Gamma_3,<p,B>,\Gamma_2$

$\Gamma_1,<p,A \supset B>,\Gamma_2 \to \Gamma_3$ $\quad$ $\Gamma_1 \to \Gamma_2,<p,A \supset B>,\Gamma_3$

$\Gamma_1, \Gamma_2 \to <p,A>,\Gamma_3$ $\quad$ $<p,A>,\Gamma_1 \to \Gamma_2,\Gamma_3$

$\Gamma_1,<p,\neg A>,\Gamma_2 \to \Gamma_3$ $\quad$ $\Gamma_1 \to \Gamma_2,<p,\neg A>,\Gamma_3$

Quantifier rules:

$\Gamma_1, <pd,A[d/x],>,<p,\forall xA>,\Gamma_2 \to \Gamma_3$ $\quad$ ($\forall$: left')

$\Gamma_1,<p,\forall xA>,\Gamma_2 \to \Gamma_3$

$\Gamma_1, \to \Gamma_2,<pd,A[d/x]>,<p,\exists xA>, \Gamma_3$ $\quad$ ($\exists$: right')

$\Gamma_1, \to \Gamma_2,<p,\exists xA>, \Gamma_3$

$\Gamma_1 \to \Gamma_2,<pz,A[z/x]>, \Gamma_3$ $\quad$ ($\forall$: right')

$\Gamma_1 \to \Gamma_2,<p,\forall xA>,\Gamma_3$

$\Gamma_1, <pz,A[z/x]>,\Gamma_2 \to \Gamma_3$ $\quad$ ($\exists$: left')

$\Gamma_1,<p,\exists xA>,\Gamma_2 \to \Gamma_3$

Here d is a new dummy, z is a new parameter, p is a sequence of parameters and dummies, Γ_1, Γ_2, and Γ_3 are arbitrary sequences of pairs, consisting of sequences (of dummies and parameters) and formulas, A, B are arbitrary formulas.

Applying first rules "from bottom to top" to the input sequent and afterwards to its "heirs", and so on, we finally obtain a so-called deduction tree.

A deduction tree D is called a proof tree for the input sequent (in mG) if and only if there exists a substitution of terms for variables, s, such that (1) s is admissible for set of all sequences of parameters and dummies from D and (2) after application of s to the formulas from all upper sequents of D we obtain axioms, that is, the sequents $\Gamma_1 \to \Gamma_2$ such that Γ_1 and Γ_2 contain a common formula.

The main result concerning the calculus mG is as follows.

Theorem. Let $A_1,...,A_m,B_1,...,B_n$ be the formulas of the 1st order language. There exists a proof tree for the input sequent $<,A_1>,...,<,A_m> \to <,B_1>,...,<,B_n>$ in calculus mG if and only if there exists a proof tree for the input sequent $A_1,...,A_m \to B_1,...,B_n$ in calculus G.

Proof.

(=>) Let D be a proof tree for the input sequent $<,A_1>,...,<,A_m> \to <,B_1>,...,<,B_n>$ in the calculus mG, and s be a substitution, which converts all upper sequents of D into axioms and is admissible for set P of all sequences of parameters and dummies from D. Without any loss of generality, we may assume that terms of s do not contain dummies for otherwise these dummies could be replaced by a constant, say, c0.

Since s is admissible for P, it is possible to construct the following sequence p consisting of parameters and dummies which form the sequences of P:

(i) every p' P is a subsequence of p, and

(ii) the substitution s is admissible for {p} (i.e. there is no an element <z,t,p> T({p},s) such that $t/z \in s$.

Such a sequence p may be generated, for example, by the convolution algorithm from [3], applied to a list of all the sequences from P provided that in the convolution algorithm are treated parameters as existence quantifiers, and dummies universal quantifiers.

Property (i) of the sequence p and formulation of the propositional and quantifier rules permit to make the following assumption:

When D was constructed, propopositional and quantifier rules were applied ("from bottom to top") in the order that corresponds to looking through p from the left to right: i.e. when the first quantifier rule was applied, the first variable (a parameter or a dummy) of p was generated, when the second quantifier rule was applied, the second variable of p was generated, and so on.

Now it is possible to covert the tree D into proof tree D' for the input sequent $A_1,...,A_m \to B_1,...,B_n$ in calculus G. To do this, let us "repeat" the process of the construction of D in the above order p and execute the following transformations:

1) Suppose that in a processed node of D one of the following rules was applied:

Γ_1, <pd,A[d/x],>,<p,$\forall$xA>,$\Gamma_2 \to \Gamma_3$ ($\forall$: left')

Γ_1,<p,$\forall$xA>,$\Gamma_2 \to \Gamma_3$

or

$\Gamma_1, \to \Gamma_2$,<pd,A[d/x]>,<p,$\exists$xA>, Γ_3 ($\exists$: right')

$\Gamma_1, \to \Gamma_2$,<p,$\exists$xA>, Γ_3

and t/d s for some term t. The term t is free for d in A, because the order of applications of quantifier rules is reflected by p, and property (ii) is satisfied. Therefore, the admissibility in the classical sense will be observed when the above rules ($\forall$: left') and ($\exists$: right') are replaced in D by rules ($\forall$: left) and ($\exists$: right) of the calculus G: and all other occurrences of d in D are replaced by t.

Γ_1, A[t/x], $\forall$xA, $\Gamma_2 \to \Gamma_3$ ($\forall$: left)

Γ_1,$\forall$xA, $\Gamma_2 \to \Gamma_3$

or

$\Gamma_1 \to \Gamma_2, A[t/x], \exists xA, \Gamma_3$ ($\exists$: right)

$\Gamma_1 \to \Gamma_2, \exists xA, \Gamma_3$

2) In other cases the rules of the calculus mG are replaced by their analogs from G by a simple deleting of sequences of parameters and dummies from these rules.

It is evident that D' is a deduction tree in the calculus G. Furthermore, the way of conversion of D into D' allows making the conclusion that upper sequents of D' are axioms of the calculus G. Thus, D' is a proof tree for the input sequent $A_1,...,A_m \to B_1,...,B_n$ in G.

(<=) Let D' be a proof tree for the input sequent $A_1,...,A_m \to B_1,...,B_n$ in G. Convert D' into tree D, which, as be can seen bellow, is a proof tree for the input sequent $<,A_1>,...,<,A_m> \to <,B_1>,...,<,B_n>$ in mG. For this purpose "repeat" (“from bottom to top”) a process of construction of D', replacing in D' every rule application by its analog in mG and subsequently generating substitution s. (Initially s is the empty substitution.)

1) If an applied rule is one of the following:

$\Gamma_1, A[t/x], \forall xA, \Gamma_2 \to \Gamma_3$ ($\forall$: left)

$\Gamma_1, \forall xA, \Gamma_2 \to \Gamma_3$

or

$\Gamma_1 \to \Gamma_2, A[t/x], \exists xA, \Gamma_3$ ($\exists$: right)

$\Gamma_1 \to \Gamma_2, \exists xA, \Gamma_3$

then it is replaced by

$\Gamma_1, <pd,A[d/x],>,<p,\forall xA>,\Gamma_2 \to \Gamma_3$ ($\forall$: left')

$\Gamma_1,<p,\forall xA>,\Gamma_2 \to \Gamma_3$

or

$\Gamma_1, \to \Gamma_2,<pd,A[d/x]>,<p,\exists xA>, \Gamma_3$ ($\exists$: right')

$\Gamma_1, \to \Gamma_2,<p,\exists xA>, \Gamma_3$

accordingly with adding t/d to the existing substitution s, where d is a new dummy, and with substituting d for those occurrences of t into "heirs" of the formula A[t/x], which appeared as a result of applying of a replaced rule "inserting" the term t.

2) In all other cases replacement of the rules of G by the rules of mG is evident. (Note that $<,A_1>,...,<,A_m> \to <,B_1>,...,<,B_n>$ is declared as input sequent of D. The rules ($\exists$: left)

and ($\forall$: right) may be considered as those inserting new parameters).

Since D' is a proof tree in the calculus utilizing the classical notion of admissible substitution, then it is clear that the finally generated substitution s is admissible (in the new sense) for a set of all sequences of parameters and dummies from D. Therefore, D is a proof tree for the input sequent $<,A_1>,...,<,A_m> \to <,B_1>,...,<,B_n>$ in mG. *Q.E.D*.

Corollary 1. For any formulas $A_1,..., A_m, B_1,..., B_n$ the formula $(A_1 \wedge ... \wedge A_m) \supset (B_1 \vee ... \vee B_n)$ is valid if and only if there exists a proof tree for the input sequent $<,A_1>,...,<,A_m> \to <,B_1>,...,<,B_n>$ in calculus mG.

Proof.

In accordance with [2] the formula $(A_1 \wedge ... \wedge A_m) \supset (B_1 \vee ... \vee B_n)$ is valid if and only if there exists a proof tree for the input sequent $A_1,...,A_m \to B_1,...,B_n$ in the calculus G. On the basis of the Theorem the latter condition holds true if and only if a proof tree for the input sequent $<,A_1>,...,<,A_m> \to <,B_1>,...,<,B_n>$ can be constructed in calculus mG. *Q.E.D*.

To demonstrate the deduction technique, consider the sequent $A_1,A_2 \to B$ from the above example and establish its provability in calculus G. To do this, construct a proof tree for the input sequent $<,A_1>,<,A_2> \to <,B>$ in calculus mG and use the Theorem.

Applying to the initial sequent only quantifier rules we can receive the following sequent:

$<d_1z_1,R_1(d_1) \vee R2(z_1)>,<,A_1>,<d_2z_2,R_1(z_2) \vee R_2(d_2)>,<,A_2> \to <d_3z_3,R_2(d_3) \vee R_3(x_3)>, <d_3z_3, R_2(d_3) \vee R_3(x_3)>,<,B>$, where $d_1,...,d_4$ are dummies, $z_1,...,z_4$ are parameters.

Now let us apply prorositional rules to the last sequent as long as they are applicable. As a result, we get a deduction tree D. If we generate the substitution $s = \{z_2/d_1, z_3/d_2, c_0/d_3, z_1/d_4\}$ (c_0 is a constant), then we can draw the following conclusions concerning s and D:

1) s is admissible for the set of all sequences of dummies and parameters from D, and

2) every upper sequent from D may be transformed into axioms by applying of s to it.

So, in accordance with the above Theorem the sequent $A_1,A_2 \to B$ is provable in the calculus G. *Q.E.D*.

Some Reconstruction of mG

The formulation of the calculus mG shows that the order of the quantifier rule applications is immaterial. In the calculus mG the quantifier rules are needed to determine a quantifier structure of formulas from the input sequent. This observation gives us possibility to construct a modification mG' of the calculus mG, which contains the so-called doubling rules instead of all the quantifier rules.

Doubling rules:

$\Gamma_1,<pdz_1...z_k,A>,<pd'u_1...u_k,A[d'/d,u_1/z_1,...,u_k/z_k]>, \Gamma_2 \to \Gamma_3$ (D: left)

$\Gamma_1,<pdz_1...z_k,A>,\Gamma_2 \to \Gamma_3$

$\Gamma_1 \to \Gamma_2,<pdz_1...z_k,A>,<pd'u_1...u_k,A[d'/d,u_1/z_1,...,u_k/z_k]>,\Gamma_3$ (D: right)

$\Gamma_1 \to \Gamma_2,<pdz_1...z_k,A>,\Gamma_3$

Here p is a sequence (maybe, empty) of parameters and dummies, the most right variable of which (in non-empty case) is a parameter, d is a dummy, for i=1,...,k z_i is a dummy or parameter, and u_i is a new dummy or a parameter (in accordance with z_i).

In calculus mG' a deduction process starts with an input sequent of the form: $<p_1,M_1>,..., <p_m,M_m> \to <q_1,N_1>,...,<q_n,N_n>$, where $M_1,.., M_m, N_1,..., N_n$ are formulas without quantifiers, and $p_1,...,p_m,q_1,...,q_n$ are sequences of parameters and dummies, which are determined by the formula $(A_1 \wedge ... \wedge A_m) \supset (B_1 \vee ... \vee B_n)$, tested for validability, by the following way:

Let $A'_1, ..., A'_m, B'_1, ..., B'_n$ be some prefix normal forms of the formulas $A_1, ..., A_m, B_1, ..., B_n$, respectively. Then for every i=1,...,m (j=1,...,n) M_i is a matrix of A'_i (N_j is a matrix of B'_j), and p_i (q_j) is obtained by means of replacing in prefix of A'_i (B'_j) of every universal (existential) quantifier by a new dummy and of every existential (universal) quantifier by a new parameter.

All other notions (addmissible substitutions, deduction trees, proof trees, and so on) are the same as in the case of the calculus mG.

Corollary 2. For any formulas $A_1, ..., A_m, B_1, ..., B_n$ the formula $(A_1 \wedge ... \wedge A_m) \supset (B_1 \vee ... \vee B_n)$ is valid if and only if there exists a proof tree for the input sequent $<p_1,M_1>,...,<p_m,M_m> \to <q_1,N_1>,...,<q_n,N_n>$ in the calculus mG'.

Proof.

The formula $(A_1 \wedge ... \wedge A_m) \supset (B_1 \vee ... \vee B_n)$ is valid if and only if $(A'_1 \wedge ... \wedge A'_m) \supset (B'_1 \vee ... \vee B'_n)$ is valid, where $A'_1, ..., A'_m, B'_1, ..., B'_n$ are prefix normal forms of $A_1, ..., A_m, B_1, ..., B_n$, respectively. It is easy to see that a proof tree for the input sequent $<p_1,M_1>,...,<p_m,M_m> \to <q_1,N_1>,...,<q_n,N_n>$ in mG' may be constructed on the basis of

a proof tree for the input sequent <,A'_1>, ... , <,A'_m> $\rightarrow$ <,B'_1>,...,<,B'_n> and vice versa. To complete the proof, use Corollary 1. *Q.E.D*.

Remark. In calculus mG', the quantifier structures of formulas A_1, ..., A_m, B_1, ..., B_n are taken into account by means of sequences p_1, ..., p_m, q_1, ..., q_n. Selection of sequences for determination of quantifier dependencies does not play a principal role and was made for the purpose of visualizing and simplifying of the subject matter. It is possible to construct a (correct and complete) version of calculus mG' using analogs of "schemes" [5] instead of sequences (which also consist of parameters and dummies and reflect the quantifier structures of initial formulas more exactly) and modifying the rules (D: left) and (D: right). Observe also that Herbrand theorem in the form A from [5] may be easily obtained on the basis of a correctness and completeness of the version of calculus mG'.

Conclusion

In this paper the questions of implementation of computer-oriented sequent calculi are not considered because the development of efficient calculi requires optimizing the order of the propositional rule applications and selecting a method for generating of terms which may produce a proof tree. Bypassing details observe that for this purpose the unification algorithm combined with the introduced notion of admissible substitution is suitable. It was the approach that investigated at the level of modern vision [7] of the Evidence Algorithm programme, EA, advances by V. Glushkov. By now, the first version of the System for Automated Deduction, SAD, has been implemented (see Web-site 'http://ea.unicyb.kiev.ua'). This implementation is based on a number of papers devoted to EA and SAD (see, for example, [8-10]).

Bibliography

1. Gentzen G. Untersuchungen uber das Logische Schliessen. Math. Zeit., 39, 1934, 176-210.
2. Gallier J.H. Logic for Computer Science: Foundations of Automatic Theorem Proving. - New York: Harper and Row, Inc., 1986, 513 pp.
3. Kanger S. Simplified Proof Method for Elementary Logic. - Comp. Program. and Form. Sys.: Stud. in Lodic, Amsterdam: North-Holland, Publ. Co., 1963, p. 87-93.
4. Lyaletski A.V. Variant of Herbrand Theorem for Formulas in Prefix Form (in Russian). – Kibernetika, 1981, No 1, p. 112-116.
5. Robinson J. A Machine-Oriented Logic Based on Resolution Principle. - J. of the ACM, 1965, p. 23-41.
6. Herbrand J. Recherches sur la Theorie de la Demonstration. - Travaux de la Societe des Sciences et de Lettres de Varsovie, Class III, Sciences Mathematiques et Physiques, 1930, 33.
7. Kapitonova Y., Letichevsky A., Lyaletski A., and Morokhovets. Algoritm Ochevidnosti - 2000 (project). - Proc. of the 1st Int. Conf. UkrPROG'98, Kiev, Ukraine, 1998, 68-70.
8. Degtyarev A., Lyaletski A., and Morokhovets M. Evidence Algorithm and Sequent Logical Inference Search. - Lecture Notes in Artificial Intelligence, 1705, 1999, 99-117.
9. Degtyarev A., Lyaletski A., and Morokhovets M. On the EA-Style Integrated Processing of Self-Contained Mathematical Texts.-Symbolic Computation and Automated Reasoning (the book devoted to the CALCULEMUS-2000 Symposium: edited by M. Kerber and M. Kohlhase), A K Peters, Ltd, USA, 2001, 126-141.
10. Verchinine K., Degtyarev A., Lyaletsky A., and Paskevich A. System for Automated Deduction (SAD): Linguistic and Deductive Peculiarities. - Advances in Soft Computing (Intelligent Information Systems 2002. (M.A.Klopotek, S.T.Wierzchon, M.Michalewicz (eds)). Physica/Springer Verlag, Heidelberg New York, ISBN 3-7908-1509-8, 2002, 413-422.

Author information

Alexander V. Lyaletski – Faculty of Cybernetics, Kiev National Taras Shevchenko University, Senior Scientist Researcher; 2, Glushkov avenue, building 6, 03022 Kiev, Ukraine; e-mail: lav@unicyb.kiev.ua

Section 7: Philosophy and Methodology of Informatics

ФРАГМЕНТ ОБЩЕЙ СХЕМЫ ИНФОРМАЦИОННО-ЭНЕРГЕТИЧЕСКОЙ МОДЕЛИ ЧЕЛОВЕКА

С.Н.Берестовая, Ю.В.Капитонова

***Аннотация**: В статье предложен фрагмент общей схемы модели человека, основанной на рассмотрении его как кибернетической системы. Исходные положения модели заимствованы из системологии, эзотерической философии и психологии. Отличительной чертой модели является представление в ней человека в виде четырёхуровневой системы, включающей телесный, внутренне-бессознательный, сознательный и внешне-бессознательный уровни. Каждый уровень представлен тремя взаимосвязанными и взаимодействующими составляющими: материальной, информационной и энергетической, которые принимают, хранят, обрабатывают и отторгают, соответственно, материю, информацию и энергию. .*

***Ключевые слова**: материя, информация, энергия, душа человека.*

Введение

Мы знаем, что человек обладает телом, имеющим определённую структуру и определённые функциональные возможности. Он обладает также разумом, волей и чувствами, которые обуславливают его поведение – последовательность предпринимаемых им действий. Человек действует, вообще говоря, целеустремлённо. В каждый момент времени его поведение определяется преследуемой им целью (одной или несколькими) и необходимостью сохранять свою жизнеспособность в процессе её достижения. Являясь частицей человечества, он принимает участие в эволюционном процессе и подчиняется его законам. Поэтому преследуемые им цели обуславливаются как его непосредственными нуждами (например, потребностью в пище, одежде, жилье), так и потребностями эволюции. Человек живёт в определённой среде (части мира, с которой он соприкасается непосредственно). В процессе достижения поставленной цели он, с одной стороны, воздействует на эту среду, изменяя её, а с другой – вынужден сам реагировать на происходящие в ней изменения. Поэтому деятельность человека определяется рамками следующего цикла: исходя из поставленной цели и условий жизнедеятельности, он принимает решение относительно предпринимаемого действия; реализует это действие; анализирует изменения в себе и окружающей среде, произошедшие за время выполнения действия; по результатам анализа, возможно, корректирует цель. Управление этим сложным циклическим процессом осуществляется интеллектом человека [Пиаже, 1969]**.**

Мир познаётся человеком через ощущения, эмоции и чувства, на основании которых под управлением разума и воли формируется последовательность предпринимаемых им действий. С древних времён ведётся спор, являются ли эти качества проявлениями определённой сущности человека, называемой душой (по-гречески психе) [Аристотель, 1976], или они не имеют под собой никакой реальной основы (ортодоксальный материализм). Отрицание души ставит науку психологию в трудное положение, поэтому сейчас по мере всё большего углубления в предмет появилась настойчивая необходимость признания реального существования в человеке систем, функционирование которых обуславливает проявление его разума, воли и чувств [Холодная, 2002; Ильин, 2001]. Выявление этих систем, по-видимому, является первым шагом в направлении научного обоснования существования души и дальнейшего её изучения.

Некоторый свет на существование души можно пролить, если допустить, что ограниченность наших возможностей не позволяет нам ощутить все проявления Универсума (Вселенной), и на этом основании сделать предположение о существовании в нём помимо доступного нам - *проявленного* мира, мира скрытого от нас - непроявленного. Оба мира одинаково реальны и оба представлены определёнными системами, только материал, из которого эти системы построены, различен. О существовании непроявленного мира говорили многие философы, а теперь и физика вплотную подошла к его изучению [Бреннан, 1994]. Но если Универсум состоит из проявленного и непроявленного миров, то и человек как часть Универсума может состоять из систем, принадлежащим обоим мирам: его тело - проявленному, а душа – непроявленному. Непроявленную часть человека в определённых случаях можно наблюдать в виде свечения вокруг его тела. Поэтому в христианской духовной живописи Иисуса Христа и святых изображают с нимбами вокруг головы. Во многих эзотерических учениях – в древнеиндийских ведах, у теософов, розенкрейцеров, индейских целителей, тибетских, индийских и японских буддистов и во многих других описано это свечение и высказаны предположения относительно его природы и структуры.

Стремясь применить в психологии средства кибернетики, мы задались целью разработать общую модель человека, которая включала бы не только его тело, но и душу. В данной статье представлен фрагмент общей схемы этой модели. Исходные её положения заимствованы из психологии , системологии и эзотерической философии. В ней человек рассматривается как кибернетическая система, состоящим из телесного, внутренне-бессознательного, сознательного и внешне-бессознательного планов. Каждый из планов проявляется в трёх аспектах: материальном, информационном и энергетическом. Дана краткая функциональная характеристика систем, принадлежащих рассматриваемым планам.

Основания для построения модели

Психология различает разумный (интеллектуальный) и эмоциональный (аффектный) аспекты феномена человека. Разумный аспект человека, его разум есть согласно структурно-интегральной методологии М.А.Холодной [Холодная, 2002] способность человека логически и творчески мыслить, познавать действительность и оперировать знаниями. Разум проявляется через процесс мышления. В основе разума лежит интеллект как форма организации ментального (разумного) опыта человека. Интеллект представлен в человеке *ментальными структурами* – своеобразными психическими механизмами, которые, реагируя на поступающую в них информацию, порождают информационные объекты (мысли, решения, сообщения и т.п.), составляющие продукт умственной деятельности. В процессе взаимодействия человека с внешним миром, миром других людей и человеческой культурой в целом ментальные структуры, накапливая *ментальный опыт*, видоизменяются, обеспечивая, таким образом, актуализацию субъективного пространства отражения, в рамках которого и строятся конкретные образы конкретных ситуаций. Динамически изменяющийся ментальный опыт человека, актуализирующийся в условиях его познавательного взаимодействия с миром, в котором он существует, составляет его *ментальное пространство*. Участвуя в создании контекста мыслительной деятельности, т.е. окружения, в котором происходит генерация мыслей и принятие решений, ментальное пространство является важной составляющей мыслительного процесса. В рамках ментального пространства происходит мыслительный процесс. И, наконец, *ментальная репрезентация* – это субъективная форма «видения» происходящего, умственная картина события. Наличие ментальной репрезентации является свидетельством существования особого рода психической реальности, которая хотя и инициируется внешним воздействием, но зарождается и обеспечивается в психике человека. Особенности репрезентации происходящего предопределяют характер последующей интеллектуальной деятельности и, в том числе, показатели её эффективности.

С точки зрения кибернетики нам видится ментальная структура как аналог программного обеспечения, ментальное пространство – как аналог базы знаний, а ментальная репрезентация – как аналог результата применения ментальной структуры для конкретизации ментального пространства в конкретной ситуации. Исходными данными для программы, реализующей ментальную структуру, является информация, хранящаяся в ментальном пространстве (накопленные ранее знания и опыт) и информация, которая воспринимается органами чувств человека и характеризует определённую переживаемую им ситуацию. А результатом выполнения такой программы является возникающий в сознании человека образ

переживаемой ситуации – ментальная репрезентация, и, возможно, некоторые изменения ментального пространства.

Эмоциональный аспект человека включает его *ощущения*, *эмоции* и *чувства.* Под *ощущением* понимается информация, поступающая от органов зрения, слуха, вкуса, осязания и обоняния. *Эмоция* – это реакция человека на принятое им сообщение, будь то некоторое ощущение или информация иного рода. Проявление эмоций зависит от испытываемых человеком *чувств* - определённых характеристик его *психофизиологического состояния.* Эмоций связаны с конкретными ситуациями, они проявляются «здесь и теперь» и поэтому кратковременны. Чувства выделяют в окружающей человека среде те объекты, которые имеют для него стабильную мотивационную значимость, и поэтому долговременны. Отличительными признаками чувств являются: отношение к объекту, а не к ситуации, притом к объекту личностно значимому; устойчивость этого отношения; соответствующее отношению поведение.

Эмоции и чувства обычно сопровождаются переживаниями, душевным волнением, изменениями в различных системах человека. Их проявление зависит от его психофизиологического состояния и, как правило, вызывает действия, направленные, с одной стороны, на сохранение целостности организма и обеспечение его жизнедеятельности в данных условиях, а с другой — на продвижение к намеченной цели. Изучая поведение человека, Е.П.Ильин [Ильин, 2001] пришёл к выводу, что человек обладает определённой функциональной системой, поддерживающей и изменяющей его психофизиологическое состояние. По его утверждению эта система является многоуровневой. Она включает психический уровень (в том числе переживания человека), физиологический (центральная нервная система, вегетативная система) и поведенческий (психомоторные реакции, мимика, пантомимика). В любом психофизиологическом состоянии все эти уровни определённым образом представлены, и только по совокупности показателей, отражающих каждый из этих уровней, можно судить о состоянии человека. Аристотель писал, что эмоциональные процессы реализуются совместно душой и телом, а Р.Декарт утверждал, что страсть, возникающая в душе, имеет своего телесного двойника.

Эмоциональное состояние человека характеризуется, в частности, приливом или упадком сил, следовательно, оно имеет непосредственное отношение к протекающим в нём энергетическим процессам. Протекающие в человеке эмоциональные процессы инициируют как моторную, так и интеллектуальную функции человека, связывая все процессы в единый комплекс.

Переходя к рассмотрению модели, напомним, что *системология рассматривает Универсум и все его составляющие как системы*, т.е. совокупности взаимосвязанных элементов, действующих как единое целое [Дружинин, 2001]. Элемент системы, в свою очередь, может быть системой. Любая система имеет три ипостаси: *материальную*, *энергетическую* и *информационную*. Материальная ипостась системы (телесное начало) представлена материалом, из которого построены её элементы; информационная ипостась (разумное начало) представлена информационными потоками, связывающими систему в единое целое и обеспечивающими целенаправленность её функционирования; энергетическая ипостась (действенное начало) представлена энергией, поддерживающей способность её элементов действовать. Всякая система имеет два аспекта: внутренний, касающийся взаимодействия её элементов, и внешний, обеспечивающий взаимодействие системы с внешней средой, т.е. связь системы с другими системами. Связи как внутри системы, так и вне её осуществляются посредством обменов материей, информацией и энергией.

При построении модели нами была использована интересная идея, известная в эзотерической философии под названием закона синархии [Шмаков, 1994], и состоящая в следующем: *Универсум состоит из реальностей и феноменов различного качественного достоинства, сгруппированных в сложную иерархию планов, каждому из которых соответствуют ценности определённого достоинства.* Под планом понимается определённый разрез Универсума. Каждый из планов имеет особую природу и присущие ему законы. Пронизывая друг друга и взаимодействуя между собой, планы образуют единое целое - Универсум, в котором существуют законы, общие для всех планов. Второе утверждение эзотерической философии, которое мы будем использовать при построении модели, гласит: *три ипостаси: мистика, разум и воля, пронизывая Универсум, проявляются на каждом из его планов по-своему*. При этом *мистика* определяется как сущность и содержание бытия, обнаруживающегося в иерархиях Универсума; *разум* - как начало, утверждающее субъективную самобытность и форму всякого единичного бытия, его место в иерархиях Универсума и соподчинённость восходящим планам; а *воля* - как источник и двигатель жизни.

Легко видеть близость положений системологии и эзотерической философии и то, что понятия *мистика*, *разум* и *воля* в эзотерической философии близки понятиям *материя*, *информация* и *энергия* в системологии. Разница состоит лишь в том, что системология использует эти понятия как абстракции, а эзотерическая философия привязывает их к феноменальному и ноуменальному мирам, полагая, что последний относится к области сознания. Мы же, приняв положения, общие для обоих из рассматриваемых направлений, полагаем, что непроявленный мир существует реально, независимо от сознания человека. Таким образом, для нас основополагающими утверждениями являются следующие:

- *Универсум есть иерархическая многоплановая система;*
- *на различных планах материя, информация и энергия могут иметь различные проявления;*
- *одни из планов Универсума относятся к проявленному миру и доступны для наблюдения человеком, а другие к непроявленному и недоступны для него;*
- *существуют законы, общие для проявленного и непроявленного миров.*

Под *материей* мы будем понимать субстрат, который заполняет формы и является основой элементов системы. *Информация* задаёт форму, с помощью которой материя расчленяется на множественность конкретно-эмпирических явлений, организованных в систему. С помощью информации задаются связи между элементами систем, определяющие их целостность, устойчивость, способы функционирования. *Энергия* есть источник жизнедеятельности. Материя, форма и энергия суть объективно существующие самобытные начала Универсума, взаимодополняющие и взаимодействующие друг с другом.

Полагаем, что и информация, и энергия имеют материальный носитель, что позволяет дать им конкретное представление. Путём сопоставления свойств проявленного и непроявленного миров и перенесения (на основании закона синархии) в непроявленный мир тех законов, которые характерны для проявленного мира, мы приходим к выводу, что по аналогии с ролью воды в проявленном мире в непроявленном также должна существовать жизнеобеспечивающая субстанция. Вполне допустимо, что эта субстанция содержит в себе носители информации и энергии, которые назовём, соответственно, *информационнообеспечивающей субстанцией* (*иос*) и *энергообеспечивающей субстанцией* (*эос*).

Структура модели

В рамках предлагаемой модели будем различать *материальное*, *информационное* и *энергетическое Я* человека, т.е. те его ипостаси, которые принимают, хранят, обрабатывают и отторгают, соответственно, материю, информацию и энергию, обеспечивая в единстве своего функционирования жизнедеятельность человека.

Материальное Я человека – это материал, из которого он построен: клетки, органы, некоторые полевые сущности, пока мало изученные наукой, и т.п. К материальному Я относятся также системы регенерации этого материала. Например, обмен веществ, включая системы питания и очистки.

Информационное Я человека - это различные системы приёма, обработки и передачи информации, а также база знаний, в которой, в частности, хранятся определённого вида информационные составляющие, обычно называемые матрицами и моделями. Матрицы описывают структуры отдельных систем человека, а модели могут быть представлены некоторыми автоматами, задающими действия человека в различных ситуациях, определяя тем самым его поведение. Таким образом, матрицы служат информационной основой для обменных и аффектных процессов, а модели поведения - для различных процессов управления, происходящих в человеке, в том числе и поведенческих.

Энергетическое Я человека определяет способность человека действовать и чувствовать. Оно включает механизмы поглощения и выброса энергии, распределения её между различными системами человека. От энергетического состояния зависит самочувствие человека, его воля и способность реализовывать те или иные модели поведения.

Исходя из знаний, накопленных философией, психологией, биологией и медициной, будем полагать, что каждое из рассматриваемых Я принадлежит четырём планам: *телесному*, *внутренне-бессознательному*, *сознательному* и *внешне-бессознательному*, перечисленным здесь в порядке от низшего к высшему. Первый из этих планов относится к проявленному миру, а три другие – к непроявленному. Здесь мы ввели два новые понятия: «внешне-бессознательное» и «внутренне-бессознательное», поскольку обычно используемые в психологии близкие им понятия «подсознание» и «надсознание» не имеют однозначного

определения. Понятие «надсознание» используется редко, а в понятие «подсознание» обычно включается и внутренне-, и внешне-бессознательное.

Функциональное назначение телесного плана состоит в оказании определённого целенаправленного воздействия на физический мир и осуществление контактов с ним. Это воздействие может быть осуществлено телом лишь при условии его пребывания в жизнедеятельном состоянии, поэтому, рассматривая тело, следует различать внешний и внутренний аспекты его деятельности. Внутренне-бессознательный план ответственен за бессознательные формы энергообмена с внешней средой, безусловные и условные рефлексы, выполнение действий, доведенных до автоматизма, поддержание систем тела и самого себя в состоянии устойчивого равновесия и побуждение его к выполнению простейших форм взаимодействия с внешним миром. На сознательном плане происходит сознательный энергообмен с внешней средой, осознание человеком информации, поступающей из внутренне- и внешне-бессознательных планов (его чувств и желаний), а также тех знаний, которыми этот план оперирует. Здесь же выполняется анализ знаний, имеющихся в распоряжении сознания, и синтез новых с использованием аппаратов логического вывода и принятия решений. Через внешне-бессознательный план осуществляется приобщение человека к Универсуму, выполнение высших форм энергообмена с внешней средой и осуществление высшей умственной деятельности – акта творчества.

Данная модель предполагает следующие виды взаимодействий: в рамках одного плана между проявлениями в нём трёх Я; в рамках одной ипостаси между её проявлениями в различных планах; между проявлением Я в каждом из планов и соответствующим планом внешней среды. Так тело, взаимодействует с физическим миром, обменивается с ним на вещественном уровне. На уровне сознания производится вербальный и образный обмены, а на уровне внешне-бессознательного плана используется телепатический обмен информацией.

Итак, мы видим человека, состоящим из четырёх планов, в каждом из которых проявляется три ипостаси (рис.1). Отсюда следует, что человек представим как минимум двенадцатью системами. Для упрощения дальнейшего изложения пронумеруем их как показано на рисунке. Эти системы связаны как по вертикали, образуя определённое Я, так и по горизонтали, образуя определённый план, и могут служить моделями соответствующих подсистем человека.

Мы пока ещё не можем дать точное описание структуры и функционирования выделенных систем. Укажем здесь лишь их назначение и покажем, как перечисленные выше проявления разумного и эмоционального аспектов психики человека могут быть соотнесены с системами его информационного и энергетического Я.

Проявление материального Я на нижнем уровне (система М1) представлено телесной материей, состоящей из атомов и молекул, организованных в клетки, коалиции которых образуют органы и ткани. Его строение и функционирование хорошо известно из медицины. О проявлениях материального Я на более высоких уровнях (системах М2 – М4) пока существуют лишь гипотезы и догадки, на которых мы не будем здесь останавливаться. В пределах тела информация передаётся посредством нервных сигналов и циркулирующими в организме жидкостями (кровь, лимфа, моча). Что является носителем энергии на телесном уровне пока не известно. Существует предположение, что это молекулы воды, находящиеся в особом состоянии, присущем только живым организмам.

Системное рассмотрение информационного и энергетического Я человека является принципиально новым направлением, развивающимся на стыке психологии и кибернетики. Поэтому высказанные здесь утверждения, хотя и являются обоснованными, тем ни менее требуют дальнейших экспериментальных подтверждений.

Информационное Я реализует ментальные структуры человека и организует информационные потоки как внутри человека, так и при его информационном обмене с внешней средой. Система И1, выполняя обработку проходящей через человека иос, обеспечивает целостность и целенаправленность функционирования его тела. Основным органом переработки информации на этом уровне является мозг. Он принимает информацию от внутренне-бессознательного плана и транслирует её в сообщения, рассылаемые соответствующим составляющим человека (органам, мышцам и т.п.) для дальнейшей обработки и исполнения. Через мозг производится и обратная связь: информация от отдельных составляющих передаётся на уровень внутреннего бессознательного. В состав системы И1 входит распределённая база знаний – в каждом функциональном элементе, каждом органе и даже в каждой

клетке имеется её часть, содержащая знания, необходимые для функционирования этой составляющей тела.

Проведя аналогию между естественными (человеком) и искусственными (компьютером) системами переработки информации, приходим к выводу, что, по-видимому, (система И2) подобна встроенному программному обеспечению, внесение изменений в которое со стороны проявлений этого Я на сознательном и внешнебессознательном планах (системы И3 и И4) затруднено. В результате работы системы И3 создаются ментальное пространство и ментальные репрезентации. Система И4, в частности, несёт ответственность за процесс творчества и возникновение инсайтов. Информационные системы высших планов – И2, И3 и И4, являясь формами организации ментального опыта человека, составляют его интеллект. Они реализуют программы восприятия, сенсорного научения, акта понимания, рассуждения и т.д., что в той или иной степени структурирует отношения между средой и организмом. Именно в таком структурированном поведении и состоит информационный (когнитивный) аспект жизнедеятельности человека. Эти программы легко поддаются внесению изменений и благодаря этому служат основой приспособляемости человека к условиям его жизнедеятельности. Таким образом, интеллект человека реализуется его информационным Я, однако, не имея необходимого энергетического уровня, ни одна система не может работать нормально, поэтому и энергетическое, и в какой-то мере и материальное Я принимают участие в интеллектуальной деятельности.

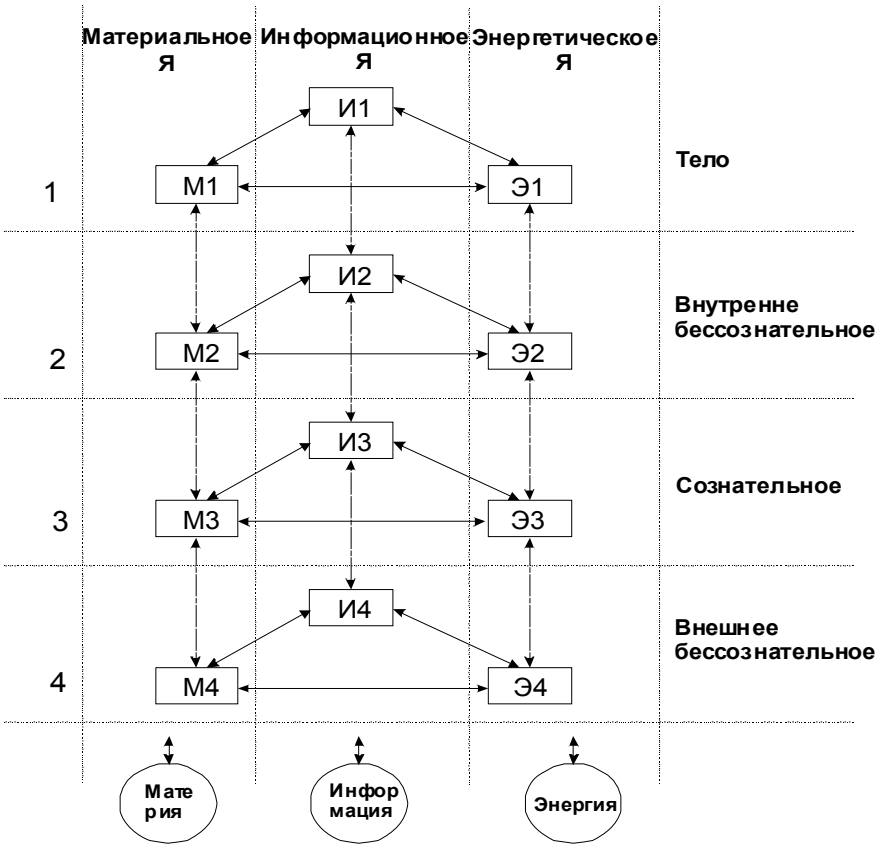


Рис 1. Фрагмент схемы информационно-энергетической модели человека.

Энергетика определяет аффективный аспект человека (его способность действовать и чувствовать) и лежит в основе всякого поведения, а происходящие энергетические обмены со средой по необходимости предполагают существование неких форм или структур, определяющих те возможные пути, по которым происходят эти обмены. В рамках данной модели мы утверждаем, что *энергетическое состояния человека в целом и его отдельных подсистем (органов) проявляется через эмоции и чувства*.

Система Э1 выполняет распределение поступающей в неё потоков эос между всеми подсистемами телесного уровня. Она представлена, согласно воззрениям древнекитайской медицины [Гаваа Лувсан, 1992], сетью, состоящей из перекрещивающихся меридианов (каналов, по которым течёт энергия). Эта сеть обеспечивает энергетическую связь органов тела между собой, а через специальные точки,

называемые биологически активными, осуществляется их связь с покровами тела. Поэтому, воздействуя соответствующим образом на эти точки, можно вносить изменения в работу отдельных органов тела человека. Среди меридианов выделено 12 основных, которые составляют 6 пар по одному ян-меридиану (доставка энергии к органам тела) и инь-меридиану (отток энергии). Каждый ян-меридиан в конечной своей части соединяется с начальной частью соответствующего инь-меридиана, который, в свою очередь, в конечной части смыкается с начальной частью ян-меридиана. Два других меридиана представляют главный поток энергии, циркулирующий вдоль позвоночника и связывающий между собой основные меридианы. Помимо этих четырнадцати каналов существует ещё 8 каналов, которые объединяют точки различных каналов, образуя сеть. Их назначение - перераспределение энергии между меридианами.

Системы Э2 – Э4 ответственны за распределение энергии между системами своего плана, за поддержание необходимого энергетического уровня человека в целом и за проведение энергообменов с внешней средой. Об их структуре известно лишь, что в главный энергетический поток упираются вершинами семь вращающихся конусообразных вихрей, называемых главными чакрами [Бреннан, 1994] и служащих для обмена энергией с окружающей средой (всасывание и выброс эос).

Заключение

Аффективная (связанная с энергетическим состоянием) и когнитивная (связанная с переработкой информации) жизнь являются неразделимыми, оставаясь в то же время различными. Чувства и разум не противостоят одно другому, а являются двумя разновидностями поведения. Аналогично тому, как материальное Я вдыхает воздух и пьёт воду, информационное Я потребляет иос, а энергетическое Я – эос, распределяя их между всеми системами человека. Если энергетическое насыщение человека соответствует его потребностям, то он чувствует себя бодрым и вполне счастливым. Если же такого соответствия нет, то наблюдаются всевозможные расстройства его функционирования и восприятия внешнего мира. В восстановлении энергетического баланса принимают участие, действуя совместно, энергетическое и информационное Я. Благодаря этому, и когнитивный, и аффективный аспекты в действительной жизни проявляются всегда объединёнными, не являясь самостоятельными особенностями жизнедеятельности человека.

Рассмотренный в статье фрагмент схемы информационно-энергетической модели является одной из первых попыток системного взгляда на структуру и функционирование человека. Такое представление является предпосылкой для комплексного рассмотрения человека с кибернетических позиций. Оно, несомненно, послужит основой для многих интересных открытий.

Библиография

[Пиаже, 1969] Ж. Пиаже. Психология интеллекта, из-во «Просвещение», Москва, 1969.

[Аристотель, 1976] Аристотель. Трактат «о душе», сочинения в 4-х томах, т.1, из-во «Мысль», Москва, 1976.

[Холодная, 2002] М.А. Холодная. Психология интеллекта, из-во «Питер», Санкт-Питербург, 2002.

[Ильин, 2001] Е.П. Ильин, Эмоции и чувства, из-во «Питер», Санкт-Питербург, 2001.

[Бреннан, 1994] Б. Э. Бреннан. Руки света, из-во «ОВК», Санкт-Питербург, 1994.

[Дружинин, 1976] В.В. Дружинин, Д.С. Конторов. Проблемы системологии, из-во «Советское радио», Москва, 1976.

[Шмаков, 1994] В. Шмаков, Основы пневматологии (теоретическая механика становления духа), из-во «София», Киев, 1994.

[Гаваа Лувсан, 1992] Гаваа Лувсан, Очерки методов восточной рефлексотерапии, из-во «Здоровье», Киев, 1992.

Сведения об авторах

Светлана Берестовая - НАНУ и-т Кибернетики им. В.М.Глушкова, старший научный сотрудник; пр-т Академика Глушкова, 40, 03680, ГСП, Киев-187, Украина; e-mail: kap@d105.icyb.kiev.ua

Юлия Капитонова - НАНУ и-т Кибернетики им. В.М.Глушкова, заведующая отделом; пр-т Академика Глушкова, 40, 03680, ГСП, Киев-187, Украина; e-mail: kap@d105.icyb.kiev.ua

A KNOWLEDGE-ORIENTED TECHNOLOGY OF SYSTEM-OBJECTIVE ANALYSIS AND MODELLING OF BUSINESS-SYSTEMS

M. Bondarenko, V. Matorin, S. Matorin, N. Slipchenko, E. Solovyova

***Abstract**: A new original method and CASE-tool of system analysis and modelling are represented. They are for the first time consistent with the requirements of object-oriented technology of informational systems design. They essentially facilitate the construction of organisational systems models and increase the quality of the organisational designing and basic technological processes of object application developing.*

***Keywords**: Knowledge, systemology, natural classification, object modelling, conceptual knowledge.*

The civilization sustainable development is based on formation of the informational society as a first stage of the noosphere. At the same time, as the transition to the informational society, as economic activity in it become based on knowledge. This knowledge represents the "**informational resource**". It directly influences at the material factors of progress and ensures the "phase transition of knowledge into a power", i.e. efficiency of business, production and any administrative solutions.

The submission about the tendency of knowledge-oriented development of an alive nature is entered into scientific practice by V.I. Vernadskiy under a title "*the Dan's principle*". The knowledge-oriented development should be considered as the universal tendency enveloping not only biological, but also all other complicated systems. The social (organizational) and information systems also develop ***in a direction of increasing of a knowledge role for their sustainable functioning***.

This tendency is exhibited in the unprecedented growth of knowledge and scientific information; increasing of a role of inclusive, depth knowledge; rapid development of methods and means of knowledge processing, analytical activity, acute need of the appropriate experts, influence of informational resources to all sides of the human activity. The technologies and methods of purchase, extraction, submission, processing of knowledge (knowledge management, knowledge engineering) in substantial aspect also develop in the knowledge-oriented direction (data mining, text mining, knowledge discovery, knowledge mining, object modelling, ontological engineering). In the foreign expert's opinion, the development of these directions is braked by absence of the effective methodologies by the availability of developed technologies.

Accumulated in the given spheres experience and potential even more acutely shows the necessity of the account of depth knowledge, objective factors, system and simultaneously object approach to modelling of complicated systems. Grew the role of the veritable human's resource – "conceptual knowledge", which becomes the core of the informational resources, knowledge bases, and ontology models. In such new spheres of the scientific-practical activity as, for example, business process reengineering, decision support making, object paradigm has appeared the similar necessity, which was already expressed in expediency of the organization's mission definition, context account, systems analysis first of all from the point of view of their functionality, correspondence to requests more high level.

The systemology [1] can become the unique scientific basis of such researches. Systemology is a system approach of the new noospheric stage of science development, which comes to change the differentiation of sciences in analytical paradigm - second in the whole history of science after antique stage.

Systemology allows to work successfully with the complicated systems of the first nature i.e. not human created, and with open systems. At the same time, in difference from other system approaches, is ensured the possibility to consider as a system not only objects, but also classes of objects (systems-classes). The development of the systemology of systems-classes has allowed us to synthesize the system and classification analysis for a solution of problems of conceptual modelling of the low formalized problem areas [2, 3].

Systemology most objectively allows getting the next things for the complicated systems of any nature and with any minuteness:

- to understand the reasons of origin, dynamics of becoming and development;
- to define the influence to other systems;

- to explain the outcomes of adaptation and interaction;
- to predict development in various conditions;
- to make conclusions about necessary measures of stable development;
- to prevent crisis situations and to reduce risks;
- to take into account the main properties and priorities.

Systemology really takes into account system effect, i.e. for the first time considers the system as a qualitatively new essence, instead of reduces it to the sum of component parts. It is ensured owing to the consideration of the system for the first time as:

- integral object, instead of as a set;
- the main properties of a system are explained proceeding from properties of a super system;
- the system is considered as functional object;
- "substation" of a system is taken into account, i.e. "material" from which it is made;
- the shaping and operation of a system of any level is considered from “above” and is determined by the "request" of its super system.

Systemology represents an exposition, oriented on methodological use, of concepts and principles of dialectics, which can be interpreted in terms of any concrete science. Besides, it is a unique system approach, which is agreed at a conceptual level not only with formal logic, object-oriented ideology, but also with a complex of modern scientific-practical disciplines engaging the problems of studying and perfecting of organizational systems (the theory of organization, logistics, and business engineering).

Systemological methods can be applied in cognitive direction of researches, which is major component of knowledge-oriented technologies. It is connected first with the orientation on human is now most necessary for maintenance of harmonic interaction of computer systems with the human.

The development and application of systemology in scientific-educational Knowledge acquisition laboratory (NUL PZ) with the cognitive methods has allowed to decide the fundamental, delivered more than 150 years back, problem of a “natural classification” (NK), to reveal and to formalize its regularities and criterions [3]. NK (systematization) as the ideal of a classification is considered as a privileged system chosen by nature, takes into account the essential properties and relations of objects, the maximum amount of the purposes and can form the basis of the most objective and reality adequate models of knowledge. The features of such classification were studied by many scientists, because it has the greatest value, cognitive and prognostic force and makes a basis of a scientific picture of the universe, but only with the help of systemological approach we succeed in opening its laws. The rules of NK can be taken into account in any problem area and allows creating the effective methods of knowledge systematization and conceptual classification modelling (systemological classification analysis).

The methods of system analysis and instrumental program CASE-tools of their supporting are widely used at the present time for decision of business, administrative and production problems. However, methods and means of traditional system-structural analysis (SADT, DFD, BPwin, etc.), that are used for business-processes modelling, are historically based on procedure-oriented programming paradigm. Therefore the results of their application can’t be immediately used during the developing of object-oriented software.

The most of modern program systems, especially large, at present time are created namely within the frameworks of object-oriented approach. However, the object-oriented analysis (OOA) and language UML are primordial used for software developing. Therefore, they are badly adapted for solution of the problems of business analysis and modelling. At the same time, such problems obligatorily arise, especially during the creation of complex program applications. And what is more, a standard process of object-oriented software developing (Rational Unified Process - RUP) begins with the technological process of business modelling.

A given discrepant situation stipulates the actuality of system and object-oriented methodologies integration. The researches in this direction, carried in NUL PZ, allowed to work up a new original **system-object (systemological) approach and** object-oriented **systemological methodology of analysis and designing (OMSAD)** [4, 5], permitting the marked contradiction. Analytic methods and instrumental means of such approach allow automating the considerable part of analytic work and essentially raising its effectiveness.

Let us consider the basic peculiarities of system-object approach and systemological methodology, and also procedures and possibilities of the new method of system analysis, for the first time consistent with the requirements of object-oriented design.

The traditional system approach (analysis) is peculiar to the *procedure* (functional) system decomposition, and object approach – is peculiar to the *object* system decomposition. At the same time all specialists, as of system analysis, as of object approach consider them as orthogonal. In it's turn, system-object (systemological) approach allows to combine exposure processes of the functional and object structure of analysed system. Thus, the basic peculiarity of the given approach is providing the unity of the decomposition of analysed system, as on functional, as on objective (substantive) sign. This reaches due to the consideration of any system not as a set, but as a *functional «flowing» object [1, 2].* Acknowledgement the status of such an object after the system provides a simultaneous calculation of structural, functional and substantial system existence aspects.

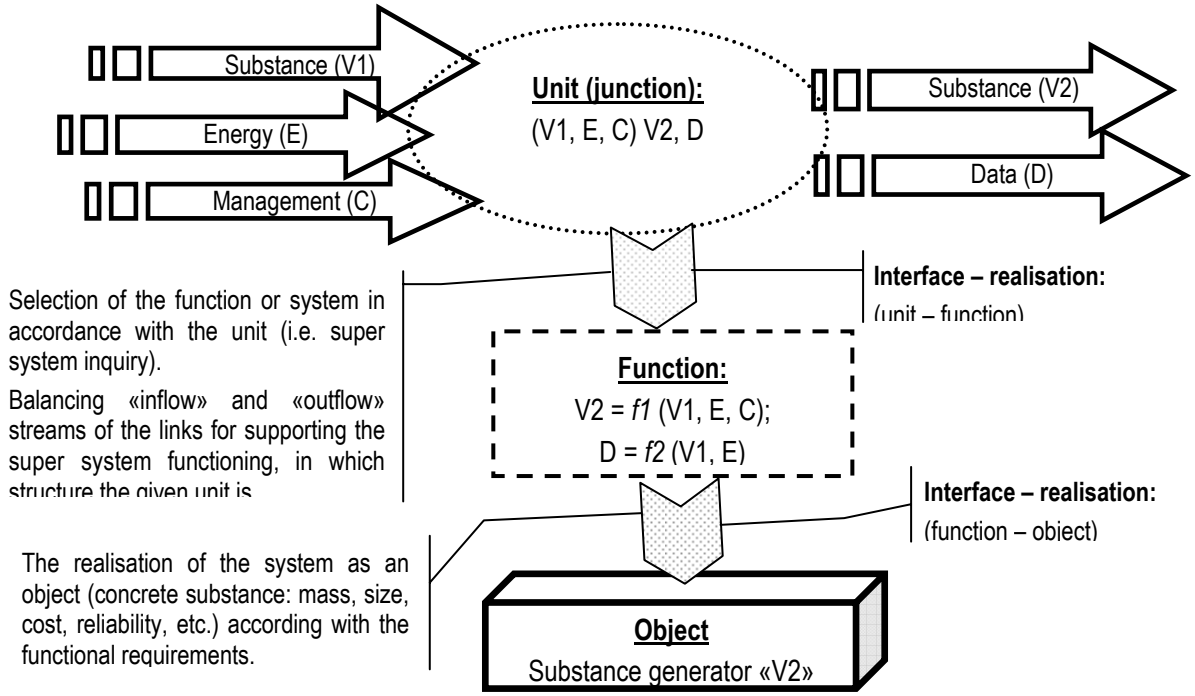


Figure 1. «Unit – Function – Object» approach.

To begin with, any system is a component part of the system structure of higher level (super system), because any system is connected and co-operates with other systems. Herewith any link between systems is the process of mutual *exchange* of elements of definite deep layers of connected systems. Thus, a feature of system is understood as manifestation of it's activity to be included into links, into *exchange flows* with other systems in the super system structure. Consequently, from structural point of view a system is a crossroad of incoming and outgoing links (streams), i.e. unit (node).

Secondly, the functioning (activity, work, behaviour) of any system provides or supports the functioning of the super system, to which this support is necessary. At the same time functioning of the system as a support of the functional ability of the super system consists in the providing of the balance of "influx" and "outflow" on the incoming and outgoing links. Consequently, from the functional point of view the system is a function, which provides a balance of incoming into the system and outgoing from the system streams in accordance to that unit, where this system is in the present moment.

Thirdly, any system is not only a unit and function, but also a substance, which plays a role of definite unit in the structure of the super system and provides its functional balance. Consequently, from the substantial point of view a system is an object, realising a function, set by a unit in the structure of the super system.

Given reasoning allows the representing of any system in appearance of the three elements construction – UFO-element (figure 1) [6], i.e. at the same time:

- as the structural element of the super system – unit, as a crossroad of the relations with the other systems;
- as the functional element, doing a definite role for supporting super system by balancing the given unit – function;
- as the substantial element – object, realising the given function in the appearance of some material formation, having constructive, operational and other characteristics.

The basic peculiarity of the OMSAD methodology is the **formal-semantic adaptive alphabet** of the UFO-elements, and also **categorical principle** that is used during the analysis and designing of systems. Alphabet is a collection of units **(**crossroads of system links), collection of functions, balancing these units, and collection of objects, realising these functions. At the same time, we use facet classification for units collection, defined by the taxonomic categorical classification of the kinds of system's links (figure 2).

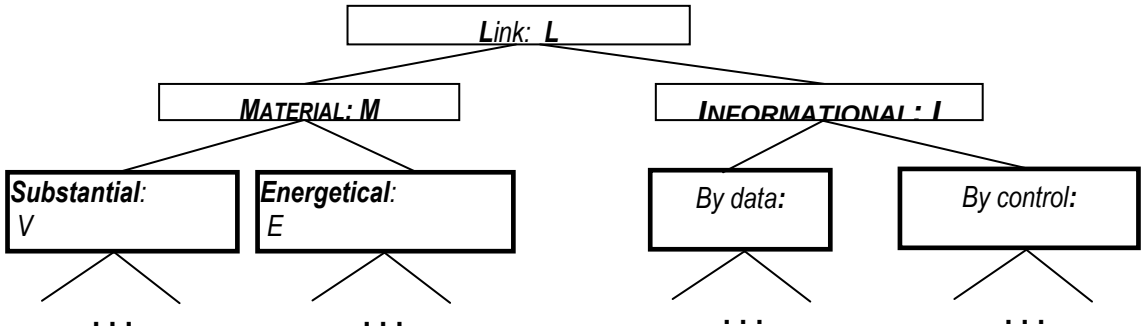


Figure 2. Basic taxonomic classification of system links.

The links classification provides the parametric units classification and constructive determination of symbol semantics of these units. Naturally, the links and units classifications (functions and objects) can be specialised with any degree of accuracy for any concrete domain. The use of classifications for forming the alphabetical collection of the UFO-elements and the possibility of their specialisation turns this collection into the formal-semantic adaptive alphabet.

Parametrical taxonomic classification of UFO-elements represents a classification, in which the objects are systematized depending on functions, which they are realizing, function - depending on what units they are balancing, and the units are determined by that, what crossroad of link they are. This is a conceptual model of application domain in the terms of knots, functions and objects witch plays a role of a " categorical" grid, through which the analyst looks at the domain. The specialization of such a categorical classification model should be carried out in the correspondence with the recommendations of the *systemological classification analysis* offered in the work [3] and directed on the construction of the classifications, witch takes into account the properties and regularities of the natural classification. In the correspondence with these recommendations during the construction and specialization of the classification the good, natural classification will be obtained, if the definite sequence of operations mentioned below is observed.

Any units got by combining of the links from the classification can be considered as alphabetical elements. However with practical point of view it's expediently to consider not all of the possible combinations, but only such ones, which corresponds to the actual physical laws (for example, to preservation laws). Point is that energy does not exist without any material bearer, information does not exist without any material bearer and administration does not exist without any data transmission. This leads to the relatively small number of variants on the level of the links of the base classification (figure 2 In the given tables we use brief markings for data on the material (VD = D) and power (VED = G) bearers, and for control data on the material (VDC = C) and power (VEDC = Q) bearers. This determines that, at present time, only paper (D and C) and electronic (G and Q) information bearers have the wide diffusion.

The use of alphabet (libraries) of UFO-elements allows formulating the combining rules of these elements naturally following from the systemological approach, for constructing UFO-configurations. We offer to call these rules the **rules of system decomposition:**

1. The rule of association: elements should be linked together according to the qualitative and quantitative characteristics of links inherent in them;

2. The rule of balance: during the connection of elements to each other (according rule 1) the qualitative end quantitative balance of the inflow and outflow of input and output functional links must be observed at units of the system structure;

3. The rule of realisation: during the connection of elements to each other (according to the rules 1 and 2) the interfaces accordance and the accordance of the objective and functional characteristics must be observed;

4. The rule of closeness: internal (supporting) links (streams) of elements in system must be reserved.

Offered alphabet and named rules forms **a formal-semantic normative system** of systemological analysis and modelling, formalising by the pattern theory of Grenander funds.

Table 1

		Entries:					Exits:		
		Production	**Providing**			**Administra-tor**	**Product**	**Informational**	**Wastes**
			Substantial	**Energetic**	**Informational**				
Business system		V, E, D(G), C(Q)	V	E	D(G)	C(Q)	V, E, D(G), C(Q)	D(G)	V, E
Production	Substance	V_{in}	V_{pr}	E_{pr}	$D(G)_{pr}$	$C(Q)_{pr}$	V_{out}	$D(G)_{out}$	V_{wst}, E_{wst}
	Energy	V_{in}, E_{in}	–''–	–''–	–''–	–''–	E_{out}	–''–	–''–
	Information	$D(G)_{in}$	–''–	–''–	–''–	–''–	$D(G)_{out}$ $C(Q)_{out}$	–''–	V_{wst}
Transport	Substance	V	–''–	–''–	–''–	–''–	V	–''–	–''–
	Energy	E	–''–	–''–	–''–	–''–	E	–''–	V_{wst}, E_{wst}
	Information	D(G), C(Q)	–''–	–''–	–''–	–''–	D(G), C(Q)	–''–	V_{wst}
Allocation	Substance	V	–''–	–''–	–''–	–''–	V	–''–	–''–
	Information	D(G)	–''–	–''–	–''–	–''–	D(G)	–''–	–''–

Besides, OMSAD methodology is using a categorical principle during the construction of models. This principle postulates the necessity of prior assignment (definition) of the synthesised (designed) systems from categorical classification of such systems. Named principle, in fact, in the obvious form fixes the common sense used in the practical analytic work. The point is that decomposition (analysis) and aggregation (synthesis) procedures can be successfully realised only in that case, if they are directed by the final result. During the realisation of synthesis operation, it's necessary to know something at least about the kind of synthesised system, and during the realisation of analysis operation it's necessary to know something about the types of the parts, on which analysed system can be decomposed. Thus, mentioned above alphabet is a realisation of the categorical principle from the point of view of system analysis procedure. For solving the modelling and designing problems of organisational systems OMSAD methodology is using the systems categories, represented in the table 1.

The experience of the practical using of systemological methodology showed, that context model of any organisation (business-systems), and also of any of it's subdivision, can be represented as unit from the table 1. For example, workshop, model and tool shops, naturally, are represented as the systems of the material production. Department of main constructor, office of the production technical training, economic planning department, accountancy, labour and salary department, marketing department, etc. is represented as the systems of information production. Department of technical control, department of main mechanic, department of main technologist, provision department, sales department, department of technical documentation, etc. is represented as distributive systems.

Formal-semantic normative analysis and modelling system and business-system categories can be considered, in particular, as the development and addition, for example, of the popular technology SADT. As is well known, this technology grants the formal universal possibilities on constructing of the functional business-processes structures. However it doesn't take into account the semantics of the domain and does not give to the analytic the information about the concrete interactions between the analysed systems and their possible filling. So, a context modelling and systems decomposition with the SADT funds are heuristic procedures and don't have any support with the proper CASE-tools (for example BPwin) on the substantial level.

Systemological approach «Unit – Function – Object» and OMSAD methodology allowed to work up a new method of business-systems analysis and modelling (**UFO-analysis**), which allows to adapt it's funds to the concrete data domain, i.e. to take into account it's semantics [6, 7]. Besides, the systems representation with this method as configurations of UFO-elements provides the concordance of the derivable models with the requirements of object-oriented design.

Briefly, the following main steps can represent UFO-analysis procedures:

- Revealing units links in the structure of the modelling (designing) system based on functional links of the system as a whole, defining by the customer or solving problem;
- Revealing of functionality supporting (providing, balancing) found units;
- Determining objects, corresponding to the revealed functionality, i.e. those realising it.

The specific peculiarity of this analysis method is providing automation possibility of these steps. Automation reaches due to using of formally semantic adaptive alphabet. At the same time it's necessary to take into account prepared beforehand classification of UFO-elements (UFO-library), which contains suitable elements for the given problem (data domain). In this case the first step may be identified with the system analysis stage, the second - with it's design, and the third - with it's implementation.

For automated application of UFO-analysis method we developed a program complex «UFO-toolkit», which is the CASE-tool, using knowledge base of the special configuration for providing a component approach to modelling, using semantics of domain and intellectualisation of interaction with user [7]. The tool is intended for object and simulation models construction of complex dynamic (organisational) systems. It has the following features:

- noticeably reduces the designing labour-output ratio owing to intensified automation of analytic activity;
- increases the objectivity of the analysis and the adequacy of modelling;
- automates a models creation process, through the use of ready (alphabetical, library) functional objects, presented in the knowledge base of the Tool in the form of UFO-elements;
- provides «intelligent» interaction with user, making familiar the ready component (UFO-elements).

At the same time if alphabetical elements appears to be program objects, realised as ready classes, then we can talk about UFO-analysis as a part of component technologies and business-objects technology CORBA (Business Object Facility – BOF). In the last situation the program CASE-tool, automating UFO-analysis procedures, can function within the frameworks of component business-objects architecture (Business Object Component Architecture – BOCA). At the same time, it will carry out an organiser (Framework) role, which, integrating business-objects into the functioning system, gives them the working places for realising their tasks. If we consider the alphabetical elements as the engineering elements, then UFO-analysis will be confirmed with the CALS-technology.

Thus, UFO-analysis method represents the development and concrete definition of the OMSAD methodology. It allows to use the formalised rules of revealing classes and objects of application domain during the OOA process and to realise the system analysis of the events of different nature, considering them as functional flowing objects. Consequently, UFO-analysis can be considered as the method of system-objective analysis and modelling.

On the whole the considered method and Tool (UFO-technology) provide to user:

- objectivity of analysis and synthesis procedures of organisational systems;
- economy of the man-hours of analysis and modelling, because these procedures both comes to the construction of only one model;

- simplicity and availability of the business-processes analysis and modelling by specialists without special training;
- uniform presentation of external and internal models of the business-system, described by the same modelling language;
- facilitate of models adaptation to the concrete domain (taking into account the semantics domain);
- the possibility of creation and use the libraries (repository) of the model components for different application fields.

Besides, they have the following merits:

- provide the concordance of the system analysis results with requirements of object-oriented design, previously considering as orthogonal;
- provide a possibility of immediate use of the system analysis results during the creation of object-oriented software;
- raises the level of formality and automations of the modelling and analysis procedures;
- guarantee a concordance of all system characteristics due to unification of the different system consideration aspects in one model;
- provide facilitate of the construction of visual models of different abstraction level, representing at the same time a functional and objective structure of system;
- provide a possibility of the modelling of functional system characters, not having a mathematical interpretation or interpreted by any mathematical means, and also simulation of system functioning without any special modelling algorithm.

Represented analysis and modelling technology is used for correcting information-analytic business-systems accompaniment and provides an essential rise of the effectiveness of their activity. Developed method and program tool essentially facilitate the construction of organisational systems models and increase the quality of the organisational design and initial technological processes of developing object applications.

Bibliography

1. Melnikov G.P. Systemology and linguistic aspects of cybernetics. New York, Paris, Monreal, Tokyo, Melbourne. Gordon and Breach.- 1988.- 440 p.
2. Bondarenko M.F., Matorin S. I., Solovyova E. A. Analysis of systemological tools for conceptual modelling of application fields // automatic document and mathematical linguistics. New York: Allerton Press Inc. 1997. V. 30, No. 2. P. 33-45.
3. Соловьёва Е.А. Естественная классификация: системологические основания. Харьков: ХТУРЕ, 1999.222 С.
4. Matorin S.I. A new technology of system-object analysis and its application for business-systems modelling // EEJET № 1(1) 2003 P 15-20.
5. Matorin S.I. A New Method of Systemological Analysis Co-ordinated with the Object-Oriented Design Procedure. I // Cybernetics and Systems Analysis. Plenum Publishing Corporation, 2001. V. 37, No. 4. P. 562-572.
6. Matorin S.I. A New Method of Systemological Analysis Co-ordinated with the Object-Oriented Design Procedure. II // Cybernetics and Systems Analysis. Plenum Publishing Corporation, 2002. V. 38, No. 1. P. 100-109.
7. Маторин С.И. Анализ и моделирование бизнес-систем: системологическая объектно-ориентированная технология / Под ред. М.Ф. Бондаренко; Предисловие Э.В. Попова. (ISBN 966-659-049-2) Харьков: ХТУРЭ, 2002. 322 с.

Author information

Michael Bondarenko – Rector of Kharkov National University of Radioelectronics, Professor
Vasiliy Matorin - Kharkov National University of Radioelectronics, student
Sergei Matorin - Kharkov National University of Radioelectronics, doctorant
Nikolay Slipchenko - KHNURE, chief of scientific department, professor
Ekatherina Solovyova – KHNURE, head of Knowledge Acquisition Laboratory, professor, 14 Lenin Avenew, Kharkov, Ukraine 61166, nulpz@kture.kharkov.ua

THE INFLUENCE OF COMPUTER ENVIRONMENT ON THE INDIVIDUAL'S PERSONALITY

V. Kolomeyko

***Abstract**: In article the problems of mutual adapting of the humans and computer environment are reviewed. Features of image-intuitive and physical-mathematical modes of perception and thinking are investigated. The problems of choice of means and methods of the differential education the computerized society are considered.*

***Keywords**: image-intuitive modes of perception, physical-mathematical modes of perception, differential education.*

Introduction

At present the human society and computerized environment are going through the complicated process of mutual adaptation. The new approaches to the solution of the tasks and problems, making scientific researches, studying, many more. Some of the processes are completely new for the humanity that is why they demand undivided attention of the researchers and society on the whole. It is important not only to study tendencies and to analyze possible consequences but to work out the means and methods of purposeful control of them. The purpose of this work is to single out some key-problems, to set possible ways of their solution.

Image-Intuitive and Physical-Mathematical Modes of Perception and Thought

The existed modes of studying and description of objects, processes and phenomena may be conditionally united into two groups, named further as physical-mathematical and image-intuitive modes of perception and thinking. The basis of physical-mathematical modes is formed by traditional for physics and mathematics approaches and methods, including formalization, Laplacian determinism [1]. The basis of image-intuitive mode is orientation to the image thought, intuition, subconscious, collegial and other informal methods of making decisions [2].

It's natural that such division is rather schematic without clearly marked borders. It is easy to explain because very often in practice a kind of symbiosis of these two approaches and the attempts of strict division may seem to be controversial, artificial, incorrect.

For humanity image-intuitive mode is more usual, because it has appeared incomparably earlier historically. So there is no wonder that small children in their attempt to know the word use image-intuitive mode only. Even their abstractions and generalizations are based on the combination of imagines and inherent intuitive methods of perception. But the traditional views towards science and education prefer physical-mathematical mode. Such prevalence very often reveal itself in the primary school where to children with imaginary, intuitive thought physical-mathematical mode of cognition of the world is instilled. But only for minority it is useful.

Rather typical test, connected with description of the leaf's flight from the tree. The child of pre-school age describes such flight rather adequately. But as for the senior students very often they try to do this with the help of physic's textbook not taking into the consideration the striking difference between experimental and theoretical results.

It is well known [3] that every individual has his particular features of thought, perception, cognition of the world. For one people the complicated mathematical formulas bring the feeling of harmony and the understanding of the heart of the matter. For another they are not more than dry notes which are hard to understand, and the real harmony and understanding of heart of things is contained in quite another (the classical example — Mozart and Saliery). Wrong selected way of teaching contradicts with the innate world of the man, with his subconscious, mode of thought, becomes the obstacle for the development of the individual and society in whole.

The methods of differential education, taking into consideration the abilities, inclinations, the character of pupil were developed and used by different pedagogues [3, 4]. Not going into details of these methods, we should

admit that they are oriented into pre-computer world reality. That is why they demand reconsideration and further development.

The Formation Individual's Personality in His Intensive Coordination with Computer Environment

One of the most important factors making the influence on the forming of the modes of perception, thought and cognition of the world is global computerizing. A human being as one of the biological spices has got virtual intellectual assistants, advisors, prompters, persons to talk to, friends, the performers of the everyday work, which take a lot of duties, functions unknown before. The role of influence of these assistants is not always obvious and with only one meaning. But they exist, develop quickly enter different areas of human activity. So it is time to take into consideration that not simply an individual with his abilities, priorities, modes of perception and making decisions lives, studies, works, but the individual who has the virtual assistant.

More over, the development of the computer environment has reached such stage when the virtual assistant performs a lot of tasks and functions more effectively than the man. That is why to our mind the teaching of one or another specialist is advisable to consider as a process of creating and teaching the one human-computer team, in which a man adapts to his virtual assistant (computer), and computer adjusts to the one particular man. As the tasks, possibilities, needs of the man change during his whole life so the process of mutual adaptation of each of the participators from every man-computer team may last during whole life.

This problem is complicated, closely connected with the help of return contact with a lot of other different tasks and problems. That is why we should solve this problem from different sides. Because psychology, pedagogy, computer sciences, and fundamental preparation and plus a lot of professional human cultural knowledge abilities and skills are important here.

But we should start with the analysis of the influence of those principally new factors which computer brings into the creative, innate world of the man. To our mind the most important are as follows:

1. The high level of lability, disposition, adaptation of the computer according to the tastes, needs, and demands of different users.

2. A friendly programming environment which allows even unprepared users to establish effective cooperation with computer.

3. The elements of game which exist in the associating with the computer. Clearly expressed play origin arıses the level of motıvatıon of the users even in process of makıng everyday and burdensome work.

4. The possibilities of practical use of the most difficult methods of mathematics other sciences and subjects by non-specialists.

5. Completely new possibilities of communication and multimedia.

6. The presence of the means and methods of fixation and notation of events, actions, decisions making possible to simplify the following analysis and adaptation of the system.

7. The possibility to cooperate not with the real objects and phenomena but with their virtual copies and worlds [5].

This and other factors connected with the rapid development of the computer environment make great influence as well as on separate individuals and on the whole society. Some of these factors are considered lower.

The Development of Means of the Differentiatial Education in the Computerized Society

Taking into the consideration upper said, we keep to the mind that for every student should be selected individual combinations and proportions of different methods of teaching. The most important factors in such selection must be:

- the particular features of the mind, perception and the character of the individual;
- the particular features of the future specialization and the work of the individual.

This supposes the necessity:

- the differentiation of the profession, the areas of activity, the teaching methods, in dependence from correlation different modes of thought and making decisions;
- effective distribution of functions, tasks, roles among the society, individual and computer as it applies to one or another situation, profession, area of activities.

To put it differently there are a lot of obvious and non-obvious dependencies and connections between the choice of teaching methods and the distribution of functions in the computerized society in general and human-computer systems in particular.

We do not claim to determine what subjects and in what amount should be studied by one or other student. Our purpose is to investigate the mutual influence of the tendencies in the computer environment development and teaching methods. It is important to understand some things, to investigate general laws based on the simply and impressive examples.

As one of such examples we have chosen the tasks connected with the architect education. The foundation for such selection was follows. For the first, the profession of architect supposes the possession of both imagine-intuitive and physical-mathematical thought. For the second, the profession of architect is in the great demand according to creators of the systems of computer-aided design / computer-aided design manufacturing (CAD/CAM) that let us say about the experience and tendencies. For the third, strongly pronounced architect-builder inclinations reveal itself in a rather early childhood, that gives the reasons to consider this profession and part of the conclusions and recommendations rather broadly.

It is obvious that the education of the architect, who designs the objects with the help of computer is one thing. And the training of the architect who has to create computer programs of the calculation constructions and expenses is quite another thing. The first one based on the image-intuitive way which is inherent to the creative process. That is why he must “feel” mathematics, thermal physics, economics more than to know them (it is supposed that this general technical subjects are based upon corresponding CAD/CAM or the calculation group). The second one must know deeply all basic subjects because without it he can not “apply” them in his programs.

Both of these two architects must professionally use proper CAD/CAM and effectively make every possible constructions and calculations. The analysis of the present tendencies with the high level of confidence make possible prognosis of two things. For the first that CAD/CAM will become the main working instruments of the architects. For the second that for the majority of the architect-students the most effective methods of studying of the courses and the units of the general technical subjects will be admitted the methods, based upon the practical calculations, constructions, projects with the help of the intellectual instruments.

For example the methods of studying the courses oriented toward the first student to our mind must include the possibilities of making the virtual experiments. When a student chooses one or another construction, the results of this experiment are immediately shown: the gradients of thermal fields, voltage waveforms, the grades of heat irradiation. When student changes the construction the appropriate modifications of fields and loads appear on the monitor.

It is quite possible that at the beginning the student will advance to the acceptable decision with the help of the cut-and-try method. But the experience of use of game situation shows that very quickly appear a kind of understanding of the laws based possibly on the feeling of innate harmony. The chaotic selection of the variants is changed into the sensible use of the method of the successive approximations. A man begins to “feel” the problem, studies to solve it effectively.

At the set example with two students we have considered some extreme cases: image-intuitive thought or physical-mathematical. In ordinary life such cases take place not very often. The intermediate variants are more typical.

For example, in many professions connected with the design the best results achieve the specialists who effectively combine image-intuitive and physical-mathematical modes of thought. A lot of experienced specialists have learned on practice how to set themselves into one of another mode of thought, to select important proportions of these modes according to the character of the tasks which are being solved and works which are being made.

As people are very different and solve very different tasks so one of them are at a lack of the image-intuitive thought and other are in need of physical-mathematical thought. This lack could be compensated at the expense of choice of appropriate profession and work. But it is so not always. That is why a special attention is paid to the

creation of intellectual assistants, which are able to compensate the lack of knowledge, skills and abilities of the individual.

We should stress that computer is not very reliable from the point of view of increasing the potential of the image-intuitive mode of thought. The main achievements in this field are qualitative visualization, wide multimedia possibilities. The rest is much worse, not talking about the intuition. This is significant argument in favor of abrupt increasing the pedagogues' attention to the development of the image-intuitive mode of thought. Because the lack of development of physical-mathematical mode of thought computers compensate more successfully.

One of the most effective methods of purposeful development of one or another ways of perception and thought is the practical cooperation with the appropriate developing and teaching computer programs. These programs usually contain a big number of specially selected game situations, which substantially increase the level of user's motivation and simplify the teaching. There are a lot of such programs and games. The spectrum of them is rather wide too. Some of them are considered to be strictly specialized other is used for wide application. There is positioning in fields of activities, age, interests. For the last years developing computer games appeared which are ranged as the games for children of 3 to 5 years (we should notice that they are played even by 2 years old children with the help of adults).

The analysis of results of practical use of such games persuade us in advisability of fundamental expansion in this direction. The basing on the game approaches gives powerful stimulus for the development and teaching the child. At the same time the investigations directed on the revealing and use of another positive treats and particular features of child's perception are necessary. Without it the game may bring not only the benefit but harm. Besides this, it is very hard to develop the scenarios of games, create appropriate interesting understandable and at the same time useful play situations, without studying the particular features of perception.

In this case it is advisable to our mind to base on such instincts of creation that the majority of people has. The child willingly builds something from the sand, bricks and details of constructor. That is why at the beginning the substitution of bricks, details of constructors by their virtual analogs may be possible. Then goes the rapid increase of the number and types of virtual details. A new effect will take place in that case if the computer begin to form a kind of evaluation figures of constructions created by the child. For example if it is said about the building of virtual house so such figures may be the durability of the house, the value of the flats, their temperature conditions, the isolation of sounds... In another words the things, which the child knows and understands.

Practice shows that when a child plays so-called strategic and emulation computer games he begins to understand very quickly that a lot of things in the world are interconnected. The improvement of one thing leads to the deterioration of another. It leads to multi-criterion optimization by the certain examples. In games, which are oriented towards the children of pre-school age, should not be a lot of criterions (1 - 2). In primary school classes the number of optimization criterions may run up to 3. As regards senior school, a lot of things here depend upon the definite subjects and teaching purposes.

It is obvious that the extension of range of the studied problems leads to increase of optimization criterions. The same example with the construction of buildings supposes the studying and analysis of many factors (including visual, social, ecological). That is why the number of optimization criterions may reach 10 and 20. As for the means and methods of education in the senior classes the author, taking into consideration the limits of article, can repeat the same arguments as in the example with architects. Adding that many things are still not clear, require studying, development and approbation.

Studying the questions of differentiation of education with the help of computers we should point to the possible risks and negative moments. The most obvious those demerits and difficulties which are on the top. For the first these are the factors of negative influence of computer environment upon the children, additional stratification of society, expenditure and difficulties connected with the considerable reorganization of educational process, the individual choice of educational programs, teaching forms, means and methods. Besides the differentiation of education at the primary school classes supposes the necessity of making appropriate decisions about career-guidance because such decisions are always connected with multitudinous risks.

As the humanity has started the path of rapid computerization and there are no other real alternatives so the demerits and possible risks only stress the main thesis of article: the problems of global computerization are complex, require special attention of investigators and society on the whole.

Bibliography

1. Князева Е.Н. Одиссея научного разума. Синергетическое видение научного прогресса. – М.: ИФРАН, 1995.-228с.
2. Коломейко В.В. Методологические аспекты построения человеко-машинных систем поддержки принятия решений // Искусственный интеллект. – 2002. - № 3. – С.101 - 106.
3. Гипенрейтер Ю.Б., Романова В.Я. Психология индивидуальных различий. – М.: ЧеРо, 2002. – 776с.
4. Дифференциация в начальном звене. / Под ред. Ю.З.Гильбуха. Психолого-педагогические основы дифференциального обучения в общеобразовательной школе. – К.: Перспектива, 1996. – 53 с.
5. Коломейко В.В. Проблемы использования неформальной информации в задачах моделирования // УСиМ. – 2001. - № 6. – С.17-24.

Author information

Vladimir Kolomeyko - V.M.Glushkov Institute of Cybernetics, Prospect Acad. Glushkova, 40, 03680, Kiev-187, Ukraine. Tel: (380+44) 234 58 51 e-mail: vvklm@ukrpost.net

К СЕМИОТИКЕ НООСФЕРЫ

В. Лозовский

***Abstract**: Civilization has brought us into the noosphere world. Besides physical, around (and inside of) us exist and function also mental and cultural entities. It is impossible to perform now knowledge acquisition, knowledge base creation and organizational systems management without adequate consideration of object's noospheric statuses. We tried here to clarify basic viewpoints concerning this issue, hoping that elaboration of common methodological foundations of semiotic modeling will be useful for developers and also for users of new generation automation systems.*

***Keywords**: noosphere, antroposphere, semiotics, modeling, ноосфера, семиотика, моделирование.*

Введение

Самыми сложными, в плане управления, являются организационные системы. Для подобных систем характерно почти полное отсутствие надежных математических моделей. На ведущие позиции выдвигаются требования эргономики, естественности моделей с точки зрения «человеческого» восприятия, легкости их сопровождения и перманентной модификации. Становится ясно, что в этих условиях не обойтись без методов прикладной семиотики, позволяющих сблизить представления человека с компьютерными реализациями. С другой стороны, необходимо более строго определить статусы объектов, с которыми приходится сталкиваться в организационных системах, с учетом того, что фактически перед нами задачи управления в условиях ноосферы.

Представленный в данной работе материал отражает начальные этапы наших исследований в данной области. Отсюда – повышенное внимание философско-методологическим вопросам, формулировке основных определений. Мы поставлены в условия, когда необходимо разобраться в новой сложной системе понятий и постараться выработать некий общий язык, адекватный новому типу задач.

Ноосферный подход к управлению оргсистемами

Ограниченность возможностей строгих моделей для мягких предметных областей является следствием принципиального несоответствия формального языка реальному знанию, которым обладают специалисты в этих предметных областях. Это знание характеризуется высокой антропоморфностью: неопределенность, неоднозначность, неточность, большая доля качественных и лингвистических характеристик и оценок (большой, значительный, опасный, перспективный, устойчивый, обоснованный, «оказывает отрицательное влияние на ...», «нежелательные последствия» и т.п.). Присутствует и, так называемый, ***«человеческий фактор»*** (субъективность, эмоциональность, усталость, невнимательность, нелогичность, лень, сложная и динамичная структура межличностных отношений – симпатии, антипатии, доверие, предрассудки, обида, благодарность, месть, семейные отношения, «чувство справедливости», «чувство долга», «чувство глубокого внутреннего удовлетворения», способность работать в команде, наличие собственного механизма целеполагания и реализации целей). Существующие методы моделирования практически полностью игнорируют человеческий фактор.

Научно обоснованный подход к задачам управления в подобных областях требует, прежде всего, выработки парадигмы представления знаний о реальном мире, а также методов работы с этим знанием: накопление, интеграция, верификация, телекоммуникация. Мир вошел в эпоху информационного общества. Однако практически нет результатов по разработке языковых средств адекватных поставленной задаче. В свое время к ней пытался подойти Ньютон, но он настолько опередил свое время, что от этой задачи ему пришлось отказаться. Серьезные шаги в этом направлении были сделаны В.И.Вернадским [Вернадский, 1943], который и предложил использовать термин «ноосфера» для полного, корректного и эффективного учета всего объема знаний, накапливаемых обществом, но в его время не была проработана прикладная семиотика, подходы к представлению знаний.

Учение о ноосфере с позиций эпистемологии

Процесс образования ноосферы метафорично и романтично пояснил М.Пришвин [Пришвин, 1957]: «Где-то на невидимом небе всего человечества бродят скопленные всеми веками великие мысли, бросают тени, как облака, а по этим теням особенно чуткие люди догадываются и понимают сами мысли».

Термин «ноосфера» использовался и до Вернадского – философом-идеалистом Леруа и антропологом-католиком Тейлором де Шарденом. По мнению последнего, ноосфера на Земле возникла под воздействием божественного разума, духа. По большому счету, нам сейчас почти безразлично, как это произошло – в результате естественной эволюции или по воле Господа. Но для исследователя гипотеза божественного происхождения человека и ноосферы ставит крест на пути познания – в прямом и переносном смысле.

Мы будем исходить из того, что история Земли включает следующие последовательные стадии:

- ***космическая*** – определяющая процессы трансформации Вселенной, которые привели к образованию нашей галактики, солнечной системы и Земли;
- ***геологическая*** – неорганические процессы, происходящие в твердой, жидкой и газообразной средах (при этом космические факторы продолжают оказывать существенное влияние на процессы, происходящие на Земле);
- ***биологическая*** – появление жизни на Земле и возникновение специфических процессов, связанных с жизнедеятельностью живых организмов; человечество на ранних стадиях своего существования;
- ***цивилизация*** – дальнейшее развитие человеческой деятельности: освоение природных ресурсов, вольное или невольное вмешательство в геологические и биологические процессы, состояние окружающей среды; с течением времени, цивилизация стала оказывать все более заметное влияние на состояние Мира и на процессы его трансформации;
- ***ноосферная стадия*** – эпоха, при которой резко возросший объем знаний человечества о мире, сопровождающийся еще более увеличившимся и заметным в планетарном масштабе влиянием на окружающую среду, требует и делает возможным кардинальное изменение стратегий человеческого поведения для всесторонне гармоничного развития цивилизации с учетом всех планетарных факторов.

Дадим свое рабочее определение понятию ***«ноосферная стадия развития человечества»***. Это – такая стадия, при которой выполняются следующие условия – ***«нооусловия»*** - определяющие переход от концепции «разумного человека» к концепции «разумного общества».

1. Научный, технический, социальный, этический, моральный и культурный потенциал человечества достиг такого критического уровня, при котором возможно выполнение всех нижеследующих условий.
2. В качестве цели развития человечества принимается максимальное удовлетворение запросов каждой личности в той мере и постольку, поскольку это не противоречит удовлетворению запросов других личностей.
3. В качестве ведущего морального принципа устанавливается принцип обратимости действия и реакции (относись к другим так же, как ты хочешь, чтоб относились к тебе).
4. *Формируется глобальная распределенная база данных и знаний (ГБ) справочного, теоретического и прикладного характера о свойствах, состоянии Мира (физического и ментального) и о процессах в нем происходящих; по сути, - модель Мира.*
5. Возникает развитая информационно-телекоммуникационная инфраструктура, которая обеспечивает эффективную связь членов общества между собой и доступ к ГБ для ее формирования, контроля, сопровождения и использования при принятии решений на всех уровнях жизнедеятельности общества.
6. Юридические, социальные, административные, воспитательные, образовательные и политические функции общества ориентируются на наиболее эффективное выполнение указанных выше условий.

Выполнение п. 5, благодаря Интернету, можно считать принципиально решенным вопросом. В то же время, готовность общества к реализации п. 4 сдерживается трудностями смены парадигмы общественного мышления, культурной ориентации, недоработанностью вопросов интеграции знания и методов моделирования для «мягких» ПОб, каковыми и являются системы гуманитарных знаний и управление организационными системами.

Прикладная семиотика как парадигма информационно-ноосферных моделей

Семиотический подход к проблеме моделирования позволяет сохранить высокую степень корреляции между реальным миром и моделью, что помогает строить модель и сопровождать ее в процессе всего жизненного цикла разработки.

Семиотика – наука, исследующая структуру, свойства и динамику формальных символьных систем в их отношении с физическим и культуральным миром с позиций познающей мир системы, построение и использование символьных моделей действительности, правила интерпретации, манипуляции и поведения таких моделей.

Истоки семиотики прослеживаются, начиная от работ античных философов. Более четкое изложение базовых понятий было выполнено Г.Фреге [Frege, 1892]. Понятие сигнификации было введено А.Черчем [Church, 1956]. Фундаментальное исследование свойств базовой триады семиотики – вещей, свойств и отношений – было предпринято А.И.Уемовым [Уемов, 1963]. Э.Ф.Скороходько [Скороходько, 1962] был в числе первых, предложивших строить компьютерные модели в терминах отношений – «гх-кодов». Это направление было в дальнейшем развито в работах по ситуационному управлению [Поспелов, 1975, 1986]. Формулировка и уточнение понятий прикладной семиотики при использовании сетевых и реляционных систем представления знаний содержится в работах [Лозовский, 1979 – 1999].

Прикладная семиотика отличается тем, что в ее рамках речь идет не о чисто формальных, а о вполне реальных символьных моделях, реализуемых средствами вычислительной техники для целей моделирования и управления реальными прикладными объектами и системами.

Компьютерные семиотические модели обладают изначальной семантической глубиной, позволяя на их основе выполнять процедуры ассимиляции нового знания, проверки на непротиворечивость и полноту, строить планы целенаправленного поведения, осуществлять содержательный контроль поведения систем, диагностирование, принятие решений по управлению, взаимодействовать с системой моделирования в терминах привычных человеку поведенческих понятий: действия, состояния, цели, сценарии, процедуры, функции. Этих свойств лишены скомпилированные программные решения на языках низкого уров-

ня. Семантика на этом уровне уже потеряна, присутствуют лишь рецепты выполнения тех или иных действий – без объяснений мотивировок. Остановимся на основных определениях.

Универсум – множество всех сущностей (объектов) Вселенной (ноосферы).

Сущность, или объект – нечто, выделенное наблюдателем из универсума с позиций определенных прагматических соображений, на что направлено наше внимание, о чем мы говорим или думаем, на что ссылаемся тем или иным образом, например, «Черное море», «находиться западнее», «христианство», «гравитация», «социалистическое соревнование», «устойчивое развитие экономики региона»…

Любое исследование, анализ, моделирование предполагает вычленение из всего универсума некоторого его подмножества, которое мы будем называть ***предметной областью*** (ПОб). Обычно ПОб включает в себя как непосредственно объект исследования, моделирования и/или управления, так и его окружение, называемое средой. Их взаимодействие и взаимовлияние должно рассматриваться в комплексе.

Еще один деликатный момент связан с отношением между моделью и моделируемым объектом. Включать ли модель в Поб? И, наконец, - как быть с исследователем, который изучает данную ПОб, строит модель, делает на ее основании какие-то выводы и затем использует полученное знание в непосредственной работе с прототипным объектом? Модель, в свою очередь, тоже может быть непосредственно связана с объектом моделирования, получая с него данные непосредственных измерений. Результаты же моделирования, в свою очередь, могут прямо или косвенно оказывать воздействие на объект моделирования, на исследователя и на пользователя-управленца.

Предметная область (Поб) – множество всех сущностей (объектов), имеющих существенное отношение к решаемой задаче анализа, моделирования или управления.

Давая столь широкое определение понятию ПОб, мы отдаем себе отчет в том, что при этом мы грешим против классических философских канонов. В рамки нашего рассмотрения оказываются включенными как «материальные», так и «идеальные» сущности. С точки зрения «чистой» науки, это тяжелый грех, концептуальный винигрет. Ответов здесь два.

Во-первых, жизнь нас подталкивает к принятию ноосферного подхода к управлению ОС, к существованию в современном мире. Поскольку в мире фактически на равных правах действуют как материальные, так и «идеальные» объекты, наша картина мира должна это максимально адекватно учитывать.

Во-вторых, то, что было не под силу классической философии, становится возможным сейчас, на базе семиотики и исследований в области представления знаний. Требуется смена парадигмы в условиях, когда есть необходимые для этого средства.

ПОб включает в себя множество сущностей - d-сущностей – domain entities. В том же смысле мы будем употребяль термин «объект» - нечто, на что обращено наше внимание, что является предметом нашего рассмотрения. Подчеркнем, что под «сущностью», «объектом» мы понимаем не только предметы, а все то, что фигурирует в нашей картине мира, пусть даже нечто совершенно эфемерное.

С точки зрения ноосферного подхода в нашей интерпретации, сущности могут иметь один из трех ноосферных статусов: быть физическими, ментальными или культуральными.

Физические сущности (P-сущности) – это объекты реально существующие, или бесспорно существовавшие в физическом мире. Вопрос установления реальности существования – стародавний философский вопрос, одно из главных полей сражений между материализмом и идеализмом. Несмотря на наличие ошибочных, спорных и пограничных ситуаций (флогистон, НЛО, телекинез, всемогущий Господь), мы вынуждены использовать подобное нестрогое определение, поскольку оно все-таки вносит упорядоченность в терминологию и используемые понятия и позволяет специалистам сблизить языки описания ПОб, лучше понимать друг друга. Как это ни дискомфортно, но мы должны смириться с положением, что полностью формальным этот процесс быть не может. Просто, мы должны иметь возможность при изменении когнитивных позиций разработчиков и пользователей системы моделирования, оперативно внести необходимые исправления и уточнения в модель.

Ментальные сущности (m-объекты) – сущности, формируемые в мыслящих системах интеллектуальных субъектов (ИС) – мысли, идеи, представления - и используемые для целей познания, анализа, моделирования, прогнозирования, планирования.

Речь может идти о биологических субъектах (люди, животные), либо о субъектах искусственного происхождения – роботы, экспертные системы, системы поддержки принятия управленческих решений, системы

искусственного интеллекта. Для нас не имеет значения физическая природа этих сущностей и протекающих процессов. Важны их когнитивные – семиотические и функциональные свойства, динамика взаимодействия с другими m-объектами данного субъекта и окружающим миром, процессы преобразования информации. Помимо «образов» конкретных p-объектов (Украина, А.С.Пушкин, памятник Дюку и т.д.), в мозгу ИС могут существовать чисто абстрактные (математические теории, концепция жилища, «что такое хорошо» и т.п.), а также вымышленные, фольклорные, мифологические, религиозные понятия, которые, в принципе, не имеют p-прототипов.

Мы приходим к заключению, что m-объекты – специфические сущности, которые существуют в особой среде: в мыслящем мозгу ИС. С одной стороны, они безусловно объективны, поскольку формируются в материальной среде и с помощью материальных, вполне реальных биологических, химических, электрические процессов, имеют материальные носители, а с другой, они субъективны, присущи конкретному единичному индивидууму и с большим трудом поддаются экспликации как самим индивидуумом, так и сторонним наблюдателем – собеседником, психоаналитиком.

Интеллектуальный субъект (ИС) – это субъект, который:

- обладает уникальной способностью: строить в своем мозгу модель окружающей среды и себя в ней; эта модель должна адекватно, ***объективно отражать*** существенные свойства соответствующих p-объектов;
- обладает способностью к целенаправленной деятельности;
- способен строить и модифицировать свою систему представления знаний, включающую абстрактные m-сущности, пользоваться аксиоматическим методом, приемами работы с определениями, приемами дедукции, индукции, абдукции, обладать алгоритмичностью мышления;
- обладает способностью к обучению, самообучению, планированию своей деятельности, реализации планов, навыками активного взаимодействия с окружающей средой в процессе получения необходимой информации и реализации целенаправленных действий.

Объективность отражения устанавливается апостериори – по результатам целенаправленной деятельности данного ИС: если она успешна, можно утверждать, что модель отражала действительность объективно.

Цивилизация, интеграция знаний и умений, общение ИС, воспитание и обучение молодежи были бы невозможны, если бы результаты интеллектуального творчества оставались на уровне m-объектов отдельных индивидуумов. Мы с необходимостью приходим к выводу о существовании третьего типа сущностей – культуральных объектов (c-объектов).

Культуральные сущности (C-сущности) – это объекты «культурального мира», созданного эволюцией и цивилизацией в рамках человеческих сообществ, в самом общем представлении: науки, искусства, обычаи, религии, ритуалы, законы, нормативы, планы, «человеческие» отношения (страх, ненависть, любовь, доверие, восхищение, удивление, ирония и т.д.).

C-объекты – очень непростые сущности, это – кентавры, представляющие собой синтез объективных и субъективных характеристик. Они представляют собой, как бы, объективизированный аналог m-объектов – поскольку они отчуждены от конкретных субъектов. C-объекты не существуют «в природе». Они «материализуются» лишь в процессе интерпретации некоторой ИС.

Различие между m- и c-сущностями можно пояснить на таких примерах. Замысел «Войны и мира», появившийся у Льва Толстого – это система m-сущностей. Созданный роман, опубликованный, прочитанный и понятый различными людьми – это уже c-сущность, Она может быть предметом обсуждений; существуют более или менее канонические толкования образов этого романа, ими можно пользоваться в метафорическом смысле, и подобные метафоры образованными людьми будут восприниматься сходным образом.

Иногда возникает соблазн вообще отказаться от этой гносеологической категории, и считать c-объекты обычными p-объектами, с которыми у них действительно много общего. И очень часто знания о каком-либо c-объекте гораздо более полны и систематизированы, чем об ином p-объекте. Так, Красная Шапочка или Кот Матроскин из Простоквашино для нас гораздо более знакомы и понятны, чем гидрогеологические особенности северо-западного шельфа Черного моря.

Часто c-объекты имеют материальные корреляты в виде p-объектов. Некоторые c-объекты обходятся и вообще без материальных коррелятов – например, устное народное творчество, «неписаные законы».

Для иллюстрации «реальности» с-объектов обратимся к юридическому примеру: «фактический брак» - совместное проживание и ведение хозяйства. Если есть основания признать конкретных граждан находящимися в состоянии фактического брака, то возникают определенные реальные права и обязанности их по отношению к совместно нажитому имуществу, детям и т.д.

ИС сталкивается с окружающим миром двояко: непосредственно (ваш автомобиль налетел – упаси Господи! – на придорожный столб; при этом ни у вас, ни у прибывших сотрудников дорожно-патрульной службы не возникает сомнения в реальности автомобиля, столба, аварии, приведшей к материальному ущербу) и когнитивно – путем отображения в мозгу ИС. При этом m-объекты, формируемые в интеллектуальных сферах участников и свидетелей, конечно, разные. Процесс составления протокола аварии имеет своей целью выработку единой, официальной точки зрения на происшествие, но и после его составления, и даже подписания всеми участниками, уверенности в том, что синтез единого с-объекта произошел, нет. Вот мы и пришли к идеалистической, практически солипсистской трактовке – что для каждого из участников рассматриваемого инцидента «объективно» существуют только его собственные (субъективные!) ощущения и ментальные конструкции, m-объекты. Конечно, все это до определенной границы, за которой виновнику придется расплачиваться за нанесенный ущерб и ремонт техники вполне конкретными, «материальными» деньгами.

Граница между физическим и культуральным миром возникает, в первую очередь, в результате естественного различия между p-объектами и их культуральными коррелятами, субъективными образами и интерпретациями, которые создают познающие мир системы. Рассмотрим объект «Проект строительства канала» (4, рис. 1). Как p-объект (на схеме он не выделен) – это множество печатных листов, содержащее определенный текст. С соответствующим с-объектом связывается определенная семантическая и прагматическая интерпретация. Информация, которую извлекает каждый субъект при знакомстве с этим с-объектом, различна и определяется профессиональной ориентацией, теоретическим багажом и опытом практической работы читающего, его целями, привходящими обстоятельствами – в том числе, личными отношениями с начальством, смежными организациями и т.п. Сложность и неоднозначность оценок культуральных объектов – одна из основных трудностей создания систем управления ОС.

Еще одна методологическая трудность заключается в том, что с-объекты могут играть роль p-объектов, объектов, находящихся вне познающей мир системы. Так, преподаватель целенаправленно формирует у учащихся систему определенных с-объектов – знания и умения в определенных ПОб. Последующие экзамены как раз и направлены на выявление «объективной» картины в представлениях учащихся.

Одна из задач, перманентно решаемых человеческим обществом – стремление к экспликации m-объектов. Наука, ее передний фронт работает, в большой степени, в области субъективных, m-категорий. С течением времени, вырабатывается определенная парадигма, создается научная школа. При этом первоначально субъективные концепции (m-концепции), становясь достоянием множества специалистов, поддерживаемые научными публикациями, обменом информацией на конференциях, конкретными прикладными результатами, начинают постепенно приобретать все большую объективность, становясь культуральными объектами, т.е. объектами, созданными (эксплицированными) в процессе человеческого мышления. Наконец, новая теория, проработанная отрасль науки входят в учебные пособия и начинают преподаваться широкому кругу студентов, приобретая «как бы» объективность. Вот это «как бы» становится очевидным, когда разными научными школами в рамках разных парадигм создаются разные эмпирические теории. Некоторые с-объекты, в конце концов, могут найти физическое воплощение в виде реальных p-объектов.

Поясним приведенные выше соображения. Вернемся к конкретному схематическому примеру – фрагменту ноосферно-семиотической объектной структуры ПОб (рис. 1). Предметная область представляет собой часть универсума. До возникновения ноосферы - в отсутствие ИС - универсум состоял бы исключительно из p-объектов естественного происхождения (Земля, гравитация, электромагнитные волны, химические элементы, минералы, животный, растительный мир и т.д.). ИС воспринимает их с помощью своих органов чувств, измерительных приборов, дополняя складывающуюся картину своими представлениями, соображениями, гипотезами, логическими умозаключениями.

Данное представление поддерживает материалистическую гипотезу о первичности материи. Существование m-сущностей возможно лишь в сознании мыслящих субъектов, являющихся их носителями, которые должны были существовать *до того*, как возникла мысль. Мысль без материального носителя – бессодержательное понятие. Невозможно создать картину при отсутствии кисти, красок и холста. Формирование

соответствующих c-сущностей происходит в социуме на базе определенных m-сущностей. Происходит и обратное: трансформация c-сущностей в m-сущности отдельных индивидуумов в процессе обучения и воспитания. Эти процессы неотделимы от появления языков человеческого общения, выполняющих двойственную функцию – мыслительную и коммуникативную.

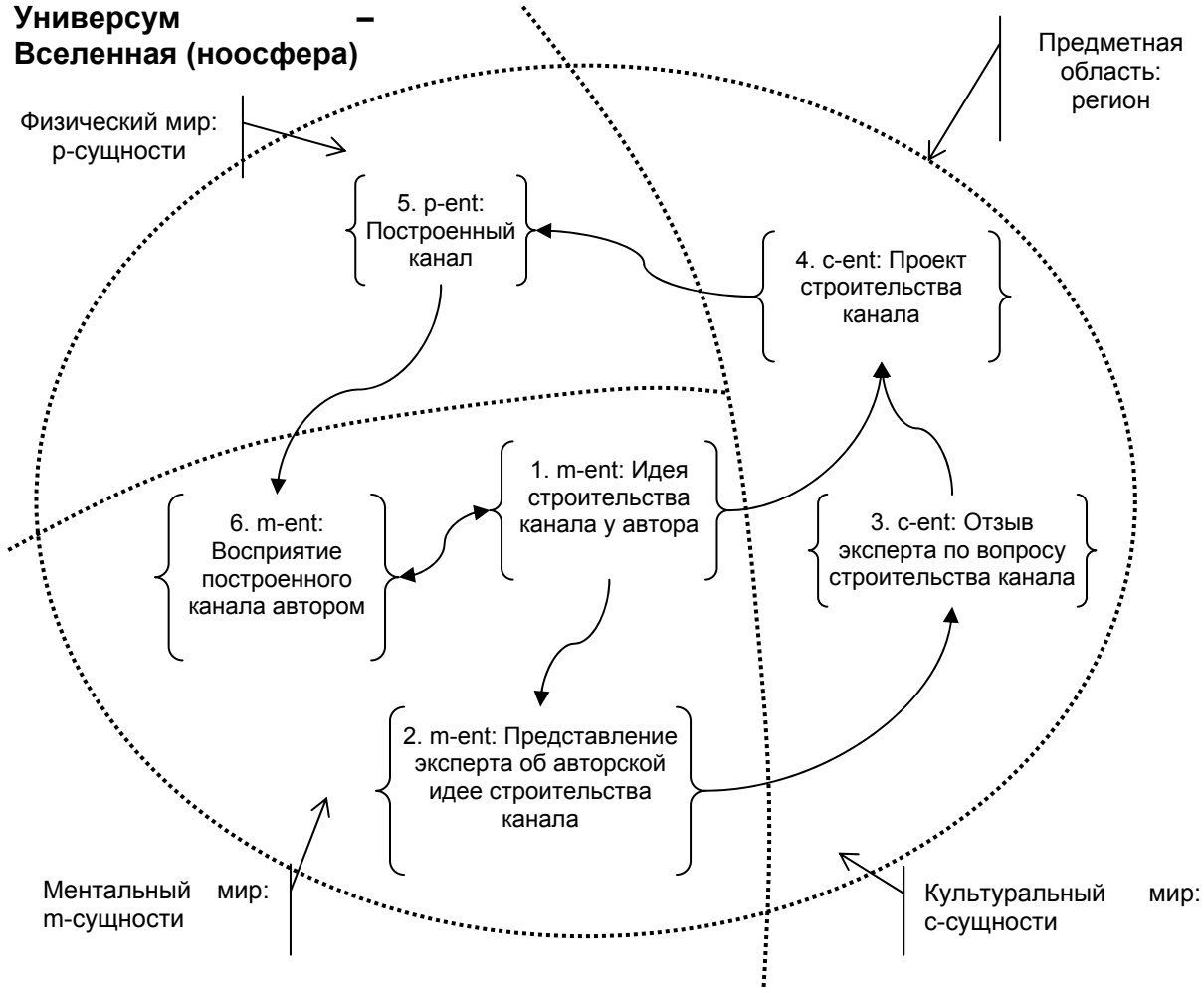


Рис. 1 Пример ноосферно-семиотической структуры предметной области

Предположим, что у некоего субъекта – автора – возникла идея строительства канала (1), рис. 1. Это – субъективная сущность (m-ent), доступная восприятию исключительно ее автора. Далее, с ней знакомится некий эксперт, который создает в своем мозгу представление об идее (1): сущность (2). Так же, как и исходный авторский замысел, это полностью субъективное представление: нет никаких гарантий относительно степени адэкватности его предмету исследования, формирование его подвержено многим неконтролируемым и неявным факторам (образовательный ценз, опыт практической работы, информированность, принадлежность к той или иной профессиональной школе, парадигматика и т.п.).

Результат работы эксперта – создание культурального объекта – c-ent (3) – отзыва на авторское предложение о строительстве канала. С этого момента мнение эксперта становится доступным другим субъектам – состоялся переход сущности из субъективной ментальной категории в культуральную – объективизация, а точнее, экспликация ментального представления.

Предположим, что идея строительства рассматриваемого канала встретила поддержку и появляется новый культуральный объект – «Проект строительства канала» (4). Он возникает в результате труда многих людей, на основе анализа многих документов, справочных материалов и т.п..

Предположим, далее, что, наконец, спроектированный канал построен физически. На нашей схеме это зафиксировано с помощью p-сущности (5). Автор первоначальной идеи, ознакомившись с ее фактическим воплощением, может сформировать в своем сознании представление об ее реализации (6).

К чему же мы пришли в результате этого рассуждения?

Во-первых. Нам следует утвердиться в мысли, насколько тесно переплетаются в информационном, семиотическом плане три выделенных нами мира – физический, ментальный и культуральный.

Во-вторых. Мы убеждаемся в том, что целенаправленное поведение человеческого сообщества в наше время должно рассматриваться с позиций ноосферных представлений о естественном триединстве рассматриваемых нами семиотических категорий. Этот вывод идет несколько вразрез с традиционным «чистым» научным подходом, предостерегающим нас от смешения материального и «идеального», выстраивающим непроницаемый водораздел между этими мирами.

В третьих. Создавая современные компьютерные экспертные и управляющие системы, претендующие на высокий уровень адэкватности реалиям и компетентности при принятии решений, необходимо научиться корректно работать с объектами всех трех миров. В частности, если традиционно в науках по управлению рассматривался процесс однократного отображения реалий физического мира компьютерной системой представления знаний, то корректная ноосферная трактовка процесса взаимодействия ИС приводит к необходимости правильного учета феномена рефлексии m-сущностей. Грубо говоря – взаимодействуя с субъектами, наделенными интеллектом, нам приходится учитывать, как ими воспринимаются наши идеи и представления. Ошибка выполнения уже этого – первого уровня рефлексии – может привести к нежелательным результатам. И это в то время как для достижения успеха коммуникативных актов и, в конечном счете, целенаправленной деятельности с учетом человеческого фактора, необходимо учитывать не менее двух, а иногда и больше уровней рефлексии [Лефевр, 1973].

В процессе эволюции, роль человеческого фактора все росла, и в настоящее время, особенно там, где речь идет об управлении оргсистемами, не может игнорироваться. Поэтому необходимо корректно учитывать особенности «идеальных» - ментальных и культуральных объектов, эффективно координируя эти процессы с деятельностью, связанной с p-сущностями. Все это – в условиях, характеризуемых множеством усложняющих факторов: неполная, недостоверная, ненадежная, неточная информация, ошибки всех возможных родов, работа с неполностью наблюдаемыми и неполностью контролируемыми системами.

Как m-, так и c-объекты обладают разной степенью объективности, точности отражения, моделирования свойств прототипного d-объекта и располагаются в разных точках оси абстракции, берущей начало в ПОб. Ее второй конец соответствует чисто абстрактным объектам. В качестве прототипного объекта может выступать любой (p-, m- или c-) объект ПОб.

Мы приходим к выводу, что в любых системах организационного управления, равно как и в любых системах представления знаний, необходимо четко различать предметный статус сущностей, с которыми мы имеем дело.

Выводы

Автоматизация управления в оргсистемах потребовала пересмотра и уточнения проблем семиотики, подходов к построению моделей предметной области в рамках ноосферных представлений. Целью данной работы было уточнение статусов сущностей, с которыми приходится иметь при этом дело. Предложено разделение предметной области на три составляющие – физический, ментальный и культуральный мир с последующим явным учетом этих статусов при построении семиотических моделей. Дальнейшие исследования в этом направлении будут направлены на разработку методов описания целенаправленной деятельности по управлению оргсистемами с учетом конкретного ноосферного статуса рассматриваемых сущностей.

Литература

[Вернадский, 1943] Владимир Иванович Вернадский, Несколько слов о ноосфере, 1943-1944, Из Архива В. И. Вернадского: http://vernadsky.lib.ru/ .

[Пришвин, 1957] М.Пришвин, Собрание сочинений, т. 5, М., 1957, 683 с.

[Kuhn, 1962] Thomas Kuhn, The Structure of Scientific Revolutions, University of Chicago Press, 1962

[Уемов, 1963] А.И.Уемов, Вещи, свойства и отношения, АН ССР, Институт философии, Издательство АН ССР, М., 1963, 184 с.

[Frege, 1892] Gottlob Frege, Über Sinn und Bedeutung, «Zeitschrift für Philosophie und philosophische Kritik» No. 100, 1892, pp. 25-50

[Church, 1956] Alonzo Church, Introduction to Mathematical Logic, Vol. 1, Princeton University Press, 1956

[Скороходько, 1962] Э.Ф.Скороходько, Информационный язык для технических наук, «Математическая и структурная лингвистика», № 1, Киев, ИК АН УССР, 1962, 50 с.

[Поспелов, 1975] Д.А.Поспелов, Большие системы. Ситуационное управление, М., «Знание», 1975, 64 с.

[Поспелов, 1986] Д.А.Поспелов, Ситуационное управление, Теория и практика, Сер.: Проблемы искусственного интеллекта, «Наука», ГРФМН, М., 1986, 284 с.

[Лозовский, 1979] В.С.Лозовский, Ситуационная и дефиниторная семантика системы представления знаний, "Кибернетика", No. 2, 1979, стр. 98 – 101

[Лозовский, 1990] В.С.Лозовский, Сетевые модели, разд. 1.3 в кн.: Искусственный интеллект, в 3-х кн., Кн. 2: Модели и методы. Справочник, п/р Д.А.Поспелова, М., "Радио и связь", 1990, стр. 28 - 49

[Lozovskiy, 1998] Lozovskiy Vitaliy (UA): Common Sense Semiotics, Conference Proceedings: Knowledge-Based Software Engineering (Smolenice, Slovakia), P. Navrat and H. Ueno (Eds.), IOS Press, Amsterdam, Berlin, Oxford, Tokyo, Washington, DC, ISSN: 0922-6389, ISBN: 90 5199 417 6 (IOS Press), 1998, pp.232-240

[Lozovskiy, 1999] Vitaliy Lozovskiy, On the Road to Parasemiotics, ASC/IC'99 - Труды 4-го международного семинара по прикладной семиотике, семиотическому и интеллектуальному управлению, Институт программных систем РАН, Российский университет дружбы народов, Российская ассоциация искусственного интеллекта, Москва, октябрь 1999, ISBN5-89574-064-2, с. 158-166

[Лефевр, 1973] В.Лефевр, Конфликтующие структуры, «Советское Радио», М., 1973.

Author information

Виталий Лозовский – Институт проблем рынка и экономико-экологических исследований НАН Украины, с.н.с., Французский бульвар, 29, Одесса, 65044, Украина; e-mail: loz@loz.intes.odessa.ua

ANALOGICAL REASONING TECHNIQUES IN INTELLIGENT COUNTERTERRORISM SYSTEMS

A.B. Markman, D.A. Rachkovskij, I.S. Misuno, E.G. Revunova

Abstract*: The paper develops a set of ideas and techniques supporting analogical reasoning throughout the life-cycle of terrorist acts. Implementation of these ideas and techniques can enhance the intellectual level of computer-based systems for a wide range of personnel dealing with various aspects of the problem of terrorism and its effects. The method combines techniques of structure-sensitive distributed representations in the framework of Associative-Projective Neural Networks, and knowledge obtained through the progress in analogical reasoning, in particular the Structure Mapping Theory. The impact of these analogical reasoning tools on the efforts to minimize the effects of terrorist acts on civilian population is expected by facilitating knowledge acquisition and formation of terrorism-related knowledge bases, as well as supporting the processes of analysis, decision making, and reasoning with those knowledge bases for users at various levels of expertise before, during, and after terrorist acts.*

Keywords*: analogical reasoning, structure-mapping theory, associative-projective neural networks, knowledge bases, terrorism, terrorist acts, antiterrorism, counterterrorism, SMT, SME, APNN*

Introduction

Since September 2001, the world has awakened to a new danger - the threat of international terrorism. Combating terrorism has become a high priority in international cooperation. A key component of counterterrorism measures is going to be the creation of tools that can supplement human reasoning to handle the vast amount of data that is being generated by counterterrorism measures. Technologies from computer science will play an important role in endeavors ranging from intelligence, prevention, preparedness planning, response training - to crisis management, reaction, mitigation, and recovery (e.g. [SAIC]). Computer-based systems to assist expert and nonexpert personnel to reason about terrorist activities (henceforth referred to as terracts) will benefit from further development of such systems. The key to these enhancements, as we propose, is the integration of existing tools and the creation of new computer-based counterterrorism systems using analogical reasoning techniques. The resulting analogical processing tools working with data and knowledge bases (KB) of experience accumulated from known terracts will provide new capabilities to counterterrorism systems. This approach can be used to support reasoning, prediction, assessment, analysis, decision-making, problem solving, planning, response actions throughout life-cycle of terrorist incidents.

Ultimately, such systems should be able to reason and learn from examples in the area of terrorism and other complex real-wold domains in the same manner as humans, who are capable of accumulating knowledge and experience by assimilating examples, working through problems, and reusing examples to solve new problems. However, humans have well-known limitations of their abilities. In particular, human memory tends to preserve the gist of prior situations, without necessarily providing access to detailed descriptions of past situations. In addition, humans often do not work efficiently under stress and time pressure. Therefore, it would be useful to have a system that supplements human reasoning and addresses people's shortcomings. Furthermore, such a system must be compatible with human reasoning processes in order to facilitate integration of a computer system's recommendations with ideas of a human user.

Analogical reasoning is an excellent candidate for serving such a function. In analogy, one situation is viewed as similar to another on the basis of relationships among the objects and actors in the situation. Once two situations are seen to be similar, predictions can be generated by carrying over information from one domain to another. In this paper, we discuss Structure-Mapping Theory (SMT; [Gentner, 1983]), which is the most prominent theory of analogical reasoning in the cognitive science literature. This theory has been implemented in a number of computational models. We discuss two implementations of SMT, the Structure Mapping Engine (SME; [Falkenhainer et.al., 1989]) and Associative-Projective Neural Networks [Rachkovskij, 2001]. We explore how these implementations can be used to supplement human reasoning in the domain of counterterrorism.

Analogical vs case-based reasoning

Past experience provides an important source of information for reasoning about terracts. In particular, it is important to recognize when a new situation is sufficiently like some previous situation that some action ought to be taken.

One approach to the use of prior knowledge involves expert systems in which domain experts are asked about the rules they use to reason. Unfortunately, experts are rarely able to articulate complete and correct general-purpose rules that they use to reason. Also, expert systems that use the rules obtained from experts often have difficulty with new cases that are not an exact match to those for which the rules were constructed.

Two other approaches apply information about specific known episodes to new situations. The first is case-based reasoning (CBR). On this approach, a KB is developed that consists of descriptions of cases that are indexed by key aspects of the environment such as the goals that the case is designed to solve. New situations with the same index call up prior cases. The prior cases are then examined for their appropriateness by determining whether the differences between the new situation and the old one are critical. If the differences are not important, then the old case is tweaked to allow it to be applied to the new situation. CBR has been applied in a number of domains [Kolodner, 1993; Schank, Kass, & Reisbeck, 1994], but it has a number of limitations including:

- it cannot deal with complex structured knowledge. Representations of cases are usually unstructured sets of attributes, without relational information;

- it is not so flexible as human reasoning. The indexing scheme that allows new cases to be accessed based on prior cases needs hand-tailoring;

- it lacks robust, domain-independent algorithms for finding similarity and applying information from a previous case to generate advice specific to the current case.

Analogical reasoning is one of the most commonly encountered and vivid cognitive processes. That is why a lot of work is devoted to the development of theories and computational models of analogy. (For an introduction, see, e.g., [Gentner & Markman, 1995, 1997; Holyoak & Thagard, 1989; Hummel & Holyoak, 1997; Thagard et. al, 1990; Eliasmith & Thagard 2001] and references therein).

Analogical reasoning provides the flexibility to reason on-the-fly given a knowledge base of historical cases by supporting comparisons between known episodes and current conditions. Counterterrorism requires the ability to flexibly extend existing ideas to new situations, and so analogical reasoning is a good match for it.

The analogical reasoning account supports a number of important sub-processes:

- finding relevant episodes in the KB and comparing them taking structure into account, as in human reasoning [Gentner, Rattermann, & Forbus, 1993];

- inferring new information in order to generate predictions [Clement & Gentner, 1991; Markman, 1997];

- retrieving and adjusting associated information in order to create action plans [Keane, 1996];

- reasoning with, learning and generalizing from examples and applying knowledge appropriately to the case situation [Forbus et al., 1999];

- providing the ability to understand analogies and metaphors in natural language texts [Fauconnier, 1997].

The ways to overcome the limitations of CBR using analogical reasoning are investigated in large-scale DARPA projects, High Performance Knowledge Bases [HPKB] and Rapid Knowledge Formation [RKF], that deal with construction of and reasoning over a large KB. As evidenced by those programs, analogical reasoning tools can serve as a basis for new AI technologies, with such applications as, e.g., crisis management or getting expertise in weapons of mass destruction.

Thus, introducing analogical reasoning to KB and expert system enables:

- a more advanced, human-like reasoning about the world (as in the HPKB project);

- support for creation of KBs (as in the RKF project);

- help in human interaction with KB, using natural language, examples and generalization;

- overcoming some limitations of human reasoning (such as the limited span of working memory, inability to extract precise description of past situations, limitations of input and exchange information channels, "standard" human thinking that limit the possibilities of creativity),

- overcoming other human limitations (such as the inability to work efficiently and to make decisions under pressure, the need to discuss with somebody, the difficulty to take responsibility, vigilance loss, etc.).

The Structure Mapping Theory and its symbolic implementation

The most advanced and elaborated theory of analogical reasoning is Gentner's Structure-Mapping Theory (SMT) of analogy and similarity [Gentner, 1983], further developed in [Gentner & Markman 1995, 1997; Markman 1997, Markman & Gentner, 2000]. The theory explains analogy and similarity in terms of comparisons involving structured representations, not just lists of features.

In an analogy, a base domain, which is the one people typically know more about, is compared to a target domain, which is typically the new situation to be reasoned about. The base and target are represented using structured hierarchical representations consisting of entities (the objects in the domain), attributes (one-place predicates that describe objects) and relations (two- or more-place predicates that relate entities, attributes, and other relations. Finding an analogy between two domains involves finding overlapping relational structure between the domains.

Two domains have overlapping relational structure if their representations contain some identical relations (i.e., they share semantic similarity), if matching relations have matching arguments (i.e., they display parallel connectivity), and if each element in one domain matches to at most one element in the other domain (i.e., there

is a one-to-one mapping). Considerable psychological evidence is consistent with the operation of these constraints in human analogical reasoning [Gentner & Markman, 1997].

Unlike CBR, analogical reasoning does not require that cases be indexed for later retrieval. Instead, access is mediated by retrieval algorithms that find base domains in memory that share similarity with a target probe. Retrieval involves two stages. First, the contents of memory are filtered to include only those domains that share substantial semantic similarity with the target. This stage does not require attention to the structure of the domain (i.e., to parallel connectivity or one-to-one mapping) and hence can be done in a computationally efficient manner. Furthermore, recent work suggests that the semantic similarity component of retrieval can be done using high-dimensional semantic space models of the lexicon, which eases the process of developing domain representations (Ramscar & Yarlett, 2003). Those domains that pass through the initial stage of retrieval are given a structural comparison to the target domain, and those domains with a good structural match are retrieved from memory and made available for further reasoning.

There is quite a bit of psychological support for the basic principles of SMT. For example, [Clement and Gentner, 1991] observed that people show a preference for systematicity in that the prefer analogies that have deeply connected relational structures to analogies that preserve only a limited set of disconnected matches. Furthermore, Markman (1997) demonstrated that one-to-one mapping is critical for the formation of analogies. In other work, Markman & Gentner [1997] have extended structure-mapping theory from analogies to comparisons in general, and have created a number of experimental methods for gathering evidence about the way people make comparisons. This theory provides the scientific basis for developing systems that approach the flexibility of human analogical reasoning, for analyzing and predicting of human behavior, for natural interaction with humans.

This psychological theory has been implemented in a number of computational models (e.g., Falkenhainer et al., 1989; Hummel & Holyoak, 1997; Keane, et. al., 1994). The most prominent model is the Structure Mapping Engine (SME) [Falkenhainer et. al.,1989], that has been used successfully in a number of state-of-the-art large scale AI (knowledge-related) projects, such as [HPKB; RKF]. SME is implemented as a symbolic model. The representations given to SME consist of predicate-argument representation structures that can be represented as directed acyclic graphs. Thus, the problem of finding an analogy between domains is computationally intractable in general, and so heuristics and limitations imposed by the SMT must be used to find analogical matches efficiently.

SME makes the analogy process computationally tractable by using a local-to-global match algorithm in which predicates in one domain are first matched to those in the other domain when they have identical semantics. After this initial sequence, matching predicates are assessed to determine whether they have matching arguments, and each set of matching predicates is checked to ensure that matches are one-to-one. This model requires manual construction of representations of items and structures based on them.

SME is also incremental. An initial correspondence between the base and target domains can be extended when more information is obtained about the domains. If analogy were only able to find correspondences between domains, it would be limited in its usefulness to counterterrorism situations. However, SME also generates candidate inferences by carrying over information from the base domain to the target when that information is connected to the correspondence between the domains and is structurally consistent with it. The central mechanisms embodied in SME are all consistent with what is known about the way people process analogies.

SME is one tool that can be used for the development of counterterrorism tools. The symbolic algorithm can be set up to operate with a KB of prior terracts and other relevant information. A second technique for analogical mapping is the APNN architecture. We describe it in the next section.

Analogical reasoning with APNNs

Associative-Projective Neural Networks [Kussul,1992; Kussul et.al. 1991] are based on a scheme for sparse binary distributed representations of information. These "structure-sensitive" distributed representations take into account both semantic and structural aspects of similarity. This approach provides an opportunity to combine the advantages of connectionist networks (semantic sensitivity, parallelism) and symbolic representations (compositionality, systematicity).

Traditional distributed representations allow a natural representation and computation of gradual similarity and make an efficient use of representational resources. Similar items are represented by correlated codevectors where similarity can be estimated using the dot product. A large information capacity is provided by the possibility to represent exponentially many items by different codevectors of the same dimensionality. Distributed representations are robust and neurobiologically plausible. Also, they allow unsupervised learning of similar representations for similar items using such methods as Learned Vector-Space Models (Latent Semantic Analysis, Context Vectors, Random Indexing, etc. [Caid et. al., 1995; Kanerva et. al., 2000]).

However, it was thought that distributed representations cannot represent nested (recursive) structures because of superposing pairs of vectors would lead to the loss of information about the relationship between elements and their arguments, see [Rachkovskij & Kussul, 2001] for discussion and references). In APNNs, Holographic Reduced Representations (HRRs) [Plate 1995; 2000], Binary Spatter Codes (BSCs) [Kanerva 1996, 1998], the scheme of [Gayler & Wales, 1998], it has been possible to create on-the-fly (without any training) distributed representations of recursive compositional structures with codevectors of the same dimensionality for arbitrary (even novel, non-similar) items.

Complex structures are chunks of a small number of component (sub)structures. The codevectors for more complex structures are built from the set of component codevectors. Because each component may itself be a complex compositional (sub)structure, structures of arbitrary complexity can be represented. To preserve grouping of components in chunks of various compositional levels, binding by Context Dependent Thinning [Rachkovskij & Kussul, 2001] is used in APNNs for component codevectors of each chunk. This allows an on-the-fly construction of a composite bound codevector from its component codevectors.

APNNs encode items of any nesting level, elementary or compositional, are represented by large codevectors of the same dimensionality. The codevectors are binary (with 0 or 1 elements) and sparse (with small fraction of 1s), e.g., with N=100,000 elements representing neurons and M=1,000 of 1s. To build APNN representations of the episodes, it is necessary to encode the entities and relations using base-level codevectors of the lowest composition level. Random independently generated codevectors were used for base-level codevectors in [Rachkovskij, 2001], though correlated codevectors for similar items are possible. Similar items are represented by codevectors with a more-than-random overlap of 1s.

Thus, distributed representations of structures can be constructed that carry immediate information on both the set of structural components of various hierarchical levels and their structural organization. Similarity of resulting bound representations is influenced both by the set of components and their arrangements. So, similar structures are encoded by similar codevectors. Therefore, it is not necessary to search for the match between the elements of two structures in order to estimate their overall similarity by dot product of their codevectors. Similarity found by one-shot dot product of APNN codevectors takes into account both semantic and structural similarity of episodes. Such representations exemplify “reduced descriptions” [Hinton, 1990] or "meaningful symbols" [Kanerva, 2000] that are central to analogical reasoning.

A mapping process has been defined for APNNs that uses alternating sequential and parallel steps to find structural correspondences between pairs of representations [Rachkovskij, under revision]. Processing is based on similarity preservation in reduced representations and includes finding similarity between elements of the same hierarchical level for mapping, and between elements of different levels for structure traversing.

In order to map two analogs, first their elements (chunks of all hierarchical levels) must be encoded by codevectors. Then, the simplest mapping technique involves placing in correspondence the analogical elements of the same hierarchical level having the largest overlap of codevectors. For interpretation of more formal analogies, a technique with synchronous traversal of hierarchical representations may be required, such as finding the corresponding roles and putting into correspondence their fillers. The consistency of mapping can be verified by checking if both techniques produce the same results. Using various attribute structures and representation schemes, and even changing them in the process of mapping may be required for mapping of more difficult analogies.

Usage of distributed associated memories with fast or even one-shot storage and fast retrieval of the most similar episodes [Frolov et.al., 2002] is also facilitated by sparse and binary character of the APNN representations. These features provide APNNs with a scaling potential and flexibility necessary for analogy-processing for large-scale real-world problems emerging at various stages of counterterrorism activity.

Application of analogical reasoning to antiterrorism tasks

Analogical reasoning can facilitate efforts to minimize the effects of terrorist acts by allowing the use of past episodes to influence the interpretation of current events. There are three stages at which analogical reasoning can be used: prevention of new acts, reasoning about the course of ongoing terrorist activity, and reasoning about the consequences of a new act.

Before terrorist acts, the task is to *prevent* them and *prepare* countermeasures. These include: revealing terrorist groups and individuals; prediction of terracts; preparation of countermeasures (general and specific); full-spectrum assessment of and preparation for threats, effects, and consequences; specialized training of personnel and people (from preventing to responding to an incident); action planning and executing in view of a potential terract. *Within* and *after* terrorist acts, the task is to *interdict*, *mitigate*, *learn*, and *prevent new* terracts, including: optimal response to acts of terrorism; evacuation and emergency medical aid; assess the affected population and damage; emergency and consequence management; prevention of follow-up terracts; study of the terract and its effects.

Applications of analogical reasoning in computer-based systems dealing with various aspects of the life-cycle of terrorist acts for minimizing their effects on civilian populations may include, but are not exhausted, by the following:

- constructing case libraries or KB of life-cycles of known terrorist attacks in order to develop countermeasures to them, to analyze what novel terracts may be and prepare to them;

- inclusion in the KB of information about previous cases that can be used for predicting terracts;

- inclusion of information about what was done after the previous terracts, including errors and correct actions, and giving action planning proposals for the current terract;

- generation of new terract scenarios by analogy to prior attacks, given an initial set of conditions, e.g., a terrorist group, its assets, information about potential targets, etc. These scenarios could then be used to test current anti-terrorist measures as well as emergency response preparations;

- supporting analysis and expert decision making for various scenarios of terracts, from prediction through the course to response;

- inventing new schemes or some aspects of terracts by relaxing some constraints on human analogical reasoning or otherwise changing real cases thus simulating reasoning of terrorists learning from previous attempts of terracts and using this information for analysis and developing countermeasures;

- a better understanding of the way people behave after a terrorist attack, in particular how people make plans by drawing analogies to that attack, such as avoiding doing things that can lead to getting into situation similar to that of a recent terract.

When performing analytic tasks, analysts use analogies in at least four ways

-for organizing and understanding information arriving about a new, emerging or otherwise unfamiliar situations;

-to sharpen understanding of a current situation by comparing it to past situations;

-to help test assumptions about new situations;

-to test for projection of our own biases, goals, values and thought systems into uncertain situations.

Consider the case of generating a prediction for a new terract. There is good evidence that when people generate new ideas, they do so by analogy to known ideas (e.g. [Gentner, et al., 1997; Moreau, Markman, & Lehmann, 2001; Ward, 1994]. Using analogical reasoning, a known incident can be combined with a new set of conditions to predict a new incident. For example, the 1972 hostage crisis at the Munich Olympic games can be combined with the conditions predicted to exist at the 2004 Summer Olympic games in Athens to create a novel scenario for these games. Scenarios generated in this way can be used to test readiness of Civil Defense personnel and to suggest potential suspicious activities that might alert authorities of possible terracts. Different patterns of terracts such as homicide bombers, hijacking planes, and the attempted bombing of the WTC at the beginning of 1990 could provide analogy to the Sept. 11 attack.

Implementation issues

We propose the following scheme to implement and test this approach. The analogical reasoning tools for analogical access, mapping, and inference operating with complex structured episodes should be implemented based the techniques and algorithms of the APNN system. Then the system should be tested on terrorism-related episodes from a KB developed for this purpose. These require:

(1) Finding, constructing, and adapting benchmark episodes that describe the life-cycle of terrorist acts for elaborating and testing analogical reasoning techniques in counterterrorism-related tasks. These benchmark episodes can be constructed by the researchers, with the help of experts where appropriate, based on information that is readily available from public-domain sources (e.g., newspapers, historical records, Internet [Terrorism-related resources]). Obviously, more detailed scenarios can be constructed using information that is not typically publicly available (e.g., military intelligence). The principles of APNNs described here will provide more detailed predictions for new situations when the base domains where they are given are similarly detailed.

(2) Implementing reasoning tools using APNNs:

- parsing of input symbolic representations and its transformation into internal XML-based format of episodes;

- setting the schemes for relational representations; distributed encoding of analogical episodes; analogical access, mapping, inference.

These tools will enable retrieval of previously supplied episodes of terracts and their usage for solving problems concerning new terracts. Then the system can be tested with the constructed complex episodes and situations describing the life-cycle of terrorist acts, such as preparations, execution, consequences. These sample terrorism-related benchmark episodes are meant to be demonstrations of the utility of this approach. Of particular interest is the degree to which descriptions of prior terracts can be used to generate predictions of new scenarios.

Conclusion

Thus, our approach consists in using techniques of Computer Science, specifically Artificial Intelligence and Neural Networks, and combining them with knowledge obtained through the progress of cognitive science in order to create analogical processing tools that provide new quality to counterterrorism systems.

The main goal is twofold:

- to develop the set of ideas and techniques supporting terrorism-related analogical reasoning using structure-sensitive distributed representations within the architecture of Associative-Projective Neural Networks;

- to implement software components and a prototype of an analogical processing toolkit that can potentially enhance the intellectual level of computer-based systems for a wide range of personnel dealing with various aspects of the pressing problem of terrorism and its effects.

The new methods and techniques of analogical reasoning with meaning- and structure-sensitive distributed representations should be developed, investigated, and applied to the problem of counterterrorism. They combine the advantages of distributed representations (possibility to be learnt, natural representation and estimation of similarity, an efficient use of representational resources, generalization potential, etc.) with the necessity of complex structured representations to describe adequately the real-world situations, such as terrorist incidents and associated scenarios, plans, analyses, predictions, countermeasures, etc.

This new knowledge will have a substantial impact on the efforts to minimize the effects of terrorist acts on civilian populations by facilitating knowledge acquisition and formation of terrorism-related knowledge bases, as well as supporting the processes of analysis, decision making, and reasoning with those KBs for users at various levels, from expert to usual personal, before, in the process, and after terracts.

This technology has a potential commercial value at the market place by enhancing intelligence level of existing computer systems, as well as providing a basis for development of a new generation of such systems, that help to provide technological solutions to the threat of terrorist acts, as well as other KB systems. This technique can also provide a basis for developing a psychological theory of human analogical reasoning that is consistent with behavioral data and is also neurally plausible.

Bibliography

[Caid, 1995] W.R.Caid, S.T.Dumais, & S.I.Gallant (1995) Learned vector-space models for document retrieval. Information Processing and Management, Vol. 31, No. 3, pp. 419-429.

[Clement & Gentner, 1991] C.A.Clement & D.Gentner (1991). Systematicity as a selection constraint in analogical mapping. *Cognitive Science, 15*, 89-132.

[Eliasmith & Thagard, 2001] C.Eliasmith & P.Thagard (2001). Integrating Structure and Meaning: A Distributed Model of Analogical Mapping. Cognitive Science, 25(2), 245-286.

[Fauconnier, 1997] G.Fauconnier (1997). *Mappings in thought and language*. New York: Cambridge University Press.

[Falkenhainer et.al., B. Falkenhainer, K.D.Forbus, & D.Gentner (1989) The Structure-Mapping Engine: Algorithm and Examples. Artificial Intelligence, 41, 1-63.

[Forbus et.al., 1999] K.D. Forbus, P.B.Whalley, J.O.Everett, L.Ureel, M.Brokowski, J.Baher, & S.Kuehne (1999). CyclePad: An articulate virtual laboratory for engineering thermodynamics. *Artificial Intelligence, 114*, 297-347.

[Frolov et.at., 2002] A.A.Frolov, D.A.Rachkovskij, D.Husek (2002). Informational efficiency of sparsely encoded Hopfield-like autoassociative memory.

[Gayler & Wales, 1998] R. W.Gayler, R.Wales (1998).Connections, binding, unification and analogical promiscuity. In: K. Holyoak, D. Gentner, & B. Kokinov (Eds.): Advances in analogy research: Integration of theory and data from the cognitive, computational, and neural sciences.(Proc. Analogy'98 workshop, Sofia). NBU Series in Cognitive Science, Sofia, Bulgaria: New Bulgarian University, pp. 181-190.

[Gentner, 1983] D.Gentner (1983). Structure-mapping: A theoretical framework for analogy. Cognitive Science, 7, pp. 155-170.

[Gentner et al., 1997] D. Gentner, S. Brem, R. Ferguson, A.B. Markman, B.B. Levidow, P. Wolff, & K.D. Forbus (1997). Conceptual change via analogical reasoning: A case study of Johannes Kepler. *Journal of the Learning Sciences*, *6,* 3-40.

[Gentner & Markman, 1995] D.Gentner & A.B.Markman (1995). Analogy-Based Reasoning. In M. A. Arbib (Ed.), Handbook of brain theory and neural networks (pp. 91-93). Cambridge, MA: MIT Press.

[Gentner & Markman, 1997] D.Gentner & A.B.Markman (1997). Structure Mapping in Analogy and Similarity. American Psychologist, 52(1), 45-56.

[Gentner et.al. 1993] D.Gentner, M.J.Rattermann, & K.D.Forbus (1993). The roles of similarity in transfer: Separating retrievability from inferential soundness. *Cognitive Psychology, 25*(4), 524-575.

[Hinton, 1990] G.Hinton (1990). Mapping part-whole hierarchies into connectionist networks. Artificial Intelligence 46, 47-75.

[Holyoak & Thagard, 1989] K.J.Holyoak, & P.Thagard (1989). Analogical mapping by constraint satisfaction. Cognitive Science, 13, 295-355.

[HPKB] DARPA's High Performance Knowledge Base Initiative http://projects.teknowledge.com/HPKB/

[Hummel & Holyoak, 1997] J.E.Hummel & K.J Holyoak (1997). Distributed representations of structure: A theory of analogical access and mapping. Psychological Review, 104, 427-466.

[Kanerva et.al., 2000] P.Kanerva, J.Kristoferson, and A.Holst (2000) "Random indexing of text samples for Latent Semantic Analysis." In L.R. Gleitman and A.K. Josh (eds.), Proc. 22nd Annual Conference of the Cognitive Science Society (U Pennsylvania), p. 1036. Mahwah, New Jersey: Erlbaum.

[Kanerva, 1996] P.Kanerva (1996). Binary Spatter-Coding of Ordered K-tuples. In C. von der Malsburg, W. von Seelen, J. C. Vorbruggen, & B. Sendhoff (Eds.). Proceedings of the International Conference on Artificial Neural Networks - ICANN'96, Bochum, Germany. Lecture Notes in Computer Science, 1112, 869-873. Berlin: Springer.

[Kanerva, 1998] P.Kanerva (1998) "Dual role of analogy in the design of a cognitive computer." In: K. Holyoak, D. Gentner, and B. Kokinov (Eds.), Advances in Analogy Research: Integration of Theory and Data from the Cognitive, Computational, and Neural Sciences (Proc. Analogy'98 workshop). NBU Series in Cognitive Science, Sofia, Bulgaria: New Bulgarian University, pp. 164-170.

[Kanerva, 2000] P.Kanerva (2000) Large patterns make great symbols: An example of learning from example. In S. Wermter and R. Sun (eds.), HYBRID NEURAL SYSTEMS (pp. 194-203). Heidelberg: Springer.

[Keane, 1996] M.T.Keane (1996). On adaptation in analogy: Tests of pragmatic-importance and adaptability in analogical problem solving. *Quarterly Journal of Experimental Psychology, 49A*(4), 1062-1085.

[Keane et.al., 1994] M.T.Keane, T.Ledgeway, & S.Duff (1994). Constraints on Analogical Mapping: A comparison of Three Models. Cognitive Science 18, pp. 387-438.

[Kolodner, 1993] J. Kolodner (1993). *Case-based reasoning*. San Mateo, CA: Morgan Kaufmann Publishers, Inc.

[Kussul, 1992] E.M.Kussul (1992) Associative neuron-like structures. Kiev: Naukova Dumka. (In Russian).

[Kussul et.al, 1991] E.M.Kussul, D.A.Rachkovskij & T.N.Baidyk (1991) Associative-Projective Neural Networks: architecture, implementation, applications. In Proceedings of the Fourth International Conference "Neural Networks & their Applications", Nimes, France, Nov. 4-8, 1991 (pp. 463-476).

[Markman, 1997] A.B.Markman (1997). Constraints on Analogical Inference. Cognitive Science, 21(4), pp.373-418.

[Markman & Gentner, 2000] , A.B.Markman & D.Gentner (2000) Structure mapping in the comparison process. American Journal of Psychology, 113, 501-538.

[Moreau, Markman, & Lehmann, 2001] C.P. Moreau, A.B. Markman, & D.R. Lehmann (2001). 'What is it?' Categorization flexibility and consumers' responses to really new products. *Journal of Consumer Research*, *27*, 489-498.

[Plate, 1995] T.A.Plate (1995). Holographic Reduced Representations. IEEE Transactions on Neural Networks, 6, 623-641.

[Plate, 2000] T.A.Plate (2000). Analogical Retrieval and Processing with Distributed Vector Representations. Expert Systems: The International Journal of Knowledge Engineering and Neural Networks, 17(1), pp. 29-40.

[Rachkovskij, 2001] D.A.Rachkovskij, (2001) Representation and processing of structures with binary sparse distributed codes. IEEE TKDE 13(2), pp. 261-276.

[Rachkovskij, under revision] D.A. Rachkovskij (2002) Some approaches to analogical mapping with structure-sensitive distributed representations (under revision).

[Rachkovskij & Kussul, 2001] D.A.Rachkovskij & E.M.Kussul (2001) Binding and Normalization of Binary Sparse Distributed Representations by Context-Dependent Thinning. Neural Computation 13(2), pp. 411-452 (paper draft available at http://cogprints.soton.ac.uk/abs/comp/199904008).

[Rachkovskij & Kussul, unpublished] D.A.Rachkovskij & E.M.Kussul. Building large-scale hierarchical models of the world with binary sparse distributed representations. http://www.bbsonline.org/Preprints/Rachkovskij/Referees/Rachkovskij.pdf

[Ramscar & Yarlett, 2003] M.Ramscar and D.Yarlett - Semantic grounding in models of analogy: An environmental approach. 27(1), 2003

[RKF] DARPA's Rapid Knowledge Formation Program http://reliant.teknowledge.com/RKF/

[SAIC] SAIC's Counterterrorism Webpage http://www.saic.com/natsec/counterterrorism

[Schank, Kass, & Riesbeck, 1994]. R.C. Schank, A. Kass, & C.K. Riesbeck (1994). *Inside case-based explanation*. Hillsdale, NJ: Lawrence Erlbaum Associates.

[Terrorism-related resources] Terrorism-related resources of Message Understanding Conference (MUC 3-4). Resources at the International Association For Counterterrorism & Security Professionals http://www.iacsp.com. UN terrorism-related resources http://www.un.org/Depts/dhl/resources/terrorism/elinks.htm. U.S. Department of State counterterrorism resources http://www.state.gov/s/ct.

[Thagard, 1990] P. Thagard, K.J.Holyoak, G.Nelson, D.Gochfeld Analog Retrieval by Constraint Satisfaction Artificial Intelligence 46(1990) 259-310

[Ward, 1994] Ward, T.B. (1994). Structured imagination: The role of category structure in exemplar generation, Cognitive Psychology, 27, 1-40.

Author information

Arthur B. Markman - Department of Psychology, University of Texas, Austin, TX 78712, USA. email: markman@psy.utexas.edu

Dmitri A. Rachkovskij, Ivan S. Misuno, Elena G. Revunova - International Research and Training Center of Information Technologies and Systems, Pr. Acad. Glushkova 40, Kiev 03680, Ukraine. email: dar@infrm.kiev.ua

THE INFORMATION

Kr. Markov, Kr. Ivanova, I. Mitov

***Abstract**: The "Information" is basic concept for the General Information Theory. The formal definition of it is presented in this work.*

***Keywords**: Information, General Information Theory, and Philosophy of Informatics*

Introduction

The General Information Theory (GIT) has been established as internal non-contradictory logical system of contentions. It is based only on primary consideration of the world as variety of entities, which are formed by relationships between entities from lower levels.

The fundamental notion of the GIT is the concept "information". All other concepts are defined on the basis of this definition.

In 1988, the not formal definition of the concept of Information was published in [Markov, 1988]. It became as a fundamental concept for the General Information Theory [Markov et al, 1993].

Now, the formal definition of the concept of Information is presented in this work. For easy reading, the philosophical explanations before formal definitions in the paper are reproduced from [Markov et al, 1993] and are given without special indication.

This paper is based on the ideas considered during very creative discussions at the International Conference "KDS 1997", September 1997, Yalta, Ukraine, and at the International Conference "ITA 2000", September 2000, Varna, Bulgaria, as well as at the previous scientific meetings organised by International Workgroup on Data Base Intellectualisation (IWDBI).

1. Entities and Relationships

In our examination we consider *the real world* as a space of *entities.* The entities are build out of other entities, connected with *relationships*. The entities and relationships between them form the internal *structure* of the entity they build. To build up the entity of a certain structural level of the world it is necessary to have:
- the entities of the lower structural level;
- establishing of the forming relationship.

The primary entity can dialectically be considered as a relationship between its entities of the first lowers structural level.

*The entities and relationships of all internal structural levels form **the entire entity**.*

The forming relationships have a representative significance for the new entity. The destruction of the essential (forming) relationships in one entity leads to its disintegration. The establishment of forming relationships between already existing entities has a determine significance for the emerging of the new entity.

The forming relationships are the reason (which leads to) of *the emergence* of individual properties, which distinguish the new entity from the forming ones. They exist at the lower structural level, but the relationship emerges only in passing to the level of the new entity.

***The relationships form up and present the entity**.*

1.1. Primary entity

Definition 1a. The ***primary entity*** E is the internal connected couple $E = (E^E, R^E)$ where:

E^E *is a set of sub-entities* of E, i.e. $E^E = \{ e_i , \varnothing \mid e_i \in E^E, \varnothing \in E^E , i = 1,2,..,n\}$,

R^E *is a set of relations* in $E^E x E^E$ with composition "$\circ$", i.e. $R^E = \{(e_i,e_j) \mid e_i,e_j \in E^E\}$,

The condition of internal connectivity of E is:

$\forall\ e_i,e_j \in E^E \rightarrow ((\exists\ (e_i,e_j) \in \mathbf{R}^E) \vee (\exists\ z_1,..,z_p \in E^E,\ z_k \neq \varnothing,\ k = 1,..,p: ((e_i,z_1) \circ (z_1,z_2) \circ .. \circ (z_p,e_j) \in R^E))$

E^E is called ***forming set*** of E,

$\mathbf{R}^E$ is called ***forming relationship*** of E. ⊐

1.2. Entire entity

Definition 1b. The ***entire entity*** $\mathfrak{E}$ is the internal connected couple $\mathfrak{E} = (\mathfrak{E}^E,\ \mathfrak{R}^E)$ where:

$\mathfrak{E}^E$ is the set of the primary sub-entities of E for which:

1. $\varnothing \in \mathfrak{E}^E$;
2. $E \in \mathfrak{E}^E$;
3. If $X \in \mathfrak{E}^E$ then X is a primary sub-entity;
4. If $X_i \in \mathfrak{E}^E$ and $U = \cup X_i$, i=I,..,n, n-arbitrary, then $U \in \mathfrak{E}^E$
5. If $X_i \in \mathfrak{E}^E$ and $V = \cap X_i$, i=I,..,m, m-arbitrary, then $V \in \mathfrak{E}^E$.

It is clear, $\mathfrak{E}^E$ is a topology in E and corresponding topological space is (E, E^E).

$\mathfrak{R}^E$ is the set of the mappings in $\mathfrak{E}^E$ x $\mathfrak{E}^E$ with composition "$\rightarrow$" : $\mathfrak{R}^E = \{ (X,Y) \mid X \rightarrow Y;\ X,Y \in \mathfrak{E}^E \}$

The condition of internal connectivity of $\mathfrak{E}$ is:

$\forall X,Y \in \mathfrak{E}^E \rightarrow ((\exists\ (X,Y) \in \mathfrak{R}^E) \vee (\exists\ Z_1,\ldots,Z_p \in \mathfrak{E}^E ,\ Z_k \neq \varnothing,\ k = 1,..,p : ((X,Z_1,) \rightarrow (Z_1,Z_2) \rightarrow \ldots \rightarrow (Z_p,Y) \in \mathfrak{R}^E))$

$\mathfrak{E}^E$ is called forming set of $\mathfrak{E}$,

$(E, \mathfrak{E}^E)$ is called forming space of $\mathfrak{E}$,

$\mathfrak{R}^E$ is called forming relationship of $\mathfrak{E}$. ⊐

2. Contacts and Interactions

The entities ***contact*** each other during the building the relationship.

The composition of contacts between entities forms their ***interaction***.

The contacts of the given structural level are processes of interaction of the lower levels of the entities.

The contacts and interaction are real and time depended processes, i.e. the establishing of the contact is an event, which needs time to be realised. Because of this, the mappings and the diagram below are time depended.

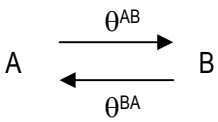
The time needed to establish the primary contact is assumed an ***integral time quantum***. For different primary contacts, the corresponded time quantum may be different.

The time needed to establish the entire contact is assumed an ***elementary time interval***. For different entire contacts, the corresponded time intervals may be different.

2.1. Direct contact

Definition 2a. The ***primary direct contact*** ψ^{AB} between A and B is the set of mappings

$$\psi^{AB} = \{ \theta^{AB}, \theta^{BA}\},$$



where:

A= (E^A, $\mathbf{R}^A$) and B = (E^B, $\mathbf{R}^B$) are primary entities and the corresponding mappings are:

$\theta^{AB} = (A \to B) = (\theta_e^{AB} : E^A \to E^B , \theta_r^{AB} : R^A \to R^B) =$

$= \{ x_i , x_j , y_k , y_l , (x_i, x_j) , (y_k , y_l) \mid x_i \to y_k , x_j \to y_l , (x_i, x_j) \to (y_k , y_l) ,$
$x_i , x_j \in E^A, y_k , y_l \in E^B, (x_i, x_j) \in R^A, (y_k , y_l) \in R^B \} .$

$\theta^{BA} = (B \to A) = (\theta_e^{BA} : E^B \to E^A , \theta_r^{BA} : R^B \to R^A) =$

$= \{ x_i , x_j , y_k , y_l , (x_i, x_j) , (y_k , y_l) \mid y_k \to x_i, y_l \to x_j, (y_k , y_l) \to (x_i, x_j),$
$y_k , y_l \in E^B, x_i , x_j \in E^A, (y_k , y_l) \in R^B, (x_i, x_j) \in R^A \} .$

Therefore, we may write the following equations:

$$\psi^{AB} = \{\theta^{AB}, \theta^{BA}\} = \{\{\theta_e^{AB}, \theta_r^{AB}\} , \{\theta_e^{BA}, \theta_r^{BA}\}\} = \{\{\theta_e^{AB}, \theta_e^{BA}\} , \{\theta_r^{AB}, \theta_r^{BA}\}\} = \{\psi_e^{AB}, \psi_r^{AB}\} \multimap$$

Definition 2b. The ***entire direct contact*** Ψ^{AB} between $\mathfrak{A}$ and $\mathfrak{B}$ is a set of mappings

$$\Psi^{AB} = \{ \Theta^{AB}, \Theta^{BA} \}$$

connected by the time depended diagram

$$\mathfrak{A} \underset{\Theta^{BA}}{\overset{\Theta^{AB}}{\rightleftarrows}} \mathfrak{B}$$

where:

$\mathfrak{A}$ = ($\mathfrak{E}^A$, $\mathfrak{R}^A$) and $\mathfrak{B}$ = ($\mathfrak{E}^B$, $\mathfrak{R}^B$) are entire entities and the corresponding mappings are:

$\Theta^{AB} = (\mathfrak{A} \to \mathfrak{B}) = (\Theta_{\mathfrak{E}}^{AB} , \Theta_{\mathfrak{R}}^{AB})$

$\Theta_{\mathfrak{E}}^{AB} = (\mathfrak{E}^A \to \mathfrak{E}^B) = \{ X, Y \mid X \to Y, X \in \mathfrak{E}^A , Y \in \mathfrak{E}^B \}$

$\Theta_{\mathfrak{R}}^{AB} = (\mathfrak{R}^A \to \mathfrak{R}^B) = \{ (X_1 , X_2) , (Y_1 , Y_2) \mid (X_1 , X_2) \to (Y_1 , Y_2) ,$
$X_1 , X_2 \in \mathfrak{E}^A , (X_1 , X_2) \in \mathfrak{R}^A, Y_1 , Y_2 \in \mathfrak{E}^B , (Y_1 , Y_2) \in \mathfrak{R}^B \} .$

$\Theta^{BA} = (\mathfrak{B} \to \mathfrak{A}) = (\Theta_{\mathfrak{E}}^{BA} , \Theta_{\mathfrak{R}}^{BA})$

$\Theta_{\mathfrak{E}}^{BA} = (\mathfrak{E}^B \to \mathfrak{E}^A) = \{ X, Y \mid Y \to X, X \in \mathfrak{E}^A , Y \in \mathfrak{E}^B \}$

$\Theta_{\mathfrak{R}}^{BA} = (\mathfrak{R}^B \to \mathfrak{R}^A) = \{ (X_1 , X_2) , (Y_1 , Y_2) \mid (Y_1 , Y_2) \to (X_1 , X_2),$
$X_1 , X_2 \in \mathfrak{E}^A , (X_1 , X_2) \in \mathfrak{R}^A, Y_1 , Y_2 \in \mathfrak{E}^B , (Y_1 , Y_2) \in \mathfrak{R}^B \} .$

It is clear, the entire contact is

$$\Psi^{AB} = \{ \Theta^{AB}, \Theta^{BA} \} = \{(\Theta_{\mathfrak{E}}^{AB}, \Theta_{\mathfrak{R}}^{AB}), (\Theta_{\mathfrak{E}}^{BA}, \Theta_{\mathfrak{R}}^{BA})\} = \{(\Theta_{\mathfrak{E}}^{AB}, \Theta_{\mathfrak{E}}^{BA}), (\Theta_{\mathfrak{R}}^{AB}, \Theta_{\mathfrak{R}}^{BA})\} = \{\Psi_{\mathfrak{E}}^{AB}, \Psi_{\mathfrak{R}}^{AB}\} \multimap$$

2.2. Transitive contact

Definition 3a. The ***primary transitive contact*** $\xi^{A(B)C}$ between A and C through B is the composition of primary contacts ψ^{AB} and ψ^{BC}, i.e.:

$$\xi^{A(B)C} = \psi^{BC} \circ \psi^{AB} = \{ \psi_e^{BC} \circ \psi_e^{AB} , \psi_r^{BC} \circ \psi_r^{AB} \} \multimap$$

Definition 3b. The ***entire transitive contact*** $\Xi^{A(B)C}$ between $\mathfrak{A}$ and $\mathfrak{C}$ through $\mathfrak{B}$ is the composition of entire contacts Ψ^{AB} and Ψ^{BC}, i.e.:

$$\Xi^{A(B)C} = \Psi^{BC} \multimap \Psi^{AB} = \{ \Psi_{\mathfrak{E}}^{BC} \multimap \Psi_{\mathfrak{E}}^{AB} , \Psi_{\mathfrak{R}}^{BC} \multimap \Psi_{\mathfrak{R}}^{AB} \} \multimap$$

2.3. Transitive self-contact

Definition 4a. The ***primary transitive self-contact*** of A with itself is the primary transitive contact $\xi^{A(B)A}$ from A to A through B. i.e.

$$\xi^{A(B)A} = \psi^{BA} \circ \psi^{AB} = \{ \psi_e^{BA} \circ \psi_e^{AB} , \psi_r^{BA} \circ \psi_r^{AB} \} \multimap$$

Definition 4b. The ***entire transitive self-contact*** of $\mathfrak{A}$ with itself is the entire transitive contact $\Xi^{A(B)A}$ from $\mathfrak{A}$ to $\mathfrak{A}$ through $\mathfrak{B}$ i.e.

$$\Xi^{A(B)A} = \Psi^{BA} \multimap \Psi^{AB} = \{ \Psi_{\mathfrak{E}}^{BA} \multimap \Psi_{\mathfrak{E}}^{AB}, \Psi_{\mathfrak{R}}^{BA} \multimap \Psi_{\mathfrak{R}}^{AB} \} \multimap$$

2.4. Interactions

Definition 5a. The ***primary interaction*** between A and B is a composition δ^{AB} of primary (direct and/or transitive) contacts between A and B:

$$\delta^{AB} = \{ \delta_1 \circ \delta_2 \circ \ldots \circ \delta_h \mid \delta_t , t=1,2,\ldots,h \in$$
$$\in \{\psi_i^{AB}, \psi_j^{AB}, \xi_k^{A(X)B}, \xi_l^{B(X)A} \mid i=1,2,\ldots,n; j=1,2,\ldots,m; k=1,2,\ldots,p ; l=1,2,\ldots,q ; X: \text{arbitrary}\} \} .$$

The main ordinary types of the primary interaction are:

- ***primary direct interaction:***
  $\delta^{AB} = \{ \delta_1 \circ \delta_2 \circ \ldots \circ \delta_h \mid \delta_t , t=1,2,\ldots,h \in \{\psi_i^{AB}, \psi_j^{AB} \mid i=1,2,\ldots,n; j=1,2,\ldots,m; \}$
- ***primary transitive interaction:***
  $\delta^{AB} = \{ \delta_1 \circ \delta_2 \circ \ldots \circ \delta_h \mid \delta_t , t=1,2,\ldots,h \in \{\xi_k^{A(X)B}, \xi_l^{B(X)A} \mid k=1,2,\ldots,p ; l=1,2,\ldots,q\}; X : \text{arbitrary}\}$
- ***primary self-interaction:***
  $\delta^{AB} = \{ \delta_1 \circ \delta_2 \circ \ldots \circ \delta_h \mid \delta_t , t=1,2,\ldots,h \in \{\xi_k^{A(X)A} \mid k=1,2,\ldots,p ; X : \text{arbitrary}\} \} \_$

Definition 5b. The ***entire interaction*** between $\mathfrak{A}$ and $\mathfrak{B}$ is a composition Δ^{AB} of entire contacts

$$\Delta^{AB} = \{ \Delta_1 \rightarrow \Delta_2 \rightarrow \ldots \rightarrow \Delta_h \mid \Delta_t , t=1,2,\ldots,h \in$$
$$\in \{\Psi_i^{AB}, \Psi_j^{AB}, \Xi_k^{A(X)B}, \Xi_l^{B(X)A} \mid i=1,2,\ldots,n; j=1,2,\ldots,m; k=1,2,\ldots,p ; l=1,2,\ldots,q ; X : \text{arbitrary}\}\} .$$

The main ordinary types of the entire interaction are:

- ***entire direct interaction:***
  $\Delta^{AB} = \{ \Delta_1 \rightarrow \Delta_2 \rightarrow \ldots \rightarrow \Delta_h \mid \Delta_t , t=1,2,\ldots,h \in \{\Psi_i^{AB}, \Psi_j^{AB} \mid i=1,2,\ldots,n; j=1,2,\ldots,m; \}$
- ***entire transitive interaction:***
  $\Delta^{AB} = \{ \Delta_1 \rightarrow \Delta_2 \rightarrow \ldots \rightarrow \Delta_h \mid \Delta_t , t=1,2,\ldots,h \in \{\Xi_k^{A(X)B}, \Xi_l^{B(X)A} \mid k=1,2,\ldots,p ; l=1,2,\ldots,q; X : \text{arbitrary}\}\}$
- ***entire self-interaction:***
  $\Delta^{AB} = \{ \Delta_1 \rightarrow \Delta_2 \rightarrow \ldots \rightarrow \Delta_h \mid \Delta_t , t=1,2,\ldots,h \in \{\Xi_k^{A(X)A} \mid k=1,2,\ldots,p; X : \text{arbitrary}\} \} \_$

3. Reflection

During the establishing of the contact, the influence of an entity changes temporally or permanently the internal structure of the other contacted entity.

In other words, the realisation (emergence) of the relationships between entities changes (temporary or permanently) their internal structure at one or at few levels. This change reflects (on) the entities by the emergence of new relationships (different from the forming relationship), and the depth of the change may be different. This means, that a (temporal or permanent) change is possible in the depth of the forming entities.

The internal change in the entities, which is due to the establishment of one relationship we denote with the notion "***reflection***".

The reflection (change in one entity) is determined by the relationship, which initiates it, and by the entities included in this relationship. The relationship and the entities it includes "reflect" on every entity included in this relationship. This reflection is not the relationship itself, but is a result from its appearance.

When one relationship is reflected (temporary or permanently), the entities, which take part in the interaction, are reflected too. This means that the forming entities and relationships are reflected (in different grade).

The entities of the world interact continuously in the time. It is possible, after any interaction may be realised another. In this case the changes received by any entity, during the first interaction, may be reflected by the new entity.

This means the ***secondary (transitive external) reflection*** exists. The chain of the transitive reflections is not limited.

Some entities have an opportunity of ***self-reflection***. *The self-reflection (self-change) of the entity leads to the creating of new relationships (and corresponding entities) in it.* It is clear, the self-reflection is a result of the interaction provided between entities in the low levels of the structure of the entity. The self-reflection is possible

only for very high levels of organisation of the entities, i.e. for entities with very large and complicated structure. Such kind of entities has relatively free sub-entities with own behaviour in the frame of self-preservation of the whole entity.

As a result of the self-reflection the new relationships (and corresponding sub-entities) may be created in the entity. These are internal defined relationships or shortly - "internal relationships".

One special case is the external ***transitive self-reflection*** where the entity reflects its own relationship as a secondary reflection during any external interaction. It is clear, the internal transitive self-reflection may exists but only on lower internal levels of the entity. The combination of the internal and external self-reflection is possible too.

3.1. Direct reflections

Definition 6a. Let $\psi^{AB} = \{ \theta^{AB}, \theta^{BA}\}$ is a primary direct contact between A and B, i.e.

$$A \underset{\theta^{BA}}{\overset{\theta^{AB}}{\rightleftarrows}} B$$

where $A= (E^A, \mathbf{R}^A)$ and $B = (E^B, \mathbf{R}^B)$ are primary entities.

The codomains of the mappings θ^{AB} and θ^{BA} are called ***primary direct reflections***, i.e.:

- for $\theta^{AB} = (A \rightarrow B) = (\theta_e^{AB} : E^A \rightarrow E^B , \theta_r^{AB} : R^A \rightarrow R^B)$ the primary direct reflection in the entity B is the codomain set

  $B_\psi^{AB} = \{ y_k , y_l , (y_k , y_l) \mid y_k , y_l \in E^B, (y_k , y_l) \in R^B ,$

  $\exists\{x_i, x_j, (x_i, x_j) \mid x_i, x_j \in E^A, (x_i, x_j) \in R^A, x_i \rightarrow y_k, x_j \rightarrow y_l, (x_i, x_j) \rightarrow (y_k , y_l)\}\};$
- for $\theta^{BA} = (B \rightarrow A) = (\theta_e^{BA} : E^B \rightarrow E^A , \theta_r^{BA} : R^B \rightarrow R^A)$ the primary direct reflection in the entity A is the codomain set

  $A_\psi^{AB} = \{ x_i, x_j, (x_i, x_j) \mid x_i, x_j \in E^A, (x_i, x_j) \in R^A,$

  $\exists\{y_k, y_l, (y_k,y_l) \mid y_k, y_l \in E^B, (y_k,y_l) \in R^B, y_k \rightarrow x_i, y_l \rightarrow x_j, (y_k,y_l) \rightarrow (x_i,x_j)\}\};$ ⌟

Definition 6b. Let $\Psi^{AB} = \{ \Theta^{AB}, \Theta^{BA} \}$ is an entire direct contact between $\mathfrak{A}$ and $\mathfrak{B}$, i.e.

$$\mathfrak{A} \underset{\Theta^{BA}}{\overset{\Theta^{AB}}{\rightleftarrows}} \mathfrak{B}$$

where $\mathfrak{A} = (\mathfrak{E}^A, \mathfrak{R}^A)$ and $\mathfrak{B} = (\mathfrak{E}^B, \mathfrak{R}^B)$ are entire entities.

The codomains of the mappings Θ^{AB} and Θ^{BA} are called ***entire direct reflections***, i.e.:

- for $\Theta^{AB} = (\mathfrak{A} \rightarrow \mathfrak{B}) = (\Theta_{\mathfrak{E}}^{AB} : \mathfrak{E}^A \rightarrow \mathfrak{E}^B, \Theta_{\mathfrak{R}}^{AB} : \mathfrak{R}^A \rightarrow \mathfrak{R}^B)$ the entire direct reflection in the entity $\mathfrak{B}$ is the codomain subentity

  $\mathfrak{B}_\Psi^{AB} = \{ Y_p , Y_q \mid Y_p , Y_q \in \mathfrak{E}^B , (Y_p , Y_q) \in \mathfrak{R}^B ,$

  $\exists\{X_m, X_n \mid X_m, X_n \in \mathfrak{E}^A, (X_1, X_2) \in \mathfrak{R}^A, X_m \rightarrow Y_p, X_n \rightarrow Y_q, (X_m , X_n) \rightarrow (Y_p, Y_q)\}\} ;$
- for $\Theta^{BA} = (\mathfrak{B} \rightarrow \mathfrak{A}) = (\Theta_{\mathfrak{E}}^{BA} : \mathfrak{E}^B \rightarrow \mathfrak{E}^A, \Theta_{\mathfrak{R}}^{BA} : \mathfrak{R}^B \rightarrow \mathfrak{R}^A)$ the entire direct reflection in the entity $\mathfrak{A}$ is the codomain subentity

  $\mathfrak{A}_\Psi^{AB} = \{ X_m , X_n \mid X_m , X_n \in \mathfrak{E}^A , (X_1 , X_2) \in \mathfrak{R}^A,$

  $\exists\{Y_p, Y_q \mid Y_p, Y_q \in \mathfrak{E}^B , (Y_p, Y_q) \in \mathfrak{R}^B, Y_p \rightarrow X_m, Y_q \rightarrow X_n, (Y_p, Y_q) \rightarrow (X_m, X_n)\}\}$ ⌟

3.2. Transitive reflections

Definition 7a. Let $\xi^{A(B)C}$ is a primary transitive contact between A and C through B, i.e.

$$A \xrightarrow[\theta^{AB}]{} B \xrightarrow[\theta^{BC}]{} C$$

where A= (E^A, $\mathbf{R}^A$), B = (E^B, $\mathbf{R}^B$) and C = (E^C, $\mathbf{R}^C$) are primary entities and $\xi^{A(B)C}$ is the composition of primary contacts ψ^{AB} and ψ^{BC}, i.e.:

$$\xi^{A(B)C} = \psi^{BC} \circ \psi^{AB} = \{ \psi_e{}^{BC} \circ \psi_e{}^{AB} , \psi_r{}^{BC} \circ \psi_r{}^{AB} \}$$

The codomain of the mapping $\theta^{BC} \circ \theta^{AB}$ is called ***primary transitive reflection***, i.e. for

$$\theta^{BC} \circ \theta^{AB} = (A \rightarrow B_\psi{}^{AB} \rightarrow C_\psi{}^{BC}) = (\theta_e{}^{BC} \circ \theta_e{}^{AB} : E^A \rightarrow E^B \rightarrow E^C, \theta_r{}^{BC} \circ \theta_r{}^{AB} : R^A \rightarrow R^B \rightarrow R^C)$$

the primary transitive reflection of A in the entity C through B is the codomain set

$$C_\xi{}^{A(B)C} = \{ z_m , z_n , (z_m , z_n) \mid z_m , z_n \in E^C, (z_m , z_n) \in R^C ,$$
$$\exists \{ x_i, x_j, (x_i, x_j), y_k, y_l, (y_k, y_l) \mid x_i, x_j \in E^A, (x_i, x_j) \in R^A, y_k, y_l \in E^B, (y_k, y_l) \in R^B,$$
$$x_i \rightarrow y_k, x_j \rightarrow y_l, (x_i, x_j) \rightarrow (y_k , y_l), y_k \rightarrow z_m, y_l \rightarrow z_n, (y_k, y_l) \rightarrow (z_m, z_n) \} \}_{\rightarrow}$$

Definition 7b. Let $\Xi^{A(B)C}$ is an entire transitive contact between $\mathfrak{A}$ and $\mathfrak{C}$ through $\mathfrak{B}$, i.e.

$$\mathfrak{A} \xrightarrow[\Theta^{AB}]{} \mathfrak{B} \xrightarrow[\Theta^{BC}]{} \mathfrak{C}$$

where $\mathfrak{A}$ = ($\mathfrak{E}^A$, $\mathfrak{R}^A$) , $\mathfrak{B}$ = ($\mathfrak{E}^B$, $\mathfrak{R}^B$) and $\mathfrak{C}$ = ($\mathfrak{E}^C$, $\mathfrak{R}^C$) are entire entities and $\Xi^{A(B)C}$ is the composition of entire contacts Ψ^{AB} and Ψ^{BC}, i.e.:

$$\Xi^{A(B)C} = \Psi^{BC}{}_{\rightarrow} \Psi^{AB} = \{ \Psi_{\mathfrak{E}}{}^{BC}{}_{\rightarrow} \Psi_{\mathfrak{E}}{}^{AB} , \Psi_{\mathfrak{R}}{}^{BC}{}_{\rightarrow} \Psi_{\mathfrak{R}}{}^{AB} \}_{\rightarrow}$$

The codomain of the mapping $\Theta^{BC}{}_{\rightarrow}\Theta^{AB}$ is called ***entire transitive reflection***, i.e. for

$$\Theta^{BC}{}_{\rightarrow}\Theta^{AB} = (\Theta_{\mathfrak{E}}{}^{BC} , \Theta_{\mathfrak{R}}{}^{BC})_{\rightarrow} (\Theta_{\mathfrak{E}}{}^{AB} , \Theta_{\mathfrak{R}}{}^{AB}) =$$
$$= (\Theta_{\mathfrak{E}}{}^{BC}{}_{\rightarrow} \Theta_{\mathfrak{E}}{}^{AB} : \mathfrak{E}^A \rightarrow \mathfrak{E}^B \rightarrow \mathfrak{E}^C , \Theta_{\mathfrak{R}}{}^{BC}{}_{\rightarrow} \Theta_{\mathfrak{R}}{}^{AB} : \mathfrak{R}^A \rightarrow \mathfrak{R}^B \rightarrow \mathfrak{R}^C)$$

the entire transitive reflection of $\mathfrak{A}$ in $\mathfrak{C}$ through $\mathfrak{B}$ is the codomain

$$\mathfrak{C}_\Xi{}^{A(B)C} = \mathfrak{A} \rightarrow \mathfrak{B}_\Psi{}^{AB} \rightarrow C_\Psi{}^{BC} = \{ Z_u, Z_v \mid Z_u, Z_v \in \mathfrak{E}^C, (Z_u, Z_v) \in \mathfrak{R}^C,$$
$$\exists \{ X_m, X_n, Y_p, Y_q \mid X_m, X_n \in \mathfrak{E}^A, (X_m, X_n) \in \mathfrak{R}^A, Y_p, Y_q \in \mathfrak{E}^B, (Y_p, Y_q) \in \mathfrak{R}^B,$$
$$X_m \rightarrow Y_p, X_n \rightarrow Y_q, (X_m, X_n) \rightarrow (Y_p, Y_q), Y_p \rightarrow Z_u , Y_q \rightarrow Z_v, (Y_p, Y_q) \rightarrow (Z_u , Z_v) \} \}_{\rightarrow}$$

3.3. Transitive self-reflections

Definition 8a. Let $\xi^{A(B)A}$ is a primary transitive self-contact of A with itself through B, i.e.

$$A \xrightarrow[\theta^{AB}]{} B \xrightarrow[\theta^{BA}]{} A$$

where A= (E^A, $\mathbf{R}^A$) and B = (E^B, $\mathbf{R}^B$) are primary entities and $\xi^{A(B)A}$ is the composition of primary contacts ψ^{AB} and ψ^{BA}, i.e.:

$$\xi^{A(B)A} = \psi^{BA} \circ \psi^{AB} = \{ \psi_e{}^{BA} \circ \psi_e{}^{AB} , \psi_r{}^{BA} \circ \psi_r{}^{AB} \}$$

The codomain of the mapping $\theta^{BA} \circ \theta^{AB}$ is called ***primary transitive self-reflection*** of A in itself through B, i.e. for

$$\theta^{BA} \circ \theta^{AB} = (A \rightarrow B_\psi{}^{AB} \rightarrow A_\psi{}^{BA}) = (\theta_e{}^{BA} \circ \theta_e{}^{AB} : E^A \rightarrow E^B \rightarrow E^A, \theta_r{}^{BA} \circ \theta_r{}^{AB} : R^A \rightarrow R^B \rightarrow R^A)$$

the primary transitive self-reflection of A in itself is the codomain set

$$A_\xi{}^{A(B)A} = \{ x_m , x_n , (x_m , x_n) \mid x_m , x_n \in E^A, (x_m , x_n) \in R^A ,$$
$$\exists \{ x_i, x_j, (x_i, x_j), y_k, y_l, (y_k, y_l) \mid x_i, x_j \in E^A, (x_i, x_j) \in R^A, y_k, y_l \in E^B, (y_k, y_l) \in R^B,$$
$$x_i \rightarrow y_k, x_j \rightarrow y_l, (x_i, x_j) \rightarrow (y_k , y_l), y_k \rightarrow x_m, y_l \rightarrow x_n, (y_k, y_l) \rightarrow (x_m, x_n) \} \}_{\rightarrow}$$

Definition 8b. Let $\Xi^{A(B)A}$ is an entire transitive self-contact of $\mathfrak{A}$ in itself through $\mathfrak{B}$, i.e.

$$\mathfrak{A} \xrightarrow[\Theta^{AB}]{} \mathfrak{B} \xrightarrow[\Theta^{BA}]{} \mathfrak{A}$$

where $\mathfrak{A} = (\mathfrak{E}^A, \mathfrak{R}^A)$ and $\mathfrak{B} = (\mathfrak{E}^B, \mathfrak{R}^B)$ are entire entities and $\Xi^{A(B)A}$ is the composition of entire contacts Ψ^{AB} and Ψ^{BA}, i.e.:

$$\Xi^{A(B)A} = \Psi^{BA} \rightarrow \Psi^{AB} = \{ \Psi_{\mathfrak{E}}{}^{BA} \rightarrow \Psi_{\mathfrak{E}}{}^{AB} , \Psi_{\mathfrak{R}}{}^{BA} \rightarrow \Psi_{\mathfrak{R}}{}^{AB} \}$$

The codomain of the mapping $\Theta^{BA} \rightarrow \Theta^{AB}$ is called ***entire transitive self-reflection***, i.e. for

$$\Theta^{BA} \rightarrow \Theta^{AB} = (\Theta_{\mathfrak{E}}{}^{BA} , \Theta_{\mathfrak{R}}{}^{BA}) \rightarrow (\Theta_{\mathfrak{E}}{}^{AB} , \Theta_{\mathfrak{R}}{}^{AB}) =$$

$$= (\Theta_{\mathfrak{E}}{}^{BA} \rightarrow \Theta_{\mathfrak{E}}{}^{AB} : \mathfrak{E}^A \rightarrow \mathfrak{E}^B \rightarrow \mathfrak{E}^A , \Theta_{\mathfrak{R}}{}^{BA} \rightarrow \Theta_{\mathfrak{R}}{}^{AB} : \mathfrak{R}^A \rightarrow \mathfrak{R}^B \rightarrow \mathfrak{R}^A)$$

the entire transitive self-reflection of $\mathfrak{A}$ in itself through $\mathfrak{B}$ is the codomain

$$\mathfrak{A}_{\Xi}{}^{A(B)A} = \mathfrak{A} \rightarrow \mathfrak{B}_{\Psi}{}^{AB} \rightarrow \mathfrak{A}_{\Psi}{}^{BA} = \{ X_u , X_v \mid X_u , X_v \in \mathfrak{E}^A , (X_u , X_v) \in \mathfrak{R}^A ,$$
$$\exists\{X_m, X_n, Y_p, Y_q \mid X_m, X_n \in \mathfrak{E}^A, (X_m, X_n) \in \mathfrak{R}^A, Y_p, Y_q \in \mathfrak{E}^B, (Y_p, Y_q) \in \mathfrak{R}^B,$$
$$X_m \rightarrow Y_p, X_n \rightarrow Y_q, (X_m, X_n) \rightarrow (Y_p, Y_q), Y_p \rightarrow X_u , Y_q \rightarrow X_v, (Y_p, Y_q) \rightarrow (X_u, X_v)\}\} \dashv$$

3.4. Interactive reflections

Definition 9a. Let the δ^{AB} is a primary interaction between A and B i.e. δ^{AB} is a composition of primary (direct and/or transitive) contacts which begins from A, i.e.:

$$\delta^{AB} = \{ \delta_1 \circ \delta_2 \circ \ldots \circ \delta_h \mid \delta_1 \in \{\psi_i{}^{AB}, \xi_k{}^{A(X)B} \mid i=1,2,\ldots,n; k=1,2,\ldots,p \},$$
$$\delta_t, t=2,\ldots,h \in \{\psi_i{}^{AB}, \psi_j{}^{BA}, \xi_k{}^{A(X)B}, \xi_l{}^{B(X)A} \mid i=1,2,\ldots,n; j=1,2,\ldots,m; k=1,2,\ldots,p; l=1,2,\ldots,q; X: \text{arbitrary}\} \}$$

The codomain of the mapping $\delta^{AB} = \delta_1 \circ \delta_2 \circ \ldots \circ \delta_h$ is called:

- ***primary interactive reflection*** iff it is belong to B;
- ***primary interactive self-reflection*** iff it is belong to A. $\dashv$

Definition 9b. Let the Δ^{AB} is an entire interaction between $\mathfrak{A}$ and $\mathfrak{B}$ i.e. Δ^{AB} is a composition of entire (direct and/or transitive) contacts which begins from $\mathfrak{A}$, i.e.:

$$\Delta^{AB} = \{ \Delta_1 \rightarrow \Delta_2 \rightarrow \ldots \rightarrow \Delta_h \mid \Delta_t , t=1,2,\ldots,h \in$$
$$\in \{\Psi_i{}^{AB}, \Psi_j{}^{AB}, \Xi_k{}^{A(X)B}, \Xi_l{}^{B(X)A} \mid i=1,2,\ldots,n; j=1,2,\ldots,m; k=1,2,\ldots,p ; l=1,2,\ldots,q ; X : \text{arbitrary}\} .$$

The codomain of the mapping $\Delta^{AB} = \Delta_1 \rightarrow \Delta_2 \rightarrow \ldots \rightarrow \Delta_h$ is called:

- ***entire interactive reflection*** iff it is belong to $\mathfrak{B}$;
- ***entire interactive self-reflection*** iff it is belong to $\mathfrak{A}$. $\dashv$

3.5. Partial and complete reflections

Definition 10a. Let $B_{\psi}{}^{AB}$ is a primary direct reflection of A in B, where A= (E^A, $\mathbf{R}^A$) and B = (E^B, $\mathbf{R}^B$) are primary entities.

$B_{\psi}{}^{AB}$ is called :

- ***primary partial direct reflection*** iff
  $\forall\{y_k, y_l, (y_k, y_l) \mid y_k, y_l \in E^B, (y_k, y_l) \in R^B\} \rightarrow$
  $\rightarrow \exists\{x_i, x_j, (x_i, x_j) \mid x_i, x_j \in E^A, (x_i, x_j) \in R^A, x_i \rightarrow y_k, x_j \rightarrow y_l, (x_i, x_j) \rightarrow (y_k, y_l)\}\};$
- ***primary complete direct reflection*** iff
  $\forall\{ x_i, x_j, (x_i, x_j) \mid x_i, x_j \in E^A, (x_i, x_j) \in R^A\} \rightarrow$
  $\rightarrow \exists\{y_k, y_l, (y_k, y_l) \mid y_k, y_l \in E^B, (y_k, y_l) \in R^B, x_i \rightarrow y_k, x_j \rightarrow y_l, (x_i, x_j) \rightarrow (y_k, y_l)\}\} \dashv$

Definition 10b. Let $\mathfrak{B}_{\Psi}{}^{AB}$ is an entire direct reflection of $\mathfrak{A}$ in $\mathfrak{B}$, where $\mathfrak{A} = (\mathfrak{E}^A, \mathfrak{R}^A)$ and $\mathfrak{B} = (\mathfrak{E}^B, \mathfrak{R}^B)$ are entire entities.

$\mathfrak{B}_{\Psi}{}^{AB}$ is called :

- ***entire partial direct reflection*** iff

$\forall\{ Y_p , Y_q \mid Y_p , Y_q \in \mathfrak{E}^B , (Y_p , Y_q) \in \mathfrak{R}^B \} \rightarrow$

$\rightarrow \exists\{X_m, X_n \mid X_m, X_n \in \mathfrak{E}^A, (X_1, X_2) \in \mathfrak{R}^A, X_m \rightarrow Y_p, X_n \rightarrow Y_q, (X_m, X_n) \rightarrow (Y_p, Y_q)\}\}$

- ***entire complete direct reflection*** iff

$\forall\{ X_m , X_n \mid X_m , X_n \in \mathfrak{E}^A , (X_1 , X_2) \in \mathfrak{R}^A\} \rightarrow$

$\rightarrow \exists\{Y_p, Y_q \mid Y_p, Y_q \in \mathfrak{E}^B , (Y_p, Y_q) \in \mathfrak{R}^B, X_m \rightarrow Y_p, X_n \rightarrow Y_q, (X_m, X_n) \rightarrow (Y_p, Y_q)\}\}$ ⊿

Definition 11a. Let $C_{\xi}^{A(B)C}$ is a primary transitive reflection of A in the entity C through B, where A= (E^A, $\mathbf{R}^A$), B = (E^B, $\mathbf{R}^B$) and C = (E^C, $\mathbf{R}^C$) are primary entities and $\xi^{A(B)C}$ is the composition of primary contacts ψ^{AB} and ψ^{BC}.

$C_{\xi}^{A(B)C}$ is called :

- ***primary partial transitive reflection*** iff any of B_{ψ}^{AB} and C_{ψ}^{BC} is a partial direct reflection.
- ***primary complete transitive reflection*** iff both B_{ψ}^{AB} and C_{ψ}^{BC} are complete direct reflections ⊿

Definition 11b. Let $\mathfrak{C}_{\Xi}^{A(B)C}$ is an entire transitive reflection of $\mathfrak{A}$ in $\mathfrak{C}$ through $\mathfrak{B}$, where $\mathfrak{A}$ = ($\mathfrak{E}^A$, $\mathfrak{R}^A$), $\mathfrak{B}$ = ($\mathfrak{E}^B$, $\mathfrak{R}^B$) and $\mathfrak{C}$ = ($\mathfrak{E}^C$, $\mathfrak{R}^C$) are entire entities and $\Xi^{A(B)C}$ is the composition of entire contacts Ψ^{AB} and Ψ^{BC}.

$C_{\Xi}^{A(B)C}$ is called :

- ***entire partial transitive reflection*** iff any of $\mathfrak{B}_{\Psi}^{AB}$ and C_{Ψ}^{BC} is a partial reflection.
- ***entire complete transitive reflection*** iff both $\mathfrak{B}_{\Psi}^{AB}$ and C_{Ψ}^{BC} are complete reflections ⊿

Definition 12a. Let $A_{\xi}^{A(B)A}$ is a primary transitive self-reflection of A in itself through B, where A= (E^A, $\mathbf{R}^A$) and B = (E^B, $\mathbf{R}^B$) are primary entities and $\xi^{A(B)A}$ is the composition of primary contacts ψ^{AB} and ψ^{BA}.

$A_{\xi}^{A(B)A}$ is called :

- ***primary partial transitive self-reflection*** iff any of B_{ψ}^{AB} and A_{ψ}^{BA} is a partial direct reflection;
- ***primary complete transitive self-reflection*** iff both B_{ψ}^{AB} and A_{ψ}^{BA} are complete direct reflections. ⊿

Definition 12b. Let $\mathfrak{A}_{\Xi}^{A(B)A}$ is an entire transitive self-reflection of $\mathfrak{A}$ in itself through $\mathfrak{B}$, where $\mathfrak{A}$ = ($\mathfrak{E}^A$, $\mathfrak{R}^A$) and $\mathfrak{B}$ = ($\mathfrak{E}^B$, $\mathfrak{R}^B$) are entire entities and $\Xi^{A(B)A}$ is the composition of entire contacts Ψ^{AB} and Ψ^{BA}.

$\mathfrak{A}_{\Xi}^{A(B)A}$ is called :

- ***entire partial transitive self-reflection*** iff any of $\mathfrak{B}_{\psi}^{AB}$ and $\mathfrak{A}_{\psi}^{BA}$ is a partial direct reflection;
- ***entire complete transitive self-reflection*** iff both $\mathfrak{B}_{\psi}^{AB}$ and $\mathfrak{A}_{\psi}^{BA}$ are complete direct reflections. ⊿

Definition 13a. Let $\delta^{AB} = \delta_1 \circ \delta_2 \circ \ldots \circ \delta_h$ is primary interactive reflection.

δ^{AB} is called :

- ***primary partial interactive reflection*** iff any of δ_t, t=1,2,…,h is a partial reflection;
- ***primary complete interactive reflection*** iff all of δ_t, t=1,2,…,h are complete reflections. ⊿

Definition 13b. Let Δ^{AB} is entire interactive reflection.

Δ^{AB} is called :

- ***entire partial interactive reflection*** iff any of Δ_t, t=1,2,…,h is a partial reflection;
- ***entire complete interactive reflection*** iff all of Δ_t, t=1,2,…,h are complete reflections. ⊿

Definition 14a. Let δ^{AB} is primary interactive self-reflection.

δ^{AB} is called :

- ***primary partial interactive self-reflection*** iff any of δ_t, t=1,2,…,h is a partial reflection;
- ***primary complete interactive self-reflection*** iff all of δ_t, t=1,2,…,h are complete reflections. ⊿

Definition 14b. Let Δ^{AB} is entire interactive self-reflection.

Δ^{AB} is called :

- ***entire partial interactive self-reflection*** iff any of Δ_t , t=1,2,…,h is a partial reflection;
- ***entire complete interactive self-reflection*** iff all of Δ_t, t=1,2,…,h are complete reflections. ⊿

Propositions

Proposition 1. The complete reflection contains all partial reflections.

Proposition 2. The complete self-reflection of an entity coincides (is identical) with it.

Proposition 3. The complete self-reflection is impossible.

4. Information

4.1. The Reflection Evidence and the Information

The real world contains unlimited number of entities. When an entity contact another there exist great possibility to join third entity in this process. It is clear; the third entity may contact and reflect each of others as well as the entire process of realisation of the contact between them.

The reflection could not be detected by the entity that contains it. This is dialectical behaviour of the reflection - this is only an internal change after interaction.

The process of realisation of the contact is a specific (temporal) forming relationship between entities. During the process of establishing the contact the entities form new (temporal) entity which in the same moment may be reflected bay the third entity.

The third entity which interacts with the two others and reflects the process of contact between them is called "***reflection evidence"*** for the correspondence between internal changes in the second entity after the contact with the first one which is theirs origin.

Because of similarity of primary and entire cases in the next text only the entire definitions will be presented.

Definition 15. Let $\mathfrak{A} = (\mathfrak{E}^A, \mathfrak{R}^A)$, $\mathfrak{B} = (\mathfrak{E}^B, \mathfrak{R}^B)$ and $\mathfrak{C} = (\mathfrak{E}^C, \mathfrak{R}^C)$ are entire entities and

$$\Psi^{AB} = \{ \Theta^{AB}, \Theta^{BA} \},\ \Psi^{BC} = \{ \Theta^{BC}, \Theta^{CB} \} \text{ and } \Psi^{AC} = \{ \Theta^{AC}, \Theta^{CA} \}$$

are entire contacts, and

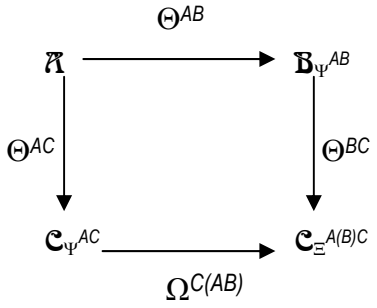
$$\mathfrak{B}_{\Psi}{}^{AB} = \{ Y_p , Y_q \mid Y_p , Y_q \in \mathfrak{E}^B , (Y_p , Y_q) \in \mathfrak{R}^B ,$$
$$\exists \{X_m, X_n \mid X_m, X_n \in \mathfrak{E}^A, (X_1, X_2) \in \mathfrak{R}^A, X_m \to Y_p, X_n \to Y_q, (X_m , X_n) \to (Y_p, Y_q)\}\}$$

is a reflection in $\mathfrak{B}$ of $\mathfrak{A}$.

Let the entire transitive reflection of $\mathfrak{A}$ in $\mathfrak{C}$ through $\mathfrak{B}$ is the codomain

$$\mathfrak{C}_{\Xi}{}^{A(B)C} = \mathfrak{A} \to \mathfrak{B}_{\Psi}{}^{AB} \to \mathfrak{C}_{\Psi}{}^{BC} = \{ Z_u, Z_v \mid Z_u, Z_v \in \mathfrak{E}^C, (Z_u, Z_v) \in \mathfrak{R}^C,$$
$$\exists\{X_m, X_n, Y_p, Y_q \mid X_m, X_n \in \mathfrak{E}^A, (X_m, X_n) \in \mathfrak{R}^A, Y_p, Y_q \in \mathfrak{E}^B, (Y_p, Y_q) \in \mathfrak{R}^B,$$
$$X_m \to Y_p, X_n \to Y_q, (X_m, X_n) \to (Y_p, Y_q), Y_p \to Z_u , Y_q \to Z_v, (Y_p, Y_q) \to (Z_u , Z_v)\}\} \text{ .}$$

If the diagram:



(1)

Is commutative than for the contact Ψ^{AB} in the case for the mapping Θ^{AB} and reflection $\mathfrak{B}_{\Psi}{}^{AB}$:

- entity $\mathfrak{A}$ is called ***reflection source***
- entity $\mathfrak{B}$ is called ***reflection recipient***
- entity $\mathfrak{C}$ is called ***reflection evidence***
- the mapping $\Omega^{C(AB)} = \mathfrak{C}_{\Psi}{}^{AC} \to \mathfrak{C}_{\Xi}{}^{A(B)C}$ is called ***information evidence***
- $\mathfrak{B}_{\Psi}{}^{AB}$ is called ***information*** in $\mathfrak{B}$ for $\mathfrak{A}$.

Proposition 4. Every reflection may be considered as information iff there exists reflection evidence in the sense of commutative diagram (1).

The diagram (1) shows a very important case of the real world - simultaneous contacts of three entities. Every one of them may be source, recipient and evidence in the same time. It is possible but not obligatory that the reverse diagrams may exist and may be commutative. There exist six cases of diagrams which represent the simultaneous contacts of three entities. We may denote them by the relation "(source, recipient : evidence)". So, the entities $\mathfrak{A}$, $\mathfrak{B}$ and $\mathfrak{C}$ may be in the next six reflection relations:

1. ($\mathfrak{A}$, $\mathfrak{B}$: $\mathfrak{C}$) , 2. ($\mathfrak{B}$, $\mathfrak{C}$: $\mathfrak{A}$) , 3. ($\mathfrak{C}$, $\mathfrak{A}$: $\mathfrak{B}$) , 4. ($\mathfrak{A}$, $\mathfrak{C}$: $\mathfrak{B}$) , 5. ($\mathfrak{C}$, $\mathfrak{B}$: $\mathfrak{A}$), 6. ($\mathfrak{B}$, $\mathfrak{A}$: $\mathfrak{C}$) .

All reflection relations are equivalent from point of view of the interrelations between reflection source, reflection recipient and reflection evidence. Because of this we will discuss only the case shown on the diagram (1) above.

For practical needs, it is more convenient to follow the next consideration.

The reflection in the recipient represents both the relationships and the subentities of the source. From other point of view, the relationships build up and present the entities. Because of this, the reflected relationships are the essence of the reflection. In other words, if there exists reflection evidence than the reflection of the forming relationship may be considered as "information" for reflected entity.

So, in the sense that an evidence exists to point what relationship (between what entities) is reflected and where it is done, we may say: "***The information is reflected relationship"*** .

4.2. General Structure of Information

The entities and theirs relationships form space hierarchies. Every entity contains all entities of its low levels. In this sense we can say that every relationship contains in itself the relationships of low levels of the entity.

As reflected relationship, the Information is reflected space hierarchy of all relationships of this one. From this point of view we can say the general structure of information reflects general structure of real relationships.

The information of one level contains space hierarchy of information of low levels. So, here, the main idea is:

The General Structure of Information is a Space Hierarchy.

4.3. Types of the Information

The information is a result from the interaction. It is a kind of the reflection. Therefore, the information has the corresponding properties. Especially, we have primary interaction, secondary (transitive) interaction, self-interaction etc. This way, there exist corresponding types of the reflection and the main types information are:

- direct information;
- transitive information;
- transitive self-information;
- interactive direct information;
- interactive transitive information;
- interactive transitive self-information.

From other point of view, the interaction may be provided on different levels of the structure of the entities. Therefore, we may talk about primary (one level) or entire (all levels) interaction and reflection. So, the corresponded information may be primary or entire information.

4.4. Information Elements and Information Memory

Definition 16. The single information triple

***i* = (source, recipient : evidence)**

defines concrete ***(single) information element .***

For instance, such element is $\mathfrak{B}_{\Psi^{AB}}$ in the definition 15 and diagram (1). ⊔

Definition 17. The ***(information) memory*** of the entity is the set of all information elements which are reflected in the entity.

It is clear, from point of view of the period of existing of the corresponded reflections in the entity, the memory may be more *temporal* or more *permanent* . ⎵

4.5. Information Spaces

The information elements are real reflections in the entities and they exist in the real world. This means that for every contact or interaction as well as for every single entity or set of entities may exist corresponded sets of information elements.

Definition 18. The set of information elements which is defined by single source and single recipient is called ***single information space***. ⎵

Definition 19. The set of information elements which is defined by many sources and single recipient is called ***common information space***. ⎵

4.6. Information Environment

Definition 20. The set of information elements which are defined by single source and many recipients is called ***single information environment*** which contains many information spaces. ⎵

Definition 21. The set of information elements which are defined by many sources and many recipients is called ***common information environment***. ⎵

Conclusion

The translation of the philosophical theory into the formal one is a good approach for verification of the scientific ideas. The concepts of Entity, Interaction, Reflection, and Information of the General Information Theory were presented formally in this paper. The primary and entire definitions given above are a step for building the formal part of the General Information Theory. Together with the philosophical explanations, it gives us a useful tool for investigation of the information phenomena in the real world.

This work is partially financed by project ITHEA-XXI of FOI Institute of Information Theories and Applications.

Authors are very grateful to all participants in the fruitful discussions at KDS and ITA International Conferences as well as to all members of the International Workgroup on Data Base Intellectualisation (IWDBI) for supporting the advance of the General Information Theory.

Bibliography

[Markov, 1988] Kr.Markov. From the past to the future of the definition of the concept of Information. Proceedings "PROGRAMMING '88", BAS, Varna 1988, p.150. (In Bulgarian).

[Markov et al, 1993] Kr.Markov, Kr.Ivanova, I.Mitov. Basic Concepts of a General Information Theory. IJ "Information Theories and Applications". FOI ITHEA, Sofia, 1993, Vol.1,No.10, pp.3-10

Author information

Krassimir Markov - Institute of Information Theories and Applications FOI ITHEA; Institute of Mathematics and Informatics, BAS; P.O.Box: 775, Sofia-1090, Bulgaria; e-mail: foi@nlcv.net

Krassimira Ivanova - Institute of Mathematics and Informatics, BAS, Acad.G.Bonthev St., bl.8, Sofia-1113, Bulgaria; e-mail: foi@nlcv.net

Ilia Mitov - Institute of Information Theories and Applications FOI ITHEA; P.O.Box: 775, Sofia-1090, Bulgaria; e-mail: foi@nlcv.net

THE INFOS

Kr. Markov, Kr. Ivanova, I. Mitov

***Abstract**: The concept "Infos" from the General Information Theory is defined in the paper. For this purpose are introduced the concepts "Information Witness", "Activity", "Information expectation", "Resolving the information expectation".*

Introduction

The General Information Theory (GIT) [Markov et al., 1993] needs a basic concept for representing the information agents. During the last twenty years, the role of such concept has been played by the concept "Information Subject". Due to analogy with the human, the concept "Information Subject" is not so general as we need. Because of this in the paper we propose new concept and introduce it formally. The establishment of the new concept become as reason to revise the definitions of some of the main concepts of the GIT. The using of the concept "Activity" compels us to redefine the concepts of "Information Witness".

This paper is closely connected to the [Markov et al., 2003] and uses the concepts defined in it without new introducing in the text.

1. The Activity

Every forming relationship as well as every relationship unites the entities and this way it satisfies some theirs possibilities for building the relationship by establishing the contact. In other words, for creating the forming relationship we need:

- entities, from which the new entity is able to built;
- possibilities of the entities for establishing the contact by satisfying of which the forming relationship may be established.

The forming relationship is the aggregate of the satisfied possibilities for establishing the contact.

It is clear that after establishing the relationship we may have any of two cases:

- all possibilities of the entities for establishing the contact are satisfied by such possibilities of other entities;
- there are any free possibilities after finishing the establishment of the new relationship - on the low levels of the entity or, if it is a new entity, on the level of the whole entity. Disintegration of the entity may generate any possibilities too.

In the second case, the entity has ***"free valency"*** which needs to be satisfied by corresponded contacts with other entities. We may say, the entity has ***activity*** generated by the free possibilities for establishing the contacts with the entities from the environment.

The process of interaction is satisfying the possibilities for contact of the entities. From point of view of the entity, the interaction may be external or internal.

During the interaction given entity may be destroyed partially or entirely and only several but not all parts of the destroyed entity may be integrated in the new entity. This means that there exist both constructive and destructive processes in the process of interaction between entities. The determination of the type of the interaction depends on the point of view of given entity. The interaction dialectically contains constructive and destructive sub-processes.

If the entity is a complex, it is possible for it to have an opportunity of self-reflection. In such case, it is able to reflect any reflection, which has been already reflected in it. In this case, because of the new internal changes (self-reflection) the entity may obtain any new ***"secondary activity"***.

The secondary activity is closely connected to the structural level of the entity, which correspond to the level of the self-reflection. This way the secondary activity may be satisfied by internal or external entity from point of view of the given entity. In other words, ***the resolving*** of the secondary activity may be ***internal or external***.

Definition 1. Let the internal connected couple $\mathfrak{E} = (\mathfrak{E}^E, \mathfrak{R}^E)$ is an entire entity.

The relation $\mathfrak{V} = (X_i , \varnothing)$ where $X_i \in \mathfrak{E}^E$, $\varnothing \in \mathfrak{E}^E$, $(X_i, \varnothing) \in \mathfrak{R}^E$, i = 1,2,..,n, is called ***free valency*** of $\mathfrak{E}$.

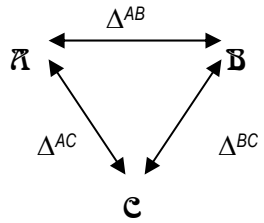
The set of relations $\mathfrak{P} = \{ \mathfrak{V}_i \mid i = 1,2,..,n \} = \{ (X_i, \varnothing) \mid X_i \in \mathfrak{E}^E, \varnothing \in \mathfrak{E}^E, (X_i, \varnothing) \in \mathfrak{R}^E, i = 1,2,..,n \}$ is called ***activity*** or ***expectation for contact*** of $\mathfrak{E}$.

2. The Information Witness

The interaction between two entities is a specific theirs relationship. If a third entity interacts with the two others and reflects ***the whole process*** of interaction between them it became not only reflection evidence. It reflects the information about them, which has been generated during the interaction. So, the third entity became an ***"information witness"*** of the interaction between two entities as well as of the existence of the information about the first entity in the second entity and vice versa.

Definition 2. Let $\mathfrak{A} = (\mathfrak{E}^A, \mathfrak{R}^A)$, $\mathfrak{B} = (\mathfrak{E}^B, \mathfrak{R}^B)$ and $\mathfrak{C} = (\mathfrak{E}^C, \mathfrak{R}^C)$ are entire entities and the entire interactions between them are Δ^{AB} , Δ^{AC} and Δ^{BC} .

The diagram of ***complex interaction*** between $\mathfrak{A}$, $\mathfrak{B}$ and $\mathfrak{C}$ is:



(1)

If, after this complex interaction, in the third entity $\mathfrak{C}$ there exists a relation between the reflections of the first entity $\mathfrak{A}$ and reflections of the internal changes in the second entity $\mathfrak{B}$ and vice versa, than for the interaction Δ^{AB} the third entity $\mathfrak{C}$ is an "**Information Witness**" (**IW**) of existence of the information about the first entity $\mathfrak{A}$ in the second entity $\mathfrak{B}$ and vice versa. ⌟

Definition 3. The set $\Omega^{C(AB)} = \{ \omega_i^{C(AB)} \mid i = 1,2,\ldots,n \}$ is called ***complex information evidence*** for existing the information in $\mathfrak{B}$ for $\mathfrak{A}$ and vice versa - the set $\Omega^{C(BA)} = \{ \omega_j^{C(BA)} \mid j = 1,2,\ldots,n \}$ is called ***complex information evidence*** for existing the information in $\mathfrak{A}$ for $\mathfrak{B}$.

The couple $\mathbf{IR}_C^{AB} = (\Omega^{C(AB)} , \Omega^{C(BA)})$ is called ***information relationship*** between the entities $\mathfrak{A}$ and $\mathfrak{B}$. ⌟

As the information witness is more complex entity so the information relationship may be more complex.

3. The Information Expectation

During the establishment of the information relationship it is possible to be generated any secondary free activity (possibilities on the low levels of the entity or on the level of the whole entity) which needs to be satisfied by corresponded contacts with other entities.

The secondary activity in the information witness generated by the information relationship is called ***"information activity"***.

Definition 4. Let $\mathfrak{A} = (\mathfrak{E}^A, \mathfrak{R}^A)$, $\mathfrak{B} = (\mathfrak{E}^B, \mathfrak{R}^B)$ and $\mathfrak{C} = (\mathfrak{E}^C, \mathfrak{R}^C)$ are entire entities.

Let $\mathbf{IR}_C^{AB} = (\Omega^{C(AB)} , \Omega^{C(BA)})$ is an information relationship in $\mathfrak{C}$ for $\mathfrak{A}$ and $\mathfrak{B}$.

The relation $\mathfrak{V}_C^{AB} = (X, \varnothing)$ where $X \in \mathfrak{E}^C$, $\varnothing \in \mathfrak{E}^C$, $(X, \varnothing) \in \mathbf{IR}_C^{AB} \in \mathfrak{R}^C$ is called (secondary) ***free information valency*** of $\mathfrak{C}$ ⌟

The set $\mathcal{Q}_C^{AB} = \{ \mathbf{V}_{i,C}^{AB} \mid i = 1,2,..,n \} = \{ (X_i , \varnothing) \mid X_i \in \mathfrak{E}^C, \varnothing \in \mathfrak{E}^C , (X_i, \varnothing) \in \mathbf{IR}_C^{AB} \in \mathfrak{R}^C , i = 1,2,..,n \}$

is called ***information activity*** or ***information expectation*** of $\mathfrak{C}$ based on the $\mathbf{IR}_C^{AB}$.

Proposition 1. The information expectation is a part of the activity of the Information witnes.

4. Resolving the Information Expectation

Because of the existing of the information expectation, i.e. the existing of the secondary information activity, the Information Witness "expects" to combine the information valencies with any others.

Definition 5. The combinig the valencies of the information expectation with some others is called ***resolving the information expectation***.

Definition 6. Let "n" is the number of free valencies in the information expectation. After the contact some of them are combined as well as the others are not. The new valencies, which are generated by the contact, do not belong to the information expectation before contact. They may form new information expectation but the basis for our reasoning will be the starting information expectation.

The normalised by "*n*" number D' of the not combined valencies is called ***degree of discrepancy*** (**D**) of the incoming reflection to the information expectation, i.e.

$$\mathbf{D} = \frac{D'}{n}$$

The normalised by "*n*" number *C*' of the combined valencies is called ***degree of combining*** (**C**) of the incoming reflection to the information expectation, i.e.

$$\mathbf{C} = \frac{C'}{n}$$

Proposition 2: There exists the equation : $\mathbf{C + D = 1.}$

From point of view of given expectation for contact the number of free valencies is fixed. After the contact, as a result of reflection, some of the free valencies of the entity may be combined with any new (internal or external) valencies. Of course, new free valencies may occur. The number "*n*" varies in the process of interaction. Every contact may change it.

The more valencies of the information expectation have been resolved, the more qualitative is the incoming information and vice versa.

Definition 7. The difference **A** between normalised number **C** of resolved valencies and normalised number **D** of not resolved valencies of the information expectation is called ***adequacy of the reflection to the information expectation***, i.e.

$$\mathbf{A = C - D}$$

Proposition 3. The values of adequacy A are in the interval [-1,1].

5. The Infos

On given level of complexity of the entities a new quality becomes - the existing self-reflection and internal activity based on the main possibilities for contact of the sub-entities as well as on the new (secondary) possibilities created after internal self-reflection.

The internal activity may be resolved by:

- the internal changes which lead to partial internal disintegration of the subentities and theirs a posterior internal integration in the new structures;
- the external influence on the environment.

The internal changes may lead to removing of some subentities if they have no possibilities for integration with the others, i.e. if they have no free valencies to be resolved in the process of integration.

The external influence is the most important. The impact on the entities around the entity is the way to resolve its activity. The destroying of the external entities and including the appropriate theirs parts in itself is the main means to exist and satisfy the free valencies.

One special kind of activity is the information one. The secondary activity need to be resolved by relevant to the information valencies corresponded (information) valencies. So, not every entity may be used for resolving the secondary activity.

This way, the entity needs a special kind of (information) contacts and (information) interaction for resolving the information activity.

Definition 8. The entity, which has:
- ***(primary) activity*** for external interaction;
- possibility for ***reflection***, i.e. possibility for collecting the information;
- possibility for ***self-reflection***, i.e. possibility for generating "secondary information";
- ***information expectation*** i.e. the (secondary) information activity for internal or external contact for resolving it

is called ***Infos*** . □

Definition 9. The resolving of the information activity is ***the goal*** of the Infos. □

This goal may be achieved by the establishment and providing (information) contacts and interaction.

Conclusion

In this paper we introduce the concept **"Infos"**. It is basic for the General Information Theory. Its definition is only the starting point for further investigations and building the **Infos Theory**.

The variety of types of Infoses in the real world need to be investigated and classified in the future research. At the first step, we may propose that may be there exist at least two main types of Infoses:

- ***infogens*** - the natural creatures;
- ***infotrons*** - the artificial creatures.

Also, the Infos Theory needs to give answers to many other very important questions, such as:

- What is the nature of the activity of the Infos?
- What is the difference between the living level of the Infos and the non-living one?
- Is it true that the boundary between non-living and living entities is self-reflection and internal activity for satisfying the secondary (information) possibilities for internal or external contact?
- Ect.

Bibliography

[Markov et al, 1993] Kr.Markov, Kr.Ivanova, I.Mitov. Basic Concepts of a General Information Theory. IJ "Information Theories and Applications". FOI ITHEA, Sofia, 1993, Vol.1, No.10, pp.3-10

[Markov et al, 2003] Kr.Markov, Kr.Ivanova, I.Mitov. The Information. (In the same proceedings).

Author information

Krassimir Markov - ITHEA - FOI Institute of Information Theories and Applications; Institute of Mathematics and Informatics, BAS; P.O.Box: 775, Sofia-1090, Bulgaria; e-mail: foi@nlcv.net

Krassimira Ivanova - Institute of Mathematics and Informatics, BAS, Acad.G.Bonthev St., bl.8, Sofia-1113, Bulgaria; e-mail: foi@nlcv.net

Ilia Mitov - ITHEA - FOI Institute of Information Theories and Applications, P.O.Box: 775, Sofia-1090, Bulgaria; e-mail: foi@nlcv.net

БАЗОВАЯ КЛАССИФИКАЦИЯ ИМЕН В ПРОГРАММИРОВАНИИ

В. Ю. Винник

***Аннотация**: Рассмотрены некоторые характерные особенности отношений именования и понятия имени в программировании. Предложена классификация имен, основанная на различии способов связи имен с их значениями и различии способов интерпретации имен в составе выражений.*

Введение

В основе любого языка лежит тот факт, что знаки языка (имена) связаны с внеязыковыми сущностями посредством отношений обозначения (*именования*). Отношение именование есть отношение между именем и обозначаемым им объектом (*денотатом*). Анализу отношений именования в естественных языках и в искусственном языке математики посвящено множество работ, среди которых основополагающее значение имели исследования Г.Фреге [1], Р.Карнапа [2] и Л.Витгенштейна [3]. Проблема отношений именования для языков программирования до недавнего времени не исследовалась. Между тем, именно феномен именования составляет нетрадиционную сущность программирования и резко отличает его от всех предшествующих систем [4].

В настоящей работе ставится цель дать в общих чертах классификацию имен в программировании, понимая имена в широком смысле, включающем не только имена переменных, но также ключевые слова языка программирования, знаки констант и вообще любые знаковые конструкции, фигурирующие в программных текстах.

При анализе предмета будем руководствоваться парадигмой конвенционализма А.Пуанкаре [5], наследующей основные положения номиналистского направления в философии математики. В соответствии с нею не будем допускать приписывания в рассуждениях самостоятельного онтологического статуса абстрактным умозрительным объектам, не будем рассуждать о них как о реально существующих объектах. Имена абстракций будем считать ничего не обозначающими фигурами языка (ср. с установкой Д.Гильберта [6] на элиминацию фиктивных терминов). Таким образом, мы не принимаем платонистский подход к математике, согласно которому математика изучает свойства и отношения реально существующих «вещей» особого рода — математических абстрактных объектов, а законы математики суть *объективные* законы мира этих «вещей». Для математического платонизма характерно положение, что абстрактному термину как имени соответствует определенный денотат — реально существующий абстрактный объект, и реально присущие последнему свойства определяют способы правильного употребления данного имени в высказываниях. В данной работе, напротив, следуя номинализму, полагаем, что такому имени вообще не соответствует никакой денотат, а законы математики суть законы математического языка. Абстрактному имени соответствует (вместо денотата) совокупность языковых соглашений (*конвенций*) об употреблении данного имени, или правил корректного обращения с ним. (Впрочем, учитывая укорененность платонистского подхода, будем привлекать его в качестве вспомогательного средства, не забывая при этом о его условности).

Классификация имен

Анализ явлений именования в контексте программирования с точки зрения парадигмы математического конвенционализма позволяет провести адекватную классификацию имен, разделив имена на *глобально-конвенциональные* (гк-имена, гки), *локально-конвенциональные* (лк-имена, лки) и *неконвенциональные* (нк-имена, нки).

Гк-имена — это имена, связь которых с денотатами (в платонистской трактовке) или правила интерпретации которых (в номиналистской трактовке) определяется общими соглашениями, которые составляют неотъемлемую часть определения самого языка. При интерпретации некоторого текста (в частности, программного) значение гки не зависит от состояния процесса интерпретации (так, при вычислении значения выражения «$x+1$» значение символа «+» не зависит от того, какое значение получила ранее переменная x). Глобальность конвенций состоит в том, что они устанавливаются

априори — до начала интерпретации текста и независимо от наличия какого бы то ни было текста. В этом смысле говорим, что гк-имена — это имена с *тривиальным* разыменованием (интерпретацией). Класс гки обозначим $\mathbf{GC}$.

Денотатами гк-имен могут быть платоновские объекты различных типов: так, цифры «0»–«9» и знаки констант π и e именуют объекты-числа, имена «+», «$\exp$», «$\sqrt{\ }$» и др. именуют функции. Значениями гки могут быть функционалы сколь угодно высоких порядков: имя «$\mathbf{R}$» в теории рекурсивных функций именует функционал примитивной рекурсии, имя «λ» именует оператор функциональной абстракции; слова «если» и «который» в естественном языке тоже можно считать именами операторов. В программировании глобально-конвенциональные имена — это те, значение которых жестко закреплено в семантике ЯП, такие как ключевые слова (**while**, **begin**, **goto**), имена стандартных процедур.

Лк-имена (класс $\mathbf{LC}$) получают свое значение исключительно в процессе интерпретации некоторого текста, причем значение сохраняется за лк-именем только на протяжении данного процесса интерпретации. Приписывание лк-имени определенного значения есть результат конвенции, *описанной в интерпретируемом тексте*. Таковы, например, конвенции «пусть $x = 3$» в тексте на естественном языке и «x:=3» в программном тексте. К лк-именам относятся имена переменных (как в традиционном математическом, так и в программистском смысле), имена определяемых программистом процедур и т.п.

Для разыменования (интерпретации) лк-имени требуется выяснение того, какая конвенция была ранее принята для данного имени. Будем говорить, что разыменование (интерпретация) лк-имен *квазитривиально*. Совокупность локальных конвенций, принятых к определенному шагу процесса интерпретации данного текста (например, состояние памяти ЭВМ перед исполнением некоторого оператора программы), назовем *динамическим контекстом* этого шага. Тогда лк-имена — это в точности те имена, значения которых зависят от динамического контекста. Иначе: разыменование (интерпретация) лк-имен *динамически-зависимо*. Важное свойство гк-имен состоит в том, что их разыменование (интерпретация), напротив, *динамически независимо*.

Несмотря на все различия, гк- и лк-имена обнаруживают две общие особенности. Во-первых, имена этих двух типов *структурно просты* в семантическом отношении. (Конечно, в синтаксическом отношении, как слова в некотором алфавите, они могут быть разложены на составные части — буквы или подслова, но эти части самостоятельного значения не имеют и не существенны при разыменовании (интерпретации) целого имени. Так, для разыменования имя переменной «count» нет смысла разбивать его на «со» и «unt».) Во-вторых, значение как гки, так и лки присваивается в результате *произвольной* конвенции (только в первом случае на уровне языка в целом, а во втором — на уровне конкретного текста). Провозглашая конвенцию, мы всегда заранее знаем, *какое* именно значение приписываем имени (*«сначала значение, затем имя»*). Учитывая отмеченную общность, объединим лки и гки в общий класс *конвенциональных* имен (к-имен, ки), $\mathbf{C} = \mathbf{GC} \cup \mathbf{LC}$. Для сколько-нибудь нетривиального программирования необходим выход за пределы класса $\mathbf{C}$.

К **нк-именам** (класс $\mathbf{C}$) относятся любые имена, имеющие сложную структуру, существенную для их разыменования (интерпретации): математические выражения (например, $1+2$, $sin(\alpha+2\beta)$), тексты программ и т.п. Денотат нки неизвестен в момент называния имени, его необходимо найти, используя для этого структуру имени и значения входящих в его состав лки и гки (*«сначала имя, затем значение»*). Произвол в отношении значений нки исключается (так, мы не можем по своему произволу провозгласить конвенцию «пусть 2+3 имеет значение 4»). В этом смысле неконвенциональные имена характеризуются *нетривиальным* разыменованием (интерпретацией). Всякое нки есть постановка некоторой задачи разыменования.

Рассмотрим подробнее класс к-имен. Языковые конвенции можно разделить на два типа: *явные* и *неявные*. Установление явной конвенции для имени A состоит в указании некоторого имени B, которое становится денотатом (синонимом) имени A. Явной конвенцией, напр., является присваивание «x:=3», в результате которого денотатом лк-имени «x» становится гк-имя «3». Определения процедур и макроопределения также дают примеры явных конвенций. Явные конвенции — это тот единственный случай, когда в рамках номиналистской парадигмы имеет смысл говорить о денотате имени как о реально существующем объекте. Ясно, что на практике явные конвенции применяются только к локально-конвенциональным именам: глобальные явные конвенции, хотя теоретически возможны, но совершенно бесполезны.

Определить к-имя посредством *неявной* конвенции — значит *дать совокупность правил интерпретации для различных случаев его употребления в составе нки*. Точнее, если ξ — определяемое имя, то его неявное определение состоит в задании списка правил вида $L_1(\xi) \to R_1 \quad (i = 1,\ldots,n)$, где $L_i(\xi)$ — некоторое выражение (нк-имя), содержащее вхождение имени ξ, R_i — произвольное выражение. Правило предписывает интерпретировать все выражения вида $L_i(\xi)$, *заменяя* их выражениями R_i. Схему (неполное выражение) $L_i(-)$ есть основания назвать *статическим контекстом* для данного вхождения имени ξ. Тогда получим, что неявно определенные имена — это в точности те имена, интерпретация которых зависит от статического контекста, или, иначе говоря, *статически зависима*. Напротив, имена, определенные явными конвенциями, характеризуются *статически независимой* интерпретацией.

С точки зрения номиналистского подхода неявно определенное имя вообще не имеет денотата: никакой лингвистический объект ему не сопоставляется. Неявно определенное имя имеет лишь значение — оно состоит в указании на соответствующую совокупность правил. Оставаясь на позициях платонизма, мы признали бы наличие денотата у такого имени, однако в качестве денотата выступал бы функционал высокого порядка или другой умозрительный объект. Неявные конвенции позволяют определять значение ключевых слов и управляющих конструкций языков программирования. Как правило, такие определения носят характер глобальных конвенций, хотя некоторые языки сверхвысокого уровня (например, язык Форт) поддерживают локальные неявные конвенции.

Рассмотрим пример неявной конвенции — определение значения ключевого слова (гк-имени) «**if**». Пускай условная конструкция языка программирования в общем случае имеет вид $\mathbf{if}(B, P_1, P_2)$, где B — выражение логического типа, играющее роль условия, а P_1 и P_2 — операторы, играющие роль ветвей условной конструкции. Определение в форме неявной конвенции состоит из трех правил, каждое правило описывает один из возможных случаев совместного употребления имени **if** с другими именами.

1) $\mathbf{if}(\mathbf{true}, P_1, P_2) \to P_1$;
2) $\mathbf{if}(\mathbf{false}, P_1, P_2) \to P_2$;
3) $\mathbf{if}(B, P_1, P_2) \to [{*}B \to \beta] \to \mathbf{if}(\beta, P_1, P_2)$,

где B в третьем правиле — выражение, отличное от символов констант **true** и **false** (нк-имя).

Правила 1 и 2 описывают простейшие случаи, когда слово «**if**» употребляется вместе с гк-именами **true** и **false**, т.е. когда условие ветвления выражено логической константой «истина» или «ложь». Правило 1 предписывает заменить выражение $\mathbf{if}(\mathbf{true}, P_1, P_2)$ на выражение P_1, а правило 2 означает замену $\mathbf{if}(\mathbf{false}, P_1, P_2)$ на P_2. Тем самым, в результирующем выражении элиминировано определяемое слово «**if**». Правило 3 относится к случаю, когда условие ветвления представлено некоторым выражением, истинностное значение которого еще предстоит вычислить. Согласно правилу, следует сначала найти тот объект β, который является результатом интерпретации выражения B (что показывает вспомогательная формула в квадратных скобках). Это будет одно из гк-имен **true** или **false**. Затем полученное гки подставляется в условную конструкцию вместо B. В результате получается конструкция, в которой условие ветвления выражено гк-именем, и ее дальнейшее преобразование проводится по правилам 1 или 2.

Продолжим структуризацию универсума имен. Покажем, что в универсуме имен можно построить две ортогональные иерархии, т.е. последовательности классов имен, такие, что имена каждого последующего класса в некотором определенном смысле сложнее имен предыдущего класса.

Абстрактные (идеальные) объекты, соответствующие тем или иным математическим и программным именам (временно допустим платонистскую онтологию умозрительных объектов), делятся на индивиды, функции над индивидами, функции над функциями (функционалы) и т.д. Иными словами, идеальные объекты образуют иерархию *уровней функциональности*. Поэтому аналогичная иерархия существует и в множестве имен, обозначающих эти объекты. Определим эту иерархию более строго.

Выделим в универсуме абстрактных объектов класс индивидов — объектов, которые не могут применяться к другому объекту в качестве операции. Множество индивидов составляет 0-й уровень функциональности, класс FP_0. Последующие классы строятся индуктивно.

Множество абстрактных объектов, имеющих *в точности* n-й уровень функциональности, будем обозначать FP_n, а множество объектов уровня *не выше* n-го — $\hat{F}P_n$. Очевидно, $FP_0 = \hat{F}P_0$, $\hat{F}P_n = \bigcup_{i=0}^{n} \hat{F}P_i$.

Пусть классы вплоть до FP_n уже построены. Объект принадлежит классу FP_{n+1}, если он есть частичная функция типа $\hat{F}P_n^k \to \hat{F}P_n$ (при $k = 1, 2, \ldots$) и при этом не принадлежит классу $\hat{F}P_n$. Иными словами, объект принадлежит классу FP_{n+1}, если он есть частичная функция типа $A_1 \times A_2 \times \ldots \times A_k \to A_{k+1}$, где все $A_i \subseteq \hat{F}P_n$ и хотя бы одно $A_j = FP_n$ $(i, j = 1, 2, \ldots, k+1)$. Наконец, полагаем, что класс F_n имен n-го уровня функциональности состоит в точности из тех имен, которые именуют объекты из FP_n.

Дадим теперь последовательно номиналистскую трактовку уровней функциональности имен, не опирающуюся на предположения о реальном существовании абстракций. При построении языка мы, его создатели, *произвольным образом* распределяем все глобально-конвенциональные имена по уровням функциональности F_n. Каждому имени уровня 1 и выше приписывается арность и типы аргументов. Затем устанавливаются правила интерпретации имен, причем таким образом, чтобы выполнялся следующее *Правило понижения уровня*. Пусть Ξ — имя уровня $n \geq 1$, арности k, и пусть α_1, … α_k — имена подходящих типов. Допускаются лишь такие правила интерпретации $\Xi(\alpha_1, \ldots, \alpha_k) \to \alpha_{k+1}$, в которых все α_i $(i = 1, \ldots, k+1)$ имеют тип не выше F_{n-1}, а хотя бы одно α_i — в точности тип F_{n-1}.

Следовательно, тот или иной уровень функциональности присущ имени не в силу объективных свойств данного имени и не в силу свойств идеального объекта — денотата данного имени, а в силу нашего соглашения использовать имя определенным образом.

Сущностной особенностью программирования является то, что имена могут именовать не только абстрактные объекты, но и другие имена. Вследствие этого имеет место иерархия *уровней косвенности*. Все неявно определенные гк-имена отнесем к нулевому уровню косвенности — классу N_0. Это те имена, к которым неприменима операция разыменования, т.к. они не именуют ничего кроме самих себя. Все явно определенные имена, имеющие своими значениями имена из N_n, по определению относим к классу N_{n+1}. Важно отметить, что иерархии уровней функциональности и косвенности ортогональны: уровень функциональности некоторого имени не связан с его уровнем косвенности.

Заключение

Построенная классификация имен по типу конвенциональности (деление на гк-, лк- и нк-имена и различение явных и неявных конвенций) общезначима, т.к. затрагивает лишь наиболее общие черты именования, существенные для программирования в целом. Вместе с тем, она допускает множество различных углублений и конкретизаций, ориентированных на то или иное частное программирование.

Литература

1. Frege G. Über Sinn und Bedeutung // Zeitschrift für Philosophie und philosophische Kritik. — 1882. — Bd. 100. — S. 25–50.
2. Карнап Р. Значение и необходимость. — М., 1959.
3. Витгенштейн Л. Философские работы. — М.: Гнозис, 1994, Ч. 1. — 520 с., Ч. 2. — 208 с.
4. Редько В.Н. Экспликативное программирование: ретроспективы и перспективы // Труды 1-й Межд. науч.-практ. конф. по программированию «УкрПрог-98». — К.: Кибцентр НАНУ, 1998. — С. 3–24.
5. Пуанкаре А. О науке. — М., 1983. — 736 с.
6. Гильберт Д., Бернайс П. Основания математики. Логические исчисления и формализация арифметики. Москва, «Наука», 1979. — 560 с.

Об авторе

В. Ю. Винник - Житомирский государственный технологический университет; (0412) 418-542; (0412); e-mail: vvinnik@ziet.zhitomir.ua ; vvin@ratibor.zt.ukrtel.net

ЭКСПЕРИМЕНТАЛЬНАЯ ОЦЕНКА ЭРГОНОМИЧЕСКИХ ПОКАЗАТЕЛЕЙ КОМПЬЮТЕРНЫХ ОБУЧАЮЩИХ ПРОГРАММ ПО ЭЛЕКТРОТЕХНИЧЕСКИМ ДИСЦИПЛИНАМ

Д.А. Яковец, Л.Х. Зайнутдинова

Аннотация: *Предложена методика экспериментальной оценки эргономических показателей компьютерных обучающих программ. Представлены результаты исследования эргономических показателей компьютерных учебных программ, созданных по методу теоретических образов.*

Ключевые слова: *компьютерные обучающие программы, эргономические показатели, метод теоретических образов*

Введение

Компьютеризация и надстраиваемый над ней процесс информатизации широко внедряются во все сферы жизнедеятельности человека, и, в том числе, в такую важную сферу социальной деятельности, как образование. При этом новые информационные технологии и средства вычислительной техники являются ядром информатизации образования. В настоящее время компьютерный класс в учебном заведении такой же необходимый и привычный атрибут, как библиотека. Разрабатываются новые средства обучения - всевозможные **компьютерные обучающие программы (КОП)**, в том числе и **электронные учебники (ЭУ)**.

На эффективность обучения влияют различные факторы, среди которых огромное значение имеет комфортность условий работы с компьютерной программой. Анализ этого аспекта взаимодействия человека и компьютера проводится в рамках **эргономики** - науки, комплексно изучающей закономерности взаимодействия человека и техники в процессе той или иной деятельности с целью выработки требований для повышения эффективности этой деятельности. В настоящем исследовании была поставлена задача оценить эргономические показатели ряда компьютерных обучающих программ по курсу электротехники.

Основная часть

Электронный учебник, как частный случай системы человек-машина **(СЧМ)**, характеризуется **системотехническими и эргономическими свойствами**. К системотехническим свойствам, согласно [Зараковский и др., 1993, С.27], относятся свойства, обуславливающие приспособление СЧМ к выполнению назначенных ей функций (эффективность, надежность, стоимость и др.). **Эргономические свойства** - характеристики СЧМ или ее элементов, которые определяются биомеханическими, физиологическими и психологическими возможностями деятельности человека.

Уже на первых этапах развития вычислительной техники проблема взаимодействия человека и компьютера стала предметом исследования в инженерной психологии, эргономики, психологии труда. Первоначально рассматривались различные аспекты операторского труда с использованием ЭВМ в АСУ. Основное внимание уделялось оптимизации аппаратной составляющей **человеко-машинного интерфейса (ЧМИ).** Сейчас, в связи с постоянным совершенствованием компьютерной техники, внимание эргономистов перенесено на программные компоненты ЧМИ.

В последние годы появились исследования, посвященные эргономическому проектированию взаимодействия человека и компьютера в системе образования: дошкольном, среднем, высшем. Причем рассматриваются в основном вопросы обеспечения оптимальных условий работы: правильный режим, микроклиматические условия, освещенность, правильная поза при работе, организация рабочего места. Вопросам эргономической оптимизации свойств КОП уделяется меньше внимания. Рекомендации по организации процедуры диалога, адаптации программы к индивидуальным особенностям учащихся, по

цветовому и пространственному оформлению информации на экране носят часто обобщенный характер. Работ, исследующих эргономические свойства КОП узкого назначения, имеющих свою специфику, например, для преподавания **общетехнических дисциплин (ОТД),** нет. Таким образом, вопрос оценки качества программных средств образовательного назначения стоит остро.

Одной из центральных проблем эргономики является изучение функциональных состояний пользователя, возникающих в ходе той или иной деятельности, и разработка адекватных методов их оценки и коррекции. **Функциональное состояние (ФС)** человека - комплекс характеристик тех функций и качеств человека, которые прямо или косвенно определяют выполнение рабочих операций [Введение в эргономику, 1974, с.94]. Традиционно исследуемые **виды ФС**: утомление, напряженность, монотония, стресс. Так как явления физического и психического утомления, а также способ их описания могут сильно зависеть от личности и установок испытуемого, следует попытаться получить и его личностный профиль, особенно в отношении стабильности и интраверсии/экстраверсии. Также глубокого изучения требуют и факторы, обуславливающие повышенную мотивацию пользователя.

Учитывая вышеизложенное, а также специфику преподавания ОТД с использованием информационных технологий и используя данные анализа научной литературы, посвященной эргономическим критериям эффективности деятельности и вопросам психологического тестирования, Яковец Д.А. разработала методику экспериментального исследования эргономических показателей компьютерных обучающих программ (рис.1).

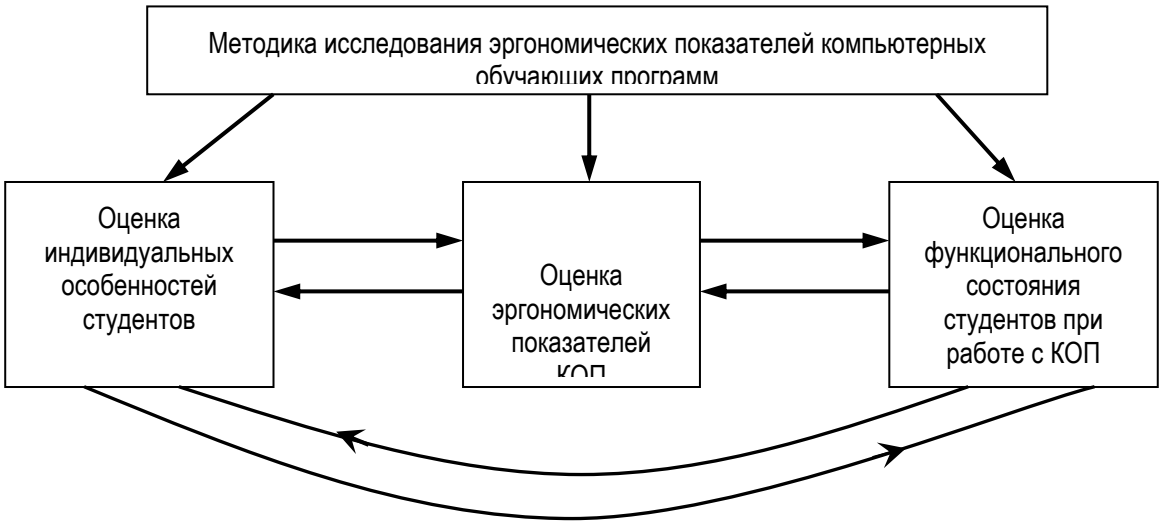


Рис.1

В настоящее время сложилась ситуация, когда наибольшее количество разрабатываемых КОП относится к области гуманитарных и естественнонаучных дисциплин для системы школьного образования. Доля КОП по общетехническим дисциплинам (электротехника, теплотехника, гидравлика, теоретическая механика и т.п.) и специальным техническим дисциплинам (по профилям выпускающих кафедр) незначительна. Одной из причин, тормозящих разработку КОП по техническим дисциплинам для системы среднего и высшего технического образования, является отсутствие теоретических основ эргономического проектирования подобных программ.

Предлагаемая в настоящем исследовании методика экспериментальной оценки эргономических показателей КОП дает возможность сравнительного анализа существующих КОП. Разработанная методика была применена для оценки нескольких компьютерных обучающих программ по курсу электротехники. Исследовались программа **«Трехфазные цепи» (TZ)**, созданная на кафедре Электротехники Астраханского государственного технического университета (АГТУ), а также программа **«Теория электрических цепей» (ТЭЦ),** разработанная в Сибирской государственной академии телекоммуникации и информатики [Бакалов и др., 1998].

Компьютерная программа ТЭЦ, на наш взгляд, может быть отнесена к наиболее распространенному для области технических дисциплин виду моделирующих программ. Программные средства для математического и имитационного моделирования позволяют расширить границы экспериментальных и теоретических исследований, дополнить физический эксперимент. При работе с программой ТЭЦ студенту предоставляется возможность выбора тех или иных параметров электрической цепи, программа ТЭЦ демонстрирует возникающее при этом изменение режима работы цепи. Недостатком подобных программ является отсутствие обратной связи и активной учебной деятельности студентов. Студент выступает в роли наблюдателя. По окончании работы с программой ТЭЦ оценка за урок не выставляется.

Программа TZ является электронным учебником. Согласно [Зайнутдинова, 1999 [1], с. 35] : "**Электронный учебник (ЭУ)** - это обучающая программная система комплексного назначения, обеспечивающая непрерывность и полноту дидактического цикла процесса обучения: предоставляющая теоретический материал, обеспечивающая тренировочную учебную деятельность и контроль уровня знаний, а также информационно-поисковую деятельность, математическое и имитационное моделирование с компьютерной визуализацией и сервисные функции при условии осуществления интерактивной обратной связи".

Программа TZ спроектирована на основе метода теоретических образов [Зайнутдинова, 1999[2]]. Известно, что учебный материал ОТД характеризуется высоким уровнем абстракции и потому тяжело воспринимается и усваивается студентами. **Предложенный метод теоретических образов обеспечивает повышение доступности изложения абстрактного учебного материала.**

Научное знание в подавляющем большинстве случаев передается в словесной форме в виде теорий, законов, понятий и сопровождается некоторыми символьными пояснениями, например, математическими формулами, это дает возможность говорить ***о вербализованной форме научных знаний.*** Из психологии известно, что для хорошего понимания и усвоения изучаемого материала необходимо, чтобы семантика текста была сведена к наглядно-образному представлению. На основании вышеизложенного, автором настоящей работы было предложено следующее определение: **"Теоретический образ – это наглядно-образное представление семантики вербализованных форм научных знаний (понятий, законов, теорий)"** [Зайнутдинова, 1997, с. 164].

Одной из интереснейших, но слабо разработанных проблем является **проблема передачи образа от одного человека к другому в процессе общения** [Ломов, 1991, с.69]. И именно она выдвигается на передний план в компьютерных обучающих системах. Теоретический образ, формировавшийся в сознании опытного преподавателя в течение многих лет, не может быть непосредственно передан учащемуся. При традиционной технологии обучения передача образа от одного человека к другому осуществляется на речемыслительном уровне. При использовании современных информационных технологий появляются принципиально новые возможности для передачи наглядно-образных представлений от педагога к учащемуся. Снижается потребность вербализации образа (словесного описания). Теоретический образ, являющийся достоянием опытного педагога, может быть с наименьшими потерями и искажениями донесен до учащегося через дидактическую программную систему. Но при этом возникает необходимость разработки новых технологий или стратегий такой передачи. На базе психологических и педагогических теорий обучения в настоящем исследовании были сформулированы рекомендации по проектированию теоретических образов для КОП [Зайнутдинова, 1999 [1], с.144-145].

Метод теоретических образов в той или иной степени способен повлиять на все компоненты учебно-познавательной деятельности учащихся, но наиболее существенные изменения имеют место в процессах восприятия, осмысления, запоминания и повторения учебного материала. Кроме того, метод должен улучшить эмоциональное отношение к учебе и уменьшить необходимость волевых усилий со стороны учащегося. При обучении с использованием метода теоретических образов учебная информация предъявляется не только в виде текстов и формул, но и в наглядно-образном виде. Идет восприятие и осмысление одновременно как вербализованной "левополушарной", так и наглядной "правополушарной" информации. Обработка "правополушарной" информации осуществляется с высокой скоростью, интегрально, целостно. Существует своего рода синергетический эффект взаимодействия лево- и правополушарных механизмов нашего мышления. При достигнутом уровне аппаратного и программного обеспечения наглядность и красочность образно представленной информации настолько приковывают внимание пользователя, что заметно снижается потребность в волевом регулировании процессов восприятия и осмысления. Метод теоретических образов обладает высокими потенциальными

возможностями предъявления учебного материала с опорой на взаимосвязь и взаимодействие понятийных, образных и действенных компонентов мышления, что особенно актуально при создании КОП для общетехнических дисциплин.

Высокая педагогическая эффективность КОП, разработанных с применением метода теоретических образов, была подтверждена исследованием [Зайнутдинова, 1999[2]]. Целью настоящей работы является сравнительная экспериментальная оценка эргономических показателей ряда КОП по курсу электротехники: моделирующей программы ТЭЦ и программы TZ, созданной с применением метода теоретических образов.

Экспериментальное исследование проводилось среди 85 студентов АГТУ второго курса дневной формы обучения. Участвовали студенты специальностей:

«Автоматизированные системы обработки информации и управления (АС) - 220200», 31 человек;

«Автоматизация технологических процессов и производств (АП) 210200», 22 человека;

« Сети связи и системы коммутации (АК) 200900», 32 человека.

Исследование проводилось в два этапа. На первом этапе оценивались индивидуальные особенности личности студентов, участвующих в эксперименте. На втором этапе оценивались эргономические показатели компьютерных обучающих программ и функциональное состояние учащихся при работе с данными программами.

Первый этап исследования

Оценка индивидуальных особенностей личности студентов проводилась один раз в специально выделенное время (на лекционном занятии).

При **оценке индивидуальных особенностей личности (ИОЛ) студентов** необходимо было выполнить несколько требований**:**

1. Методики оценки ИОЛ студентов должны быть надежны и валидны.
2. Методики оценки ИОЛ студентов должны быть удобны для группового тестирования. Необходимо обеспечить единообразные условия проведения тестирования. Процедура измерения, обработки и интерпретации исследуемых показателей должна быть регламентирована.
3. Необходимо учесть время, требуемое на проведение тестов (оценка ИОЛ должна быть проведена в течение одного лекционного занятия).

Первый этап исследования включил в себя:

1. Определение профессиональных склонностей студентов.
2. Определение степени логичности мышления.
3. Определение основных свойств нервной системы:
   - уравновешенность нервной системы (баланс нервных процессов, выражает соотношение между процессами возбуждения и торможения в клетках коры головного мозга);
   - подвижность нервной системы (способность быстро реагировать на изменения в окружающей среде);
   - сила нервной системы (отражает предел работоспособности клеток коры головного мозга – их способность выдерживать очень сильное, либо длительно действующее, хотя и не сильное возбуждение, не переходя в состояние торможения).

Определение профессиональных склонностей студентов. По мнению авторов, выявление профессиональных склонностей учащихся, работающих с КОП по электротехнике, крайне необходимо для оценки степени заинтересованности студентов в изучении ОТД. Наличие или отсутствие положительной мотивации при работе с компьютерной программой важный фактор, влияющий на эффективность обучения, восприятие свойств КОП, формирование состояния функционального комфорта или функционального дискомфорта. Согласно [Чайнова, 1985], функциональный комфорт - оптимальное функциональное состояние работающего человека, при котором достигнуто соответствие средств и условий труда его функциональным возможностям. В этом случае у человека формируется положительное отношение к деятельности, что обуславливает адекватную мобилизацию (активацию) его

психофизиологических процессов, отдаляет развитие утомления, способствует длительной и высокоэффективной работоспособности без ущерба для здоровья.

На основе анализа литературы по психологическому тестированию и в соответствии с требованиями, приведенными выше, для определения профессиональных склонностей студентов был выбран дифференциально-диагностический опросник Е.А. Климова [Горбатов, 1998].

Определение степени развития логичности мышления. Как отмечается в работе [Зайнутдинова, 1999 [2] , с. 95], при изучении общетехнических дисциплин (ОТД), к которым относится и электротехника, студенту необходимо сформировать в своей памяти достаточно большое количество (банк) теоретических понятий с учетом их взаимосвязей. При этом система научных понятий ОТД отличается высоким уровнем иерархичности и абстрактности и высокой степенью логической взаимосвязанности ее компонентов. Поэтому, по мнению авторов, степень развития логичности мышления - немаловажный фактор, влияющий на эффективность работы студентов с данными КОП. Степень развития логичности мышления исследовался с помощью теста возрастающей трудности (методика Ровена) [Столяренко, 2000, С.111]. Данный тест соответствует приведенным выше требованиям.

Определение основных свойств нервной системы студентов, участвующих в эксперименте, имеет существенное значение для данного исследования. Согласно [Словарь … , 1998, с.600], **свойства нервной системы** – устойчивые особенности нервной системы, влияющие при прочих равных условиях на индивидуальные психологические особенности человека. Совокупность свойств нервной системы, образующих тот или иной тип нервной системы, составляет физиологическую основу индивидуального своеобразия деятельности человека, оказывает непосредственное влияние на формирование того или иного функционального состояния (утомления, стресса, функционального комфорта, продуктивной или непродуктивной напряженности и др.) при различных видах деятельности. Лабораторные методики диагностики основных свойств нервной системы требуют специальных условий проведения и аппаратуры. Они достаточно трудоемки. Поэтому в данном исследовании для определения уравновешенности и подвижности нервной системы использован опросник Я.Стреляу [Столяренко, 2000, С.180], а для определения силы нервной системы - теппинг-тест [Столяренко, 2000, С.187].

Результаты первого этапа экспериментального исследования (оценка индивидуальных особенностей личности студентов) показали:

1. Среди студентов всех трех специальностей большинство (АС – 83%, АП – 86%, АК – 63 %) проявили склонность к профессиям, объектом труда которых является техника и / или знаковая информация, то есть к деятельности, предполагающей использование разного рода машин, материалов, иных продуктов цивилизации и / или основанной на предпочтениях к обработке цифр, букв, кодов, других символов.

Таким образом, было установлено, что для контингента студентов, участвующих в эксперименте, выбор профессии, в основном, адекватен склонностям личности. Следовательно, можно предположить положительную мотивацию студентов и их заинтересованность в освоении учебного материала по электротехнике.

2. Большинство студентов, участвующих в эксперименте, обладают высокой или средней степенью развития логичности мышления: на потоке АП – 83% от общего числа учащихся на специальности, АК – 97%, АС – 95%. Соответственно, студенты, участвующие в эксперименте, имеют уровень развития логичности мышления, достаточный для восприятия и усвоения учебного материала ОТД.

3. Студенты всех трех специальностей, в основном, обладают слабой и средне-слабой нервной системой. Преимущество слабой нервной системы перед сильной в способности реагировать на стимулы более низкой интенсивности. Слабая нервная система более тонко организована, более чувствительна. Следовательно, влияние тех или иных свойств КОП на функциональное состояние студентов при работе с данными программами должно проявиться достаточно ярко *(в заметной степени).*

Второй этап исследования

Оценка эргономических показателей компьютерных обучающих программ и функционального состояния студентов при работе с КОП по курсу электротехники проводилась на практических занятиях в дисплейном классе в течение нескольких занятий.

При работе с компьютером значительная часть информации воспринимается человеком через зрительный анализатор. Повышение качества отображаемой визуальной информации может идти по двум направлениям: оптимизация светового режима (яркость, освещенность в помещении, яркость, контрастность элементов на экране) и оптимизация человеко-машинного диалога. Необходимо отметить, что непрерывное совершенствование компьютерной техники ведет к снижению актуальности оценки и контроля аппаратных параметров отображения. Поэтому внимание эргономистов сегодня сосредоточено, главным образом, на области программного обеспечения процессов отображения информации. Настоящее исследование проводится в рамках данного направления.

На основе анализа научной литературы, посвященной вопросам эргономического проектирования программных средств различного назначения, и многолетней практики использования электронных учебников в учебном процессе АГТУ для оценки качества обучающих программ в настоящей работе было предложено ввести следующие эргономические показатели:

1. Используемая цветовая гамма: чрезмерно яркая; нормальная; недостаточно яркая.
2. Удобство считывания информации: шрифт излишне мелкий; нормальный; излишне крупный.
3. Пространственное расположение элементов информации на экране: неудобное; скорее удобное; удобное.
4. Предъявление информации в динамическом виде (подвижные объекты на экране): способствует лучшему пониманию учебной информации; не улучшает понимание учебной информации; раздражает.
5. Степень ясности (понятности) последовательности действий при работе с программой: последовательность действий понятна; иногда возникают трудности в понимании последовательности действий; последовательность действий не понятна.

Использован метод анкетирования. Студенты оценивали конкретную компьютерную обучающую программу по выделенным показателям в конце учебного занятия по этой программе**.**

Результаты экспериментального исследования эргономических свойств TZ и ТЭЦ представлены на рис.2. Диаграммы иллюстрируют явное преимущество TZ по сравнению с ТЭЦ по всем показателям: пространственное расположение элементов информации, степень ясности последовательности действий, цветовая гамма, удобство считывания информации (шрифт).

При оценке **функционального состояния (ФС) студентов при работе с КОП** необходимо было выполнить несколько требований**:**

1. Методики оценки ФС должны быть надежны и валидны.
2. Методики оценки ФС должны быть удобны для группового тестирования.
3. Оценка ФС не должна нарушать график учебного процесса. Время, затрачиваемое на оценку ФС не должно превышать 10 минут (5 минут в начале занятия, 5 минут в конце).

Для оценки функционального состояния (ФС) в эргономике используется два типа методов: физиологические и психологические [Эргономика, 1988]. Из **физиологических методов оценки ФС** в данном исследовании использованы измерение давления и пульса. Эти физиологические параметры измерялись избирательно у некоторых студентов в начале и конце занятия. Выбор студентов для контроля этих параметров осуществлялся на основе результатов тестирования по свойствам нервной системы. Были выбраны учащиеся с наиболее неуравновешенной и слабой нервной системой и с наиболее сильной, уравновешенной нервной системой.

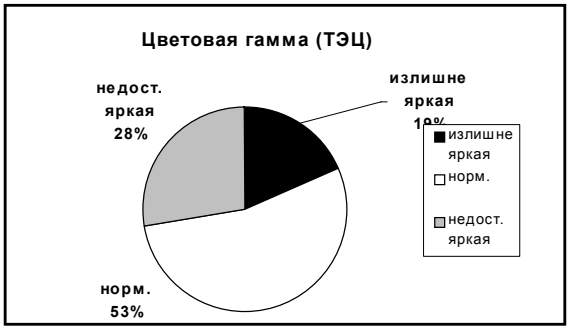
Для диагностики изменений ФС использовались **психологические тесты** на объем кратковременной зрительной памяти (тест на запоминание чисел) и объем распределения и переключения внимания (тест «Числовой квадрат») [Рабочая книга ...., 1996, с.172-174].

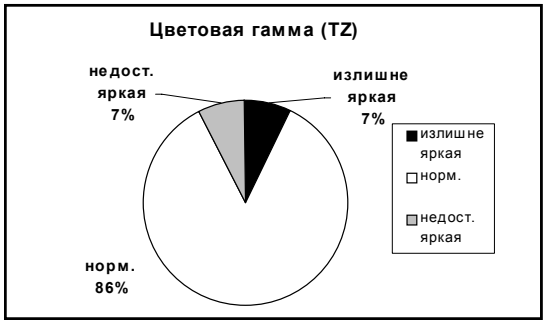
Результаты оценки ФС с использованием психометрических методик показали:

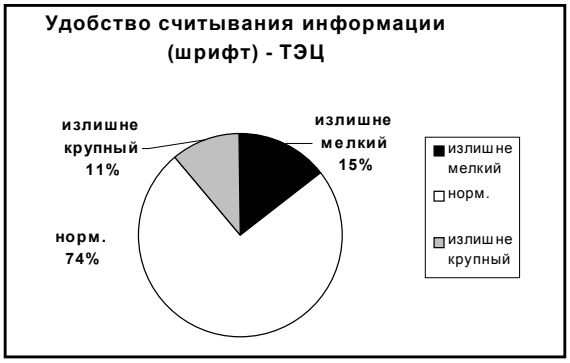
Снижение объема кратковременной памяти при работе с TZ почти у половины студентов, тогда как при работе с ТЭЦ снижение объема кратковременной памяти наблюдается у меньшего количества студентов (АС – 19%, АП+АК – 28%).

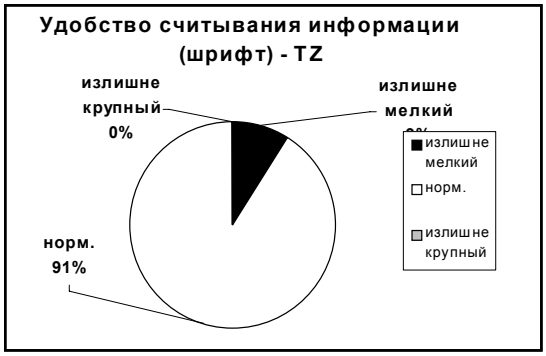
Снижение объема распределения и переключения внимания при работе с TZ больше, чем при работе с ТЭЦ (TZ -43%, ОС- 30% соответственно).

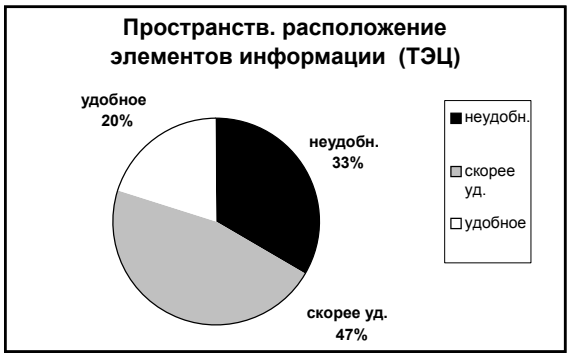
Оценка эргономических показателей программ ТЭЦ и TZ

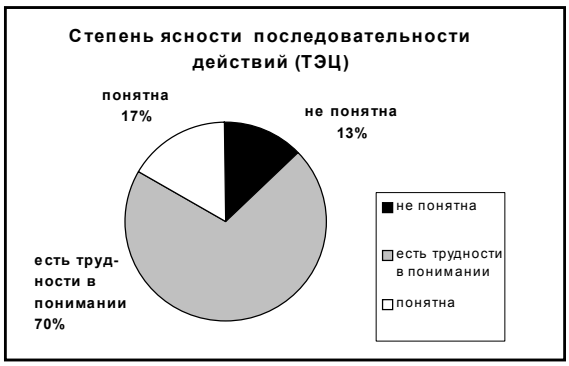












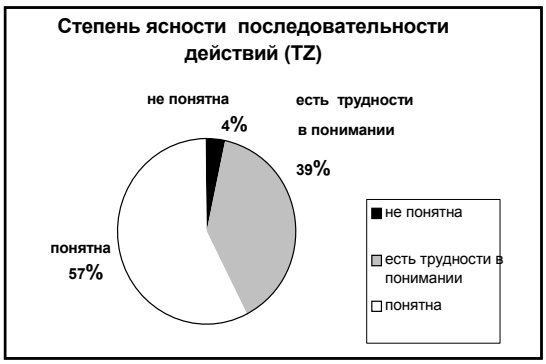


Рис.2

Полученные результаты можно объяснить большей интенсивностью учебной деятельности студентов при работе с программой TZ (студенты производят расчеты электрических цепей, построение векторных диаграмм и получают оценку по каждому заданию). Работа с моделирующей программой ТЭЦ сводится к наблюдению графиков и векторных диаграмм, оценка за урок не выставляется.

Для **субъективной оценки своего функционального состояния** учащимся в конце учебного занятия предлагался тест, подготовленный на основе теста дифференцированной самооценки (САН) [Столяренко, с.367]. Были выбраны показатели по шкалам «Самочувствие», «Активность», «Настроение», имеющие наиболее ясную и четкую формулировку. Испытуемых просили соотнести свои ощущения с рядом признаков, формулировка каждого из которых максимально сжата и представлена коротким утверждением. Оценивались следующие показатели:

1. Самочувствие (хорошее, плохое; затрудняюсь ответить).
2. Напряжение (напряжен, расслаблен, затрудняюсь ответить).
3. Бодрость (бодрый, вялый, затрудняюсь ответить).
4. Настроение (хорошее, плохое, затрудняюсь ответить).
5. Удовлетворение от работы (доволен, не доволен, затрудняюсь ответить).
6. Степень концентрации внимания (внимательный, рассеянный, затрудняюсь ответить).
7. Степень возбуждения (возбужденный, сонливый, затрудняюсь ответить).

Анализ результатов субъективной оценки учащимися своего состояния после работы с программами TZ и ТЭЦ показал:

- Значительное преимущество TZ по сравнению с ТЭЦ по показателям: «бодрость» (на 19%), «удовлетворение от работы» (на 24%).
- Преимущество TZ по сравнению с ТЭЦ по показателям: «самочувствие» (на 13%), «настроение» (на 11%).
- Обе программы вызвали одинаковую степень напряжения
- TZ потребовала более высокой степени концентрации внимания и вызвала большее возбуждение у студентов. По-видимому, это объясняется тем, что по итогам работы с TZ студент получает оценку, а после работы с ТЭЦ оценка не выставляется.

Заключение

Результаты проведенного исследования показали пригодность разработанной методики для экспериментальной оценки эргономических показателей КОП. Впервые осуществлен контроль и проведен сопоставительный анализ эргономических показателей ряда компьютерных обучающих программ по общетехнической дисциплине «Электротехника». Полученные результаты показали преимущество компьютерной обучающей программы TZ , разработанной по методу теоретических образов, по сравнению с моделирующей программой ТЭЦ.

Учет эргономических аспектов восприятия информации важен при проектировании программ учебного назначения. Создавая комфортный интерфейс, мы будем способствовать эффективному усвоению знаний учащимися, формированию у них положительного отношения к учебной деятельности, что обуславливает адекватную мобилизацию психофизиологических процессов, отдаляет развитие утомления, способствует длительной и высокоэффективной работоспособности без ущерба для здоровья.

Литература

[Бакалов и др., 1998] Бакалов В.П., Крук Б.И., Журавлева О.Б. Компьютерный учебник по теории электрических цепей // Материалы четвертой междунар. науч.-метод. конф. «Новые информационные технологии в преподавании электротехнических дисциплин». - Астрахань: АГТУ, 1998. – С.58-62.

[Введение в эргономику, 1974] Введение в эргономику. / Под ред. Зинченко В.П. –М.: Сов. Радио, 1974. -352с.

[Горбатов, 2000] Горбатов Д.С. Практикум по психологическому тестированию: Учебное пособие – Самара: Издательский дом «Бахрам-Х», 2000. –С.84.

[Зайнутдинова, 1999[1]] Зайнутдинова Л.Х. Создание и применение электронных учебников (на примере общетехнических дисциплин) // - Астрахань: Изд-во «ЦНТЭП», - 364с.

[Зайнутдинова, 1999[2]] Зайнутдинова Л.Х. Теоретические основы создания и применения дидактических интерактивных программных средств по общетехническим дисциплинам. Дис. …. докт. пед. наук. Астрахань-Москва, 1999. – 410с.

[Зайнутдинова, 1997] Зайнутдинова Л.Х. Создание теоретических образов как метод повышения эффективности электронных учебников // Материалы науч.-технич. конф. «Новые информационные технологии в региональной инфраструктуре (НИТ РИ-97)». - Астрахань: АГТУ, 1997. – С.163-167.

[Зараковский и др., 1993] Зараковский Г.М., Мунипов В.М., Шлаен П.Я. Эргономика в вопросах и ответах. Материалы понятийной базы эргономики. – Тверь, 1993. –68с.

[Ломов, 1991] Ломов Б.Ф. Проблема образа в психологии. В кн.: Вопросы общей, педагогической и инженерной психологии. (Труды д.чл. и чл.-кор. АПН СССР). - М.: Педагогика, 1991. - С. 65-72.

[Рабочая книга … , 1996] Рабочая книга практического психолога: Технология эффективной профессиональной деятельности (пособие для специалистов, работающих с персоналом). – М.: Издательский дом «Красная площадь», 1996. – 400с.

[Словарь …, 1998] Словарь практического психолога. / Сост. С.Ю. Головин. – Минск: Харвест, 1998. –800с.

[Столяренко, 2000] Столяренко Л.Д. Основы психологии. Практикум. –Ростов н/Д.: Феникс, 2000. –566 с.

[Чайнова, 1985] Чайнова Л.Д. Функциональный комфорт как обобщенный критерий оптимизации трудовой деятельности. // Техническая эстетика. –1985, -№2. –С.19-20.

[Эргономика, 1988] Эргономика: Учебник для вузов по спец. «Психология». / Под ред. А.А. Крылова, Г.В. Суходольского – Л.: ЛГУ, 1988. –182с.

Сведения об авторах:

Яковец Диляра Ахтямовна – Астраханский государственный технический университет, ассистент кафедры «Автоматизированные системы обработки информации и управления»; Россия, 414025, Астрахань, ул. Татищева, 16; E-mail: dl_sun@mail.ru

Зайнутдинова Лариса Хасановна – Астраханский государственный технический университет, зав. кафедрой электротехники, доктор педагогических наук, профессор; Россия, 414025, Астрахань, ул. Татищева, 16; E-mail: zain@astranet.ru

Section 8: Planning and Sheduling

СТРУКТУРНО-ЦЕЛЕВОЙ АНАЛИЗ СИСТЕМ НА ОСНОВЕ ЛОГИКО-ЛИНГВИСТИЧЕСКИХ ФОРМАЛИЗАЦИЙ

Л.М. Лукьянова

***Abstract**: Systems analysis (SA) is widely used in complex and vague problem solving. Initial stages of SA are analysis of problems and purposes to obtain problems/purposes in a less degree of complexity and vagueness that are combined into hierarchical structures of problems(SP)/purposes(PS). Leaders have to be sure the PS and the purposes realizing system (PRS) that can achieve the PS-purposes are adequate to the solved problem. However, usually SP/PS are not rather substantiated, because its development is based on a collective expert examination, in which logic of natural language is used, and expert estimation methods. That is why scientific foundations of SA are not supposed to have been completely formed. A structure-and-purpose approach to SA based on a logic-and-linguistic simulation of problems/purposes analysis is a step in the direction of not only formalization of the initial stages of SA to improve adequacy of its results, but also to increase quality of SA as a whole. Leaders of industrial organizing systems using the approach eliminate logical errors in SP/PS on early stages of planning and their complex solutions are well founded.*

***Keywords**: industrial organizing system, problem situation, systems analysis, quality of systems analysis,* purposes *structures correctness, structure-and-purpose approach, situations control, logic-and-linguistic simulation, analytic evaluation.*

Введение

Предметом нашего рассмотрения являются промышленные организационные системы. Возникающие в таких системах проблемные ситуации отличаются сложностью и существенной неопределенностью. Наряду с присущей сложным системам уникальностью, непредсказуемостью поведения в конкретных условиях, нестационарностью отдельных параметров, способностью адаптироваться к изменяющимся условиям среды и изменять свою структуру, организационные системы производственной сферы характеризуются иерархичностью структур, стандартизацией и унификацией выпускаемой продукции и технологического оборудования, динамичным изменением объема и ассортимента продукции, технологическим обновлением и техническим перевооружением производства. Системный анализ (СА) как методология постановки и решения проблемных задач становится необходимым условием эффективного функционирования и развития таких систем, устранения складывающихся в них проблемных ситуаций.

Подходы и методы решения проблемных задач восходят к системе ПАТТЕРН [Лопухин, 1971]. Среди современных подходов к анализу систем выделяются информационный, а также подход, основанный на комплексировании и поочередном использовании неформализованных и формализованных методов [Волкова, Денисов, 2001], а среди перспективных методов уменьшения сложности и неопределенности задач – такие специальные методы СА, как имитационное динамическое, ситуационное, структурно-лингвистическое моделирование, обзор которых имеется в [Волкова, Денисов, 2001] и метод анализа иерархий [Saaty, Kearns, 1991]. Однако эти и другие подходы и методы при их бесспорной ценности для теории и практики СА имеют существенный недостаток – начальные, наиболее важные для устранения проблемных ситуаций решения по выявлению и увязке в структуру проблем (СП) и целей (СЦ)

основываются на знаниях и опыте экспертов, т.е. на субъективных моделях, создаваемых средствами естественного языка, и коллективной интерпретации проблем/целей, СП/СЦ. Поэтому им присущ значительный субъективизм, а получаемым с их помощью результатам – логические просчеты, что обнаруживается лишь в процессе достижении целей, обусловливая невысокое качество СА в целом.
Научные основы системного анализа нельзя поэтому считать окончательно сформированными, в том числе в связи с существованием *научно-технической проблемы* для СА промышленных организационных систем – отсутствие целостного, в достаточной степени объективного подхода, реализующей такой подход методологии, соответствующих моделей и конструктивных методов.
Структурно-целевой подход к анализу организационных систем производственной сферы – шаг в этом направлении, развивающий метод [Лукьянова, 1986]. Для решения научно-технической проблемы СА необходимо, прежде всего, *исследовать закономерности целеполагания в производственной сфере*, которые до сих пор изучены не достаточно (первая подпроблема СА-проблемы); *семантику проблем/целей и отношений между ними в СП/СЦ*, которую определяют в естественном языке, что приводит к их многозначности (вторая подпроблема СА-проблемы); требования к СП/СЦ, в том числе непротиворечивость и полнота, которые также декларированы в естественном языке и поэтому многозначны и недостаточно ясны (третья подпроблема СА-проблемы). Исходя из этого, станет возможным повысить объективность и конструктивность проблемно-целевого анализа и СА-методологии в целом.

Базовые понятия постановки проблем и выдвижения целей

В системном анализе могут быть выделены два основных процесса с взаимообратным ходом времени: целеполагание (ЦП), в котором определяются желаемые результаты деятельности, и целереализация (ЦР) – получение реальных результатов в процессе достижении целей.
Определим базовые для структурного ЦП в организационных системах производственной сферы понятия: потребность, проблема, цель, СП/СЦ (семантическое поле понятий разрабатываемой методологии детально рассмотрено в [Lukiyanova, 2002]).
Потребность в чем-либо всегда объективна. Если потребность не может быть удовлетворена просто, в системе возникает проблема. *Проблема* – это противоречие между желаемым и существующим (например, между желаемой и существующей ситуациями, что может означать необходимость изменения текущей ситуации).
Цель всегда объективна. Анализ различных определений данного понятия позволил выявить его обобщенную семантику:

< Цель > ::= < Желаемый результат деятельности >
[<Структура >] [< Время >]. (1)

В определении (1) выделены три семантических множителя, последние два из которых, заключенные в квадратные скобки, факультативны. Действительно, не всякая цель требует структурного представления (например, цель, для достижения которой имеются средства), а привязка к временной шкале осуществляется при постановке задачи (время и другие перераспределяемые виды ресурсов характеризует задачу и ЦР).
Анализ более тысячи формулировок целей промышленных отраслей показал тесную связь между ними и формулировками- проблем, при этом часто цель как бы отрицает соответствующую проблему, а из (1) следует, что цель может быть простой или сложной. Также выяснено, что семантика отношений между проблемами в СП и целями в СЦ идентична.
При решении сложной и существенно неопределенной проблемы *структура главной проблемы/структура главной цели* – это такая СП/СЦ, в которой проблемы/цели находятся в определенных структурообразующих и неструктурных отношениях [Lukiyanova, 2002]. Структурные отношения составляют отношения подчинения, сопоставимости и полноты, вспомогательные неструктурные отношения используются при оценке корректности СП/СЦ, а дополнительные неструктурные отношениями позволяют учитывать ресурсы, которыми располагают руководители и управленческий персонал (УП) организационных систем.

Закономерности целеполагания в производственной сфере

Проблемы затрудняют функционирование и развитие систем и могут обусловливать потребности. Простая проблема обусловливает простую потребность, а последняя – простую цель:

$$\text{потребность} \xrightarrow{\text{мотивация}} \text{цель}. \quad (2)$$

Формула (2) выражает *первую закономерность целеполагания*. В соответствии с (2) для простой цели ЦП завершается, если УП располагают средствами для ее реализации:

$$\text{цель} \xrightarrow{\text{средства}} \text{результат}. \quad (3)$$

В этом случае новая проблема не возникает. ЦП продолжается, если УП не располагают средствами для реализации цели, что позволяет сформулировать *вторую закономерность целеполагания:*

$$\text{желаемый результат деятельности} \rightarrow \text{желаемые средства}. \quad (4)$$

Сложная проблема обусловливает сложную цель, последняя рассматривается как система, структура которой анализируется. (4) определяет *базовую стратегию ЦП.* Дополнительные стратегии приведены в [Лукьянова, 2001].

Исследование большого корпуса формулировок целей в промышленных отраслях и возможностей их декомпозиции позволили сформулировать правила рационального ограничения избыточности и многозначности целей в их формулировках, явного описания целевых частей формулировок, выражающих семантику функциональных ролей, семантику базовых для конкретной системы элементов и их определение в пространстве свойств. Были определены следующие возможности уменьшения комплексности целей: статус (внешний, внутренний), аспекты деятельности (социальный, экономический, управленческий, производственный), виды экономической и производственной (добыча, переработка и т.д.) деятельности, функции управления (анализ, планирование, организация и т.д.).

Поскольку ЦП в любой промышленной системе исходит из цели, определяемой ее надсистемой (системой, в подчинении которой находится рассматриваемая), логично считать, что такая цель выражает для подчиненной системы абсолютную ценность намечаемого в ней результата, в то время как сама эта система, являясь средством достижения определенной извне цели, и ее собственные цели, определяющие достижение внешней цели, выражают утилитарную ценность. В связи с этим при структурном ЦП определяемый каждой целью СЦ желаемый результат деятельности за исключением главного и листовых может рассматриваться с двух позиций: как абсолютно ценный для подчиняемых целей и как утилитарно ценный для подчиняющей цели (*третья закономерность целеполагания*).

Структурно-целевой анализ промышленных организационных систем

Обсуждается “структурно-целевая” парадигма СА, использующая два определяющих концепта целустремленных систем: “структура” и “”цель”, – и учитывающая доминирующую роль цели в СА. Действительно, промышленная организационная система есть средство достижения поставленных перед ней целей, и для эффективного управления необходимо анализировать структуру ее комплексных целей. При этом целесообразно учитывать семантическую связь проблем и целей (как отрицание проблем), критериев (как правил, включающих описание целей) [Lukiyanova, 2002], функций целереализующих систем (ЦРС): цели и СЦ определяют ЦРС и доминируют как в планировании решения проблемы, так и в процессе исполнения плана. Исходя из данной парадигмы, постулируется следующее.

Постулат 1. Обобщенная семантика целей и отношений между ними в организационных системах производственной сферы определяет концептуальную логическую модель анализа проблем/целей и структурную модель его результатов (иерархических СП/СЦ).

Постулат основывается на понятии иерархической структуры организационных систем производственной сферы, на функциональных ролях их частей и на понятиях абсолютной и утилитарной ценности.

Постулат 2. Формальный логико-семантический анализ проблем/целей организационных систем производственной сферы обусловливает непротиворечивость получаемых таким образом СП/СЦ, а настраиваемость логико-семантической модели на предметную область (ее открытость) – позволяет реализовать рассуждение о проблемах/целях СП/СЦ и осуществить проверку полноты СП/СЦ.

Дополнительные постулаты рассмотрены в [Lukiyanova, 2002].

Для ориентации в актуально бесконечной совокупности методов, методик, способов, процедур анализа проблем/целей и синтеза СП/СЦ, а также для выбора наиболее эффективного класса методов,

обеспечивающего выявление паралогизмов (иными словами ошибок человеческого рассуждения) была осуществлена классификация методов, включая широко распространенные. Использованы неформальная, частично-формальная и формальная степени представления следующих оснований классификации: *языков описания целей* (первый уровень классификации) как интерфейса между УП и формальной логико-семантической системой, *правил декомпозиции целей* (второй уровень), проверяющих корректность СЦ системы, *средства представления СЦ и их характеристик* (третий уровень) для организации выходного интерфейса. Результаты классификации представлены на рис. 1. Пустые классы методов выделены на рис. 1 зачерненными кружками.

Среди классов выделяются три однородных: класс K^{111} включает неформальные, класс K^{222} – частично-формальные, класс K^{333} – формальные методы. Методы остальных классов являются неоднородными.

Первый реализованный класс – K^{111}. Он включает большинство реализованных методов, таких как [Лопухин, 1971], [Черняк, 1975], [Перегудов, Тарасенко, 1989], [Saaty, Kearns, 1991], [Волкова, Денисов, 2001] и подобные им. Преимущества данных методов в их универсальности и охвате обеих фаз анализа (декомпозиции и оценивания) СЦ. Основной недостаток – многозначность целей, правил декомпозиции и свойств СЦ, обусловливающая высокий уровень субъективизма и трудности обнаружения логических ошибок в СЦ. *Второй реализованный класс – K^{113}*. Он включает метод [Поспелов, Ириков, 1986] и подобные им. Преимущества данных методов – формальное описание СЦ – не увеличивает, однако, уровень их конструктивности и, тем самым, возможности обнаружения ошибок в СЦ.

Третий реализованный класс – K^{221}. Он включает методы, эквивалентные методу [Силич, Тарасенко, 1982]. Они достаточно жестко стандартизируют формулировки целей и определяют сценарную форму описания СЦ. Это позволяет автоматически формировать СЦ на некоторых шагах декомпозиции. Основной недостаток таких методов – невозможность для анализа принципиально новых проблемных ситуаций и в других системах. *Четвертый реализованный класс – K^{231}*. Он включает методы, подобные методу [Романов, Клыков, 1974], которые являются высоко-конструктивными и, в отличие от методов третьего класса, гибкими. Однако среди автоматически формируемых ими всех возможных альтернатив СЦ могут быть семантически бессмысленные и прагматически бесполезные. Кроме того, они не располагают средствами оценки свойств СЦ, посредством которых может быть оценена логическая правильность СЦ.

Пятый реализованный класс – K^{311}. Он включает методы, которые эквивалентны методу [Кондратов, Ростанец, 1982]. Формальное описание целей в них слабо связано с возможностями декомпозиции. *Шестой реализованный класс – K^{331}.* Он включает методы, подобные методу [Nilson, 1973]. Как правило, такие методы работают в закрытых и не очень больших мирах. Другой их недостаток – недостаточная проработанность возможностей учета семантики целей, отношений между целями и свойств СЦ.

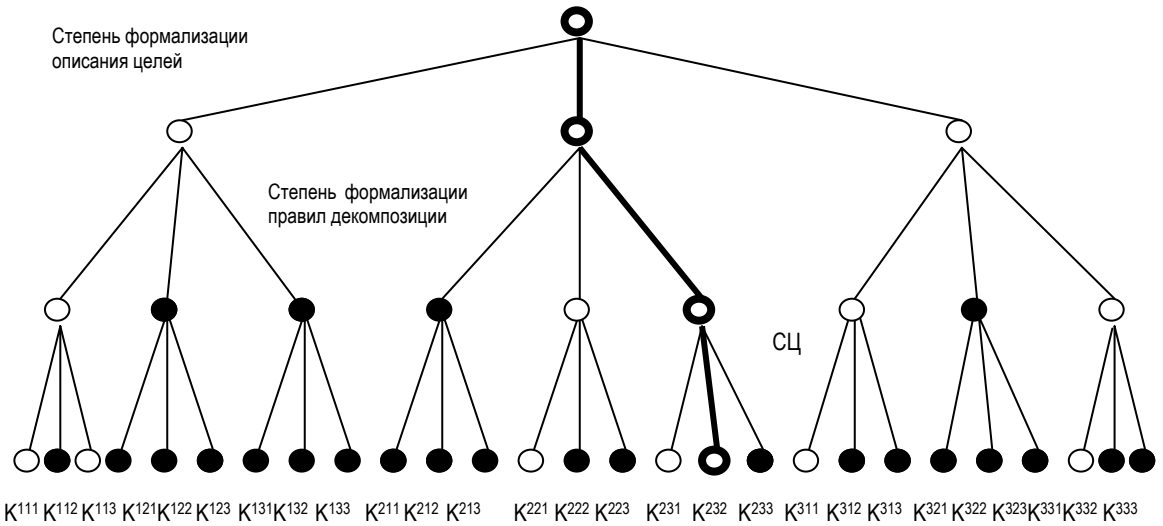


Рисунок 1. Классификация методов формирования иерархических СЦ

Классификация способствовала выбору наиболее адекватного для решения поставленной научно-технической проблемы класса методов – K^{232}. На классификационной схеме он выделен жирной линией. Данный класс обеспечивает необходимый для восприятия уровень частично-формального описания

проблем/целей, частично-формального представления СП/СЦ, формальные правила анализа проблем/целей и следующие преимущества: 1) описание проблем/целей в ограниченном естественном языке эффективно для использования экспертами/УП, формальной системой анализа проблем/целей и реализации интеллектуального интерфейса между ними; 2) основывающаяся на семантике предметной области логическая модель проблемно-целевого анализа эффективна для выявления ошибок анализа СП/СЦ; 3) графическое представление СП/СЦ с описанными в ограниченном естественном языке проблемами/целями эффективно для восприятия экспертами/УП и реализации выходного интерфейса.

Для класса K^{232} исследованы и установлены подробно рассмотренные в [Лукьянова, 2002] следующие принципы структурно-целевого анализа:

1. ЦП как анализ целей для решения проблем и ЦР как синтез результатов решения проблем – логически обусловленные процессы с взаимообратным ходом времени.
2. Целесообразен человеко-машинный анализ проблем/целей (критериев, функций ЦРС) как адекватное практическое рассуждение о проблемах/целях (*в отличие от недостаточно адекватных человеческих рассуждений или алгоритмов декомпозиции проблем/целей*).
3. Иерархические структуры системы: СП→СЦ, СЦ→СФ ЦРС, СФ ЦРС→ ЦРС, – логически и семантически связаны.
4. Складывающиеся в организационных системах ситуации обусловливают анализ проблем/целей (критериев, функций ЦРС).
   - 4.1. Формализация системного анализа должна обеспечивать учет семантики проблем/целей (критериев, функций ЦРС) и семантики отношений между ними (*логическая страта анализа*).
     - 4.1.1. Частичная лингвистическая формализация обеспечит явное выражение семантики проблем/целей (критериев, функций ЦРС).
     - 4.1.2. Логико-семантическая формализация анализа проблем/целей обеспечит (логическую) непротиворечивость, (модельную) полноту вывода и (семантическую) применимость правил вывода.
       - 4.1.2.1. Непротиворечивость иерархических СП/СЦ обусловлена принципом 3.1.2.
       - 4.1.2.2. Полнота иерархических СП/СЦ обусловлена принципом 3.1.2.
     - 4.1.3. Классификация ситуаций на целях упростит анализ (выбор текущей стратегии).
     - 4.1.4. Частичная графо-лингвистическая формализация СП/СЦ (структур критериев (СК), структур функций (СФ) ЦРС) обеспечит адекватное представление результатов структурного анализа.
   - 4.2. Формализация оценивания реализации целей обеспечит учет имеющихся для ЦР ресурсов (*математическая страта анализа*).
5. Учет в системном анализе возможностей человеческого восприятия (факультативный принцип).

Для реализации принципа 4 использована фундаментальная идея ситуационного управления [Поспелов, 1995] и предложена концептуальная модель анализа проблем/целей (критериев, функций ЦРС). Рассмотрим концептуальную модель анализа проблем/целей, приведенную на рис. 2.

В соответствии с принципами 4.1 и 4.2 структурно-целевой анализ в промышленных отраслях целесообразно стратифицировать, получая СП/СЦ логическими средствами (логическая страта), а затем оценивая проблемы/цели (математическая страта).

В соответствии с принципом 4.1.1 для реализации интеллектуального интерфейса предложен язык L_{in}^1 описания проблем/целей (/критериев/функций ЦРС). Как показал анализ, наиболее адекватным для L_{in}^1 является фреймовый язык [Лукьянова, 2001], базирующийся на двухуровневой лингвистической модели формулировки проблемы/цели, первый уровень которой (макро-описатель) есть настраиваемый по составу ролей ролевой фрейм, выражающий функциональную формулу деятельности в соответствующей системе, а второй (микро-описатель) – описатель замещающих роли понятий в пространстве свойств, посредством которых детализируется и упорядочивается внутриролевое описание. По видам свойства разбиты на непересекающиеся группы, состав которых определяется предметной областью, а каждая группа, в свою очередь, определяет собственные декомпозиционные возможности. Из-за избыточности естественно-языковых формулировок предусмотрено явное выделение их проблемных/целевых частей специальными указателями. Роль, вид свойства и указатели проблемных/целевых частей формулировок выражают внешнюю, а термины предметной области – внутреннюю семантику проблем/целей. Язык L_{in}^2, реализующий входной интерфейс с базой знаний (см. рис 2) разработан как упрощенная версия L_{in}^1.

В соответствии с принципом 4.1.2 исследована логико-семантическая формализация проблемно-целевого анализа [Lukiyanova, 2002]. Анализ показал достаточную адекватность модели, построенной на основе теории семиотических моделей [Осипов, 1995] и логики утилитарных оценок [Ивин, 1970]. Трехкомпонентная семиотическая система (см. рис. 2) включает формальную подсистему S_T, Ψ-механизм, настраивающий S_T на текущую ситуацию на кусте проблем/целей, О-преобразователь, преобразующий лингвистическое представление проблемы/цели в логико-лингвистическую формулу и наоборот:

$$O: p = (H/G)\, f_j [\, [\, [\, H/G\,]\, f_s\,] \ldots] \leftrightarrow \left\{ \begin{array}{l} (H/G)\, f_j [\, [\wedge [H/G\,]\, f_r] \ldots] \, [\supset f_7\,] \\ [\, f_s [\, [\wedge f_r\,] \ldots] \supset]\, (H/G)\, f_7, \end{array} \right\} \quad (5)$$

где p – лингвистическое представление проблемы/цели (альтернативы проблемы/цели заключены в фигурные скобки); f – ролевая фраза в p (j, r, s={1,2,…,6}).

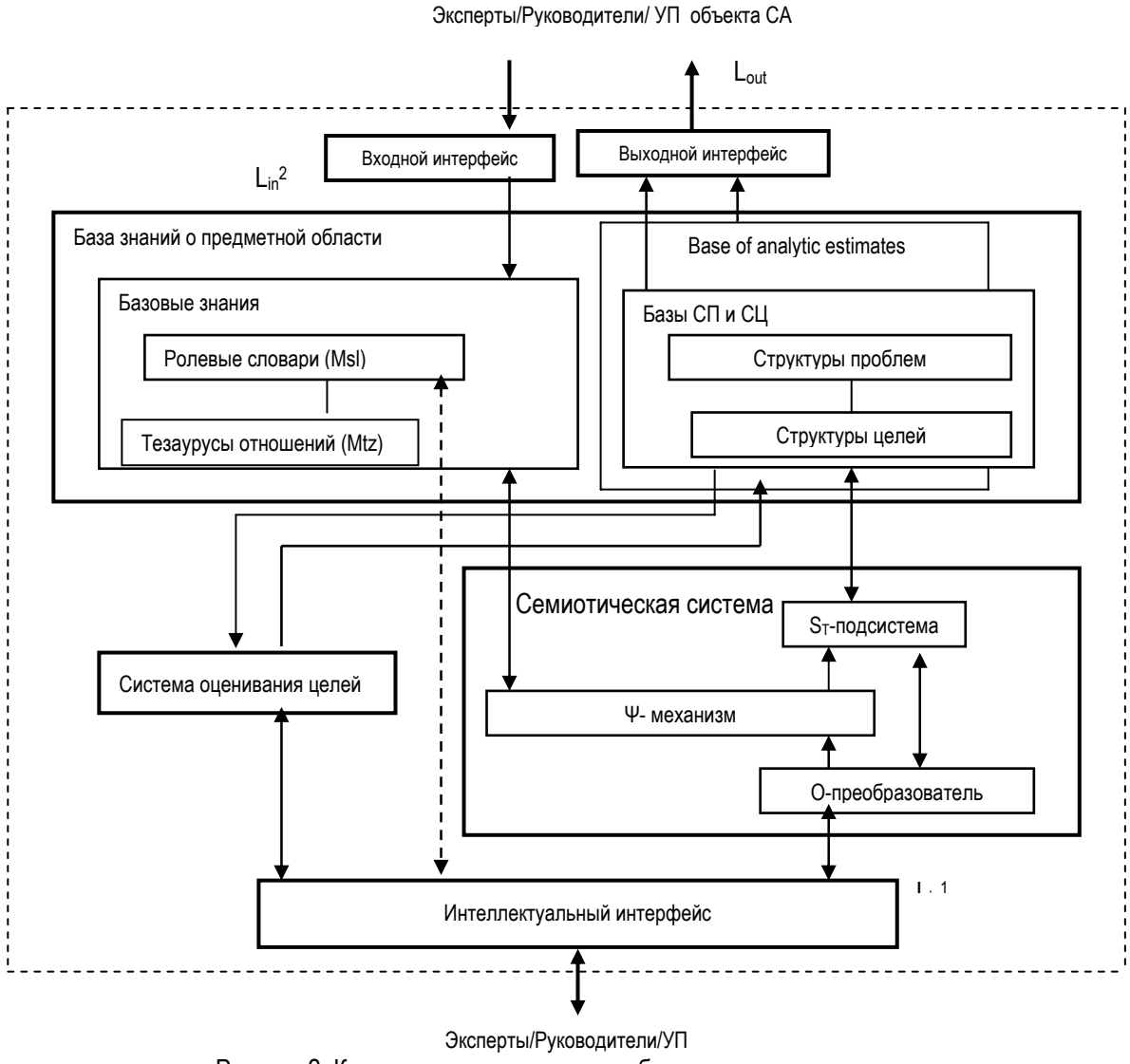


Рисунок 2. Концептуальная модель проблемно-целевого анализа

В соответствии с принципом 4.1.3 классифицированы ситуации на целях [Lukiyanova, 2002]. Получены 6 классов, один из которых является корректным, остальные определяют виды логических ошибок. В соответствии с принципом 4.1.4 предложен язык описания структур L_{out} для реализации интерфейса с базой СП/СЦ. Он основывается на теоретико-графовой древовидной модели, узлы которой описаны в L_{in}^1, и теоретико-множественном языке для описания (семантически) сложных дуг [Lukiyanova, 2002]. Посредством интеллектуального интерфейса семиотическая система получает от экспертов/УП

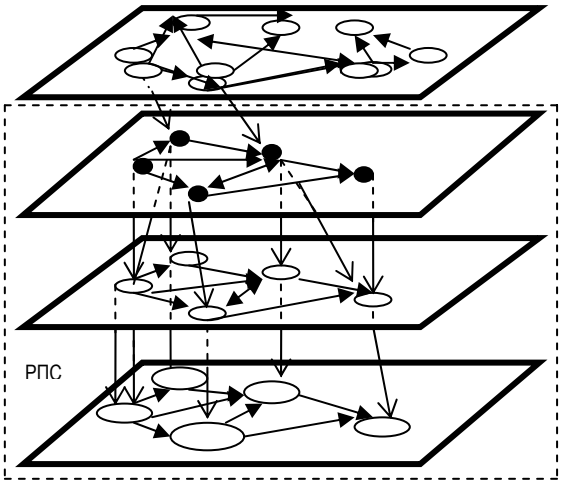
лингвистические описания проблем/целей, преобразует их с помощью О-преобразователя в логические формулы и проверяет посредством формальной подсистемы S_T корректность текущего куста СП/СЦ. При этом Ψ-механизм настраивает S_T на анализ текущего куста, используя соответствующий фрагмент базы знаний о предметной области в качестве собственных доменов. Если проверяемый куст противоречив или неполон, S_T идентифицирует ошибку и формирует рекомендацию по ее исправлению. Семиотическая система и база знаний о предметной области реализованы в Delphi.

Система оценивания целей реализует метод анализа иерархий [Saaty, Kearns, 1991] и может использоваться как для оценки отдельных проблем/целей, так и кустов из них и даже СП/СЦ в целом. Во время анализа куста проблем/целей S_T неизменяема и работает по шагам. Один шаг есть производимый по схеме $p_1 \mid\Rightarrow p_2$ (p_1 и p_2 – проблемы/цели) вывод, а возможные в соответствии с тезаурусом отношений базовых знаний семантические отношения между p_1 и p_2 являются условиями применимости правил вывода. В отличие от традиционных семантические отношения определяются как $< l_j, R_j >$, в которых первый компонент (l_j,) есть имя отношения, $l_j \in$ l, l – имена, выражающие реляционный базис Mtz предметной области [Lukiyanova, 2002]. Вывод осуществляется так: для $< p_1, p_2 >$ предполагается импликативная связь $p_1 \rightarrow p_2$, в которой p_1 полагается истинной, а соответствующий результат – абсолютно ценным; истинностное значение $p_1 \rightarrow p_2$ оценивается по базовым знаниям (см. рис. 2), и в случае истины по правилу отделения истинно p_2. Ложность $p_1 \rightarrow p_2$ означает противоречие, в этом случае S_T выводит рекомендуемую p_2'. Вывод упрощен за счет классификации ситуаций на $< p_1, p_2 >$ и $< p_1, p_2, p_i., \ldots, p_n >$.

Анализ критериев достижения целей основывается на полученной СЦ. Главный критерий обычно соответствует главной цели, локальные – локальным целям СЦ. Анализ функций ЦРС также основывается на СЦ. УП определяют функции ЦРС для каждой цели СЦ. Так оказывается сформированной СФ. Частично-формальный метод синтеза двух-, трехуровневой организационной структуры ЦРС приведен в [Лукьянова, 2001], [Lukiyanova, 2002]. Он заключается в систематизации списка СФ-функций на основе их внутрисистемной группировки по следующим признакам: субъект-объект, уровень и функция управления, характер производства и "жизненный цикл продукции" и др. В результате определяются функции управляющей и управляемой подсистем, затем в соответствии с общепринятыми правилами и нормами осуществляется группировка функций внутри каждой из подсистем ЦРС.

Заключение

Итак, исследована научно-техническая проблема СА промышленных организационных систем. Предложена новая методология структурно-целевого анализа систем. Методология охватывает все стадии СА, устанавливает закономерности ЦП, принципы структурно-целевого анализа, систематизирует используемые процедуры и получаемые результаты.



Социальная страта:
социальные показатели (например, необходимый уровень рыбной продукции).

Экономическая страта:
экономические отношения и показатели (например, минимальная прибыль от произведенной рыбной продукции).

Управляющая страта:
управляющие отношения и параметры (например, уменьшение простоев технологического оборудования).

Производственная страта:
производственные отношения и показатели (например, объем произведенной рыбной продукции).

Рисунок 3. Пример декомпозиции сложной проблемы: стратификация

Предложена концептуальная модель проблемно-целевого анализа систем, структура базы знаний предметной области, схема управления постановкой и решением проблем анализа, основанная на процедурах систематизации и классификации, частичной формализации формулировок проблем/целей/критериев/функций и их структур.
Методология используется в КЦП программ развития рыбной отрасли [Лукьянова, 1986], проблемные ситуации в отраслевом технологическом оборудовании [Лукьянова, 1988], в региональных рыбопромышленных системах (РПС). Так, в городской РПС были выявлены 43 проблемы, в ходе предварительного анализа которых зафиксированы семантическое пересечение проблем, уровень их неопределенности и сложности (статус, аспекты и виды рыбопромышленной деятельности, уровень и функции управления) проблем. В результате изменено число проблем (на 50) и содержание 10 проблем. Систематизация проблем позволила определить процент внешних проблем (10,5) и распределение проблем по видам рыбопромышленной деятельности и функциям управления. Так, большинство внутренних проблем составляют проблемы управления (55,5% от числа внутренних проблем) и экономические(26.5%). Среди управленческих проблем значительное число организационных (22,5%), проблем планирования (9.5%) и анализа (9%). Немало финансовых (13,5%) проблем. Систематизация обусловила корректную стратификацию, упрощенный результат которой приведен на рис. 3, и более обоснованный выбор экспертных групп.

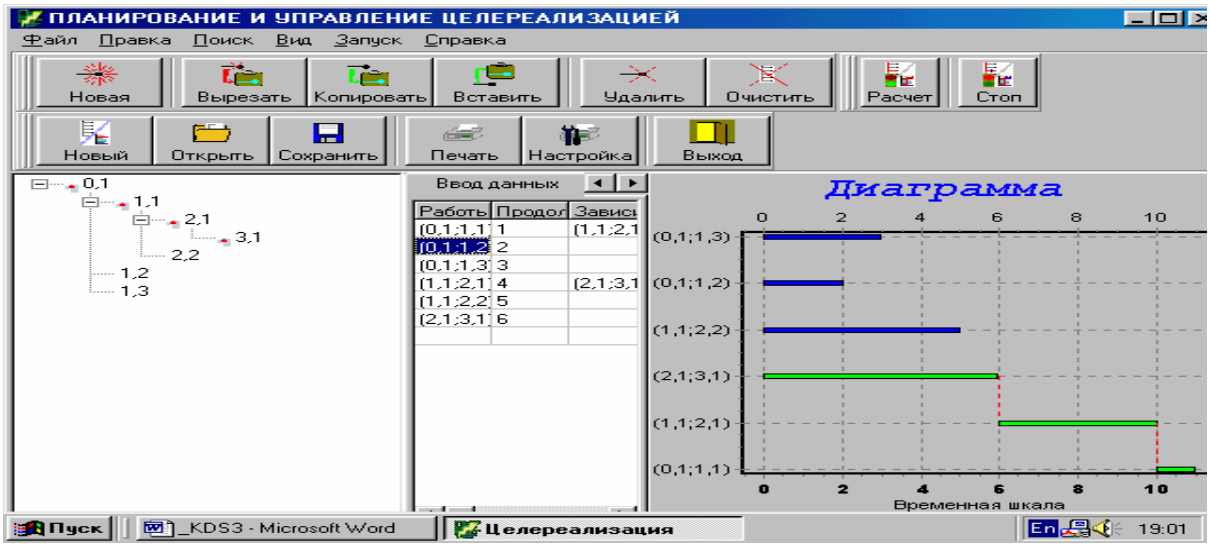


Рисунок 4. Пример промежуточной линейной диаграммы, синтезированной в соответствии с фрагментом СЦ

Затем были проанализированы причинно-следственные связи по каждому аспекту, виду деятельности, функции управления, что позволило определить главную проблему, анализ которой дал корректную СП. Аналогичным образом получена СЦ., после чего были проанализированы и синтезированы ЦРС и логически верная линейная диаграмма ЦР. Пример промежуточной диаграммы ЦР, базирующейся на фрагменте СЦ и полученной в ходе логико-лингвистического моделирования СЦ дан на рис. 4. Корректность структурно-целевого анализа подтверждена практикой решения проблем и экспертами, согласившимися со всеми выявленными ошибками ЦП и рекомендациями по их исправлению.

Литература

[Лопухин, 1971] М.М. Лопухин. ПАТТЕРН – метод планирования и прогнозирования научных работ. М., 1971.

[Волкова, Денисов, 2001] В.Н. Волкова, А.А. Денисов. Основы теории систем и системного анализа. СПб., 2001.

[Черняк, 1975] Ю.И. Черняк. Системный анализ в управлении экономикой. Экономика, М.,1975.

[Поспелов, Ириков, 1986] Г.С. Поспелов, В.А. Ириков. Программно-целевое планирование и управление. М., 1983.

[Перегудов, Тарасенко, 1989] Ф.И. Перегудов, Ф.П. Тарасенко. Введение в системный анализ. Высшая шк., М., 1989.

[Saaty, Kearns, 1991] Thomas L. Saaty, Kevin R. Kearns. Analytical planning: The organization of systems. N. J., 1991.

[Силич, Тарасенко, 1982] В.А. Силич, В.П. Тарасенко. Алгоритмизация и автоматизация процесса построения содержательной модели организационной системы и формирования на ее основе дерева целей. В кн.: Вопросы кибернетики. Методы и модели оценки развивающихся систем, с. 14-35. ВИНИТИ, М., 1982.

[Кондратов, Ростанец, 1982] В.А. Кондратов, В.Г. Ростанец. Применение программно-целевого метода в план-и развития науки и техники. В кн.: Совершенствование показателей плана, с. 153-165. НИИПиН, М., 1982.

[Романов, Клыков, 1974] В.Г. Романов, Ю.И. Клыков. Формирование дерева целей в системах ситуационного управления. Изв. АН СССР: Технич. к-ка, № 5, с. 11-15,1974.

[Nilson, 1973] Nilson N. Problem Solving Methods in Artificial Intelligence. 1973.

[Поспелов, 1995] Д.А. Поспелов. Ситуационное управление. Новый виток развития. Изв. РАН: Теория и системы управления, № 5, с. 152-158, 1995.

[Осипов, 1981-1982] Г.С. Осипов. Две задачи теории семиотических моделей управления. Изв. АН СССР: Техичн. кибернетика, № 6, с. 100-110, 1981; Изв. АН СССР: Технич. кибернетика, № 1, с. 131-137, 1982.

[Ивин, 1970] А.А. Ивин. Основания логики оценок. МГУ, М., 1970.

[Лукьянова, 1986] Л.М. Лукьянова. Метод структурирования целей (на примере структур целей для целевых программ). Изв. АН СССР: Технич. кибернетика, №3, с. 65-75, 1986.

[Лукьянова, 1988] Л.М.Лукьянова. Моделирование анализа проблем проектирования технологического оборудования. В сб.:Тезисы докл. Всесоюзн. конф. по искусств-му интеллекту, Т.1, с. 195-200. ВИНИТИ, М., 1988.

[Лукьянова, 2001] Л.М. Лукьянова. Система поддержки структурно-целевого анализа проблемных ситуаций. В сб.: Труды Междун. научно-практич. конф. KDS-2001, Т. I, с. 446-453. СЗГЗТУ, С.-Петербург, 2001.

[Lukiyanova, 2002] L.M. Lukiyanova. Methodology of Structure-aimed Analysis of Problem Situations in Organization Systems for Industry Branches. SPIIRAS Proceedings. Issue 1, v. 1, p. 297-315. SPIIRAS, SPb, 2002.

Сведение об авторе

Л.М. Лукьянова - Калининградский государственный университет, 236000, Калининград, Советский пр, 1, КГТУ, кафедраСУ и ВТ, Россия; e-mail: llm_llm@mail.ru

MANAGING INTERVAL RESOURCES IN AUTOMATED PLANNING

V.Poggioni, A.Milani, M.Baioletti

***Abstract**: In this paper RDPPLan, a model for planning with quantitative resources specified as numerical intervals, is presented. Nearly all existing models of planning with resources require to specify exact values for updating resources modified by actions execution. In other words these models cannot deal with more realistic situations in which the resources quantities are not completely known but are bounded by intervals. The RDPPlan model allow to manage domains more tailored to real world, where preconditions and effects over quantitative resources can be specified by intervals of values, in addition mixed logical/quantitative and pure numerical goals can be posed. RDPPlan is based on non directional search over a planning graph, like DPPlan, from which it derives, it uses propagation rules which have been appropriately extended to the management of resource intervals. The propagation rules extended with resources must verify invariant properties over the planning graph which have been proven by the authors and guarantee the correctness of the approach. An implementation of the RDPPlan model is described with search strategies specifically developed for interval resources.*

***Keywords**: AI, Automated Planning, Planning with resources, Propagation rule, Search strategies.*

Introduction

Various models have been proposed for extending the pure logical classical planning models in order to manage more real world features. A very promising issue toward this goal is the research line which aims to provide the planners with the ability of planning with resources. In this framework, in addition to the logical relationships among domain objects, operators and states, the planning models are able to cope with quantitative aspects of the world, such as actions which involves consumable/reusable resources, domain constraints on resources, goals involving quantities.

Several planning models for resources management have been proposed for extending virtually all the most successful planner approaches; among them it is worth noticing models UCPOP—like models [4], Graphplan-like [12], SAT—like [15,16], and also HTN based approaches [5].

The types and features of the modelled resources are also varying from unary and discrete resources [13] to reusable resources [4] and conjunctive constraints over resources [6,9]. A different approach is that of planners specialised in the management of time, as a quantitative resource; these planners allow the management of extension such as propositions which holds over time intervals , actions with durations and complex time numerical constraints [6,10]. The issue of a complex time management is beyond the scope of this work.

It is worth noticing that the introduction of quantitative resources in a planning framework has brought into planning some typical issues of scheduling and CSP, such as optimisation search and constraints management. Moreover having quantitative resources also change the typical view a planning problem can be regarded to. At the simplest level there are “pure logical” problem goals which can be specified as in the classical framework , nevertheless the plan generation phase will have to take into account of resources precondition/effects; the problem can otherwise specify “mixed” logical/quantitative goals, or even “pure quantitative goals”, (e.g. consider the problem of finding a plan for the purely quantitative goals: Consume at least 100 calories, Produce 1 Billion profit etc.); finally pure/mixed “optimisation” goals can also be specified, where no logical or quantitative goals exist (e.g. consider the problem: “producing as much profit as you can”), this latter optimisation aspect has been incorporated in PDDL 2.1 [7] where it is possible to specify an object function to be optimised.

Models of planning with resources certainly represent an important step toward a more accurate model of the real world, but, on the other hand, most of the proposed models fail to give any account of the potential uncertainty which can affect the quantities related to resources. Many facts in a real world state can be described in a satisfactory way by a boolean proposition (e.g. (on A table) (open door) etc.), but it is not very realistic to assume that an exact number can model the continuous quantities describing a given resource. Most models of planning with resources allow to describe non exact quantities in preconditions, such as, for example, an interval of values that the resources can assume in order make the action to executable (e.g. the fuel must be between 10 and 30, the voltage must be between 210 and 230, this preconditions can be modeled both in [12] as well in [9]), surprisingly the same planning models do not allow to specify intervals of values in action effects. In fact models of planning with resources admit updates and assignment operations which allow functional quantities (for example consumed_fuel can be functionally computed by distance * fuel_consumption) but where the increment of the current resource level is a single well determined numerical value (e.g. *consumed_fuel = distance * fuel_consumption = 12 * 0.25 =3* that is a single value) [7,9,12]. Indeed, it seems to be an apparent contradiction that the semantics of preconditions can be also given in terms of non exact quantities, while the semantics of effects have to be given only in term of precise values.

In this work we show how this gap between non exact preconditions and fully specified effects can be bridged by specifying in both cases quantities varying over intervals. RDPPlan, the model of planning with resources which we will describe, is based on DPPlan [1], a planner which uses a non directional search algorithm on the planning graph. RDPPlan is compatible with the resources model as described in standard PDDL 2.1, and extends it by allowing updates and assignments of quantities specified by intervals.

In the following paragraphs, after recalling the main features of DPPlan, it is introduced RDPPlan, the planning model with resources, showing that the addition of resources management does not have a great impact on the overall DPPlan approach and on the planning graph structure. Resources management in RDPPlan is realized by the modification of propagation rules and the introduction of appropriate rules for failure detection caused by resources constraints violation. Moreover, the structure and the algorithms we provide for resources management can be easily extended for update operations which operate over intervals, as shown in the fourth paragraph.

It is worth noticing that RDPPlan architecture is not committed to any particular search strategy (i.e. the resources management and failure detection is embedded into the propagation rules), the consequence is that search strategies can be easily added to RDPPlan. Strategies specifically developed for resources are described in paragraph five.

Examples and experimental results show that this approach seems to be appropriate for modeling real world situations where the consumption/production of resources cannot be expressed by a single value (for example, assuming that f*uel_consumption per Km* is a non exact quantity which ranges between *[0.25, 0.33]*, then *consumed_fuel* can be calculated as an interval by computing *distance * fuel_consumption = 12 * [0.25,0.33] = [3, 4]*).

Finally we point out some possible topics which are worth to be investigated in the RDPPlan framework, such as further strategies and heuristics for problems with resources, and further extensions to the resource model, like managements of fuzzy quantities.

DPPlan and its propagation rules

DPPlan is mainly based on GraphPlan [2]. With this planner it shares the same representation of disjunctive states, obtained by connecting facts and operators to form a graph, called *planning graph*.

DPPlan has, with respect to GraphPlan and other related planners, like IPP [11] or STAN, a completely different method for searching a solution in the planning graph.

The fundamental feature of DPPlan is that to each node of the graph a boolean value is assigned. An operator is *true* if it is executed, *false* if it is not. A fact is *true* if it is achieved by some operator, *false* otherwise. During the search phase a fact *p* can be *true* also if it is required by some operator *o* (*p* is a precondition of *o*) and *false* if it is required to be false (*p* is a negative precondition of *o*). This fact makes possible to use the propagation rules before the fact is really achieved and to cause, in the case, a backtracking earlier than it would be else obtained.

However those two situations are very different: if a fact *f* is *true* because something requires it, then *f* must be seen as a (sub-)goal and the current plan cannot be correct if this fact is not reached by any action. Only when an action achieving *f* is added to the plan, then *f* is really "*true*". The same distinction should be made between a fact which is really "*false*" (it has been deleted by some operator, or all of its achiever is false) and a fact required to be *false* (some action has it as a negative precondition).

In order to distinguish these situations, a further value, called *state*, is assigned to each fact:

- the state is "*produced*", when the fact value is reached (either true or false),
- the state is "*consumed*", if the fact value is required (either true or false), and
- the state is "*produced-after-consumed*" if the fact value has been required and then achieved.

This last value of state is useful during the backtracking phase.

In the previous paper of DPPlan we have described the rules by which it is possible to propagate a choice on the value of a node in the graph. These rules show how to update the other nodes of graph because of that choice. The operations of changing state and value of a node are named according to the type of changing and to the type of node they are applied.

The operation **use** sets the value of an operator node to *true*, while the operation **exclude** sets it to *false*.

The operation **consume** sets the value of a fact node to *true* and the state to *consumed*, **produce** sets the value to *true* and the state to *produced* (or *produced-after-consumed*), **consume-not** sets the value to *false* and the state to *consumed*, **destroy** sets the value to *false* and the state to *produced* (or *produced-after-consumed*).

All of these operations can fail if they try to give a different value to an already valued node (e.g. **destroy** and **consume** on the same fact) or can be ignored if they try to give the same (or compatible) value or state (e.g. **produce** and then **consume** on the same fact). For further details see the original paper [1].

For an operator *o*, *In(o), NotIn(o), Out(o)* and *NotOut(o)* are respectively the list of its positive preconditions, negative preconditions, positive effects and negative effects. For a fact *f*, *In(f), NotIn(f), Out(f)* and *NotOut(f)* are respectively the list of the operators which have *f* as a positive effect, as a negative effect, as a positive precondition and as a negative precondition; moreover *npp(f)* and *npd(f)* are respectively the number of possible

producer (destroyer), i.e. the number of elements of list *In* and *NotIn* whose value is still undefined. Finally, for any node *n*, *Mutex(n)* is the list of the nodes exclusive of *n*.

These propagation rules update the state and the value to the node they are applied, as well as the goal list. Note that without any particular additional procedure, DPPlan is able to solve problem with negative preconditions and goals, and with temporally qualified "*initial states*", like the fact *f* is true at time $t > 0$, and goals, like the goal *g* has to be achieved at time $t < t_{max}$.

The main algorithm operates in way resembling the celebrate Davis-Putnam algorithm for propositional logic. At the beginning all the variables receive the value *undefined*, then the procedure *produce* is performed for all the facts in the initial state, *consume* is performed for all the positive goals, and *consume-not* is performed for all the negative goals.

In its main loop, the algorithm chooses an undefined variable *v*, tries to set its value to one of the boolean values (e.g. *true*) until it finds a solution (no fact is in the state *consumed*), or some propagation fails, in this case a backtracking phase is performed by undoing all the propagations done after *v* and tries to set the value *v* to the opposite value (e.g. *false*). If a failure is obtained again, *v* can be neither *true* nor *false*, therefore the backtracking stops until the previously tried variable is unset: now the value of this variable is reversed and the search goes on. Note that when the algorithm has tried both the value for the first chosen variable, without reaching a solution, the search phase is ended, the graph is augmented with all the new applicable operators and all the new facts they produce, in a way similar to the expansion phase of GraphPlan, and a new search phase is performed.

What is completely free in our algorithm is how to choose what variable to try next and which value to try first. According to the method used, the planner can perform a forward search, a backward-chain search, a bidirectional search or simply a non directional search.

In the original paper [1] we have listed several ways of choosing a variable, essentially an operator to be tried to *use*, and then to *exclude*.

RDDPlan: Planning with numerical resources in DPPlan

RDPPlan is the extension of DPPlan to handle numerical resources. In the model together with the logical propositions, the use of numerical variables is allowed. Each numerical variable, called *resource*, can be cited as in preconditions and goals as well as in effects and initial states. Conforming to the PDDL 2.1 [7], resources are represented as a numerical functions whose parameters can be domain constants or action parameters.

Preconditions and goals

In preconditions and goals it is allowed to use conditions like `(compare resource value)` where `compare` is a comparison operator `(>, <, >=, <=, =)` and `value` can be an expression involving action parameters, numerical constants and arithmetic operators.

Since we do not allow for disjunctive preconditions and goals, several constraints on the same resource *r* reduce to a unique real interval, possibly empty, indicated with $P_{A,r}$ for the precondition of action *A* and with G_r for goals. These intervals are possibly unlimited in left and/or in right side.

Initial state and effects

In the initial state the resources are initialized by the proposition `(= resource value)` where now the value can only be a numerical constant.

In the effects a resource can be changed by proposition `(change resource value)` where `change` can be one of the operators `assign, increase, decrease` and `value` is an expression as in the preconditions. We indicate the value added of action *A* to resource *r* with $E_{A,r}$, intending $E_{A,r}$ =*value* for *increase* and $E_{A,r}$ = -*value* for *decrease*. For an action *A* which does not change a resource *r* we treat *A* as an increase operator of value 0, i.e. $E_{A,r}=0$.

As a syntactic sugar we allow to use, while expressing preconditions and effects, in the expressions called *value* some other resources, provided they are static, i.e. initialized in the initial state, but not changed by any action. A static resource is a sort of constant which remains unchanged during the plan, like weight function in domain *Depots*.

This restriction has the main effect that each resource can be changed independently from the others. Allowing to cite other (non static) resources in the preconditions would have generated more complex admissible resource domains, not reducible to the cartesian product of real intervals. On the other hand if an effect on resource *r* could depend on the value of some other (non static) resource, some interaction between resources would have arisen which are difficult to handle (e.g. increasing a resource can cause to another resource to decrease).

Realized value and desired interval

Associated to each resource *r* and each time level *t*, a numerical value R_{rt} and a real interval D_{rt} is computed during the planning phase. Every time an action is selected by the search procedure, R_{rt} and D_{rt} are updated for every resource *r* used by the action and for every time level *t* greater or equal to the time level at which the action is selected. The previous values of R_{rt} and D_{rt} are restored during the backtracking phase.

The number R_{rt} represents the current value of the resource *r* at time *t* realized by the actions since now inserted in the plan. At the start of search procedure, for each resource *r* and for every time *t*, R_{rt} is set to the value specified at the initial state.

The interval D_{rt}, called "*desired interval*", contains all the admissible values for the resource *r* that allow the execution of all the actions selected at time level *t*. At the start of search procedure for each resource *r* and for every time $t < T$ (*T* is the last time level), D_{rt} is set to $[-\infty,+\infty]$, while for each resource *r*, D_{rT} is set to G_r, the interval specified by the goals.

Solution plans and executability

Definition 1 *The obvious sufficient and necessary condition for a plan to be executable and to be a solution of the given planning problem is that for each resource r and for each time level t the condition* $R_{rt} \in D_{rt}$ *holds.*

Before describing the rules with which R_{rt} and D_{rt} are updated, we must define when two or more actions are executable at the same time level. We use the same concept of simultaneous executability as expressed in [7,15].

Definition 2 *A set of actions* $A_1, A_2, \ldots, A_m$ *is simultaneously executable if for every permutation* Π *of the actions in the set* $A_{\Pi(1)}, A_{\Pi(2)}, \ldots, A_{\Pi(m)}$

1) $A_{\Pi(1)}$ *is executable in the current state,* $A_{\Pi(2)}$ *is executable in the state after the execution of* $A_{\Pi(1)}$, $A_{\Pi(3)}$ *is executable in the state after the execution of* $A_{\Pi(1)}$ *and then* $A_{\Pi(2)}$, *and so on,*

2) the effect over the resources is always the same.

As a straightforward consequence of the second condition, an assignment on resource *r* is not simultaneously executable with any action changing *r* (additive operator).

The question remains open whether to allow an action changing *r* to be simultaneous with any action having a precondition with respect to *r*. Our approach is to allow simultaneity whenever the change does not affect the executability.

Proposition 1 *It is easy to prove that two additive actions* A_1 *and* A_2 *are simultaneously executable (with respect to r) if, let* $[\alpha_1,\beta_1]=P_{A1,r}$ *and* $[\alpha_2,\beta_2]=P_{A2,r}$ *be respectively the precondition intervals and let* $k_1=E_{A1,r}$ *and* $k_2=E_{A2,r}$ *their effect on r, we have* $\alpha \le \beta$ *where* $\alpha=\max\{\alpha_1, \alpha_1-k_2, \alpha_2, \alpha_2-k_1\}$ *and* $\beta=\min\{\beta_1, \beta_1-k_2, \beta_2, \beta_2-k_1\}$. *In the positive case we set* $D_{rt}=[\alpha, \beta]$ *if actions* A_1 *and* A_2 *are the only actions to be executed at time t. Otherwise* A_1 *and* A_2 *are marked to be mutually exclusive.*

The generalization to the case of many additive (over the same resource *r*) actions $A_1, A_2, \ldots, A_m$ is somehow straightforward.

Proposition 2 *Called* $[\alpha_i,\beta_i]=P_{Ai,r}$ *the interval precondition and* $k_i=E_{Ai,r}$ *their effects, for i=1,…,m, we have that the action* A_i*'s are simultaneously executable (with respect to r) if* $\alpha \le \beta$ *where* $\alpha = \max\{\alpha_i - km_i : i=1,\ldots,m\}$, $\beta = \min\{\beta_i - kp_i : i=1,\ldots,m\}$ *and* $kp_i = \sum_{j \neq i, k_j > 0} k_j$ *and* $km_i = \sum_{j \neq i, k_j < 0} k_j$ *for i=1,…,m.*

In the positive case $D_{rt}=[\alpha, \beta]$ *if the actions* A_i*'s are the only* actions to be executed at time t..

α and β can be computed in linear time (in the number of resources and actions) by storing (and keeping updated) the values $kp_0 = \sum_{k_j > 0} k_j$ and $km_0 = \sum_{k_j < 0} k_j$.

The case of several additive actions includes the case of simultaneous execution of additive actions on resource *r* (possibly none) with other actions which do not change *r*. In the particular case, where all the actions do not change *r*, D_{rt} reduces to the intersection of all the precondition intervals.

A case not yet covered is the simultaneous execution of an assignment on *r*, say A_1 whose assigns to *r* the value *v*, with with actions $A_2,\ldots,A_m$ which do not change *r*. It is easy to see that $A_1,A_2,\ldots,A_m$ are simultaneous executable (with respect to *r*) if the intersection of all the precondition intervals is not empty and contains *v*.

After a new action *A* is selected to be *used* at time *t*, we must check if it is simultaneously executable with the already selected actions at time *t*, by computing for each resource *r* the quantities $\alpha(r)$ and $\beta(r)$. Only when each of these interval is not empty, then the desired interval for *r* is set to be $[\alpha(r), \beta(r)]$. Moreover an incremental way, which takes only a constant time to be computed, of updating D_{rt}, after the *use* of a new additive operator *A*, is the following.

Proposition 3 *If* $D_{rt} = [\alpha,\beta]$, $P_{A,r} = [a,b]$ *and* $E_{A,r}=k$, *then the updated desired interval is* $[\alpha',\beta']$ *where*

$\alpha' = max\{a - km_0, \alpha - min\{k,0\}\}$ *and* $\beta' = min\{b - kp_0, \beta - max\{k,0\}\}$.

Updating the values R_{rt} is done by recomputing them for every resource changed by *A*, starting from time *t+1* and ending at the first time where an assignment over *r* is selected. This computation can be efficiently done by storing and keeping updated the total amount to be added to *r*, computed considering all the actions selected since now.

Extension to interval resources

A very straightforward, yet significant, extension of the simple model above explained is to allow for non completely specified initial states and effects.

Interval on the initial states and effects

Instead of initializing a resource with a unique real value, we allow to specify a real interval I_r as a range for the initial value of the resource *r*. The planner operates in an under-specified domain in which the value of some resource is not exactly known, but it is bound to be in an interval. Suppose we do not know exactly how much gasoline is in the tank of our car: we just know that it surely the real amount is between 5 and 10 liters.
Similarly it is possible to have under-specified effects of any operator: the value which is added, subtracted or assigned to the current value of a resource is not exactly known, but only a lower and an upper bound is specified. Imagine that the car in the previous example, we do not know which is the exact consumption: all we know is that the car can travel from 10 to 15 kilometers per liter.

In this enhanced model, the real quantities $E_{A,r}$ and R_{rt} are therefore replaced by real intervals, which cannot be unlimited in the left or in the right side. The intervals R_{rt} are initialized with I_r, the intervals specified in the initial state description, and are updated according the following simple rules, where the current interval for R_{rt} is $[\gamma,\delta]$, the operator to be executed is *A* and $E_{A,r} = [e_{min}, e_{max}]$: if *A* is `ASSIGN` R_{rt} becomes $[e_{min}, e_{max}]$, if *A* is `INCREASE` R_{rt} becomes $[\gamma+e_{min},\delta+e_{max}]$ and if *A* is `DECREASE` R_{rt} becomes $[\gamma-e_{max},\delta-e_{min}]$.

Solution plans and executability

The definition of what solution plan is meant is similar to the definition described in the previous section.

Definition 3 *For this model the necessary and sufficient condition for a plan to be a solution of a given planning problem is that for each resource r and time level t,* $R_{rt} \subseteq D_{rt}$.

The intended semantics of this meaning of the term *solution plan* Π is that if for every possible way of replacing each interval effect $E_{A,r}$ with a number $e_{A,r} \in E_{A,r}$ and of replacing each initial interval I_r with a number $i_r \in I_r$, the problem so obtained, which now is conform to the previous model, is solved by Π, according to the semantics expressed in the previous section. Expressed in other terms, a solution plan must solve every possible problem that is allowed by the constraints specified in the initial state and in the effects description.

The update rules for desired interval are similar to what we have seen in the previous section.

Proposition 4 *Suppose that* A_1 *and* A_2 *be two additive actions with precondition intervals* $P_{A1,r}=[\alpha_1,\beta_1]$ *and* $P_{A2,r}=[\alpha_2,\beta_2]$ *and with effect intervals* $E_{A1,r}=[e_{min,1},e_{max,1}]$ *and* $E_{A2,r}=[e_{min,2},e_{max,2}]$ *respectively. If* $\alpha \leq \beta$, *where*

$\alpha = max\{\alpha_1, \alpha_1-e_{min,2}, \alpha_2, \alpha_2-e_{min,1}\}$ *and* $\beta = min\{\beta_1, \beta_1-e_{max,2}, \beta_2, \beta_2-e_{max,1}\}$, *then* A_1,A_2 *are simultaneously executable.*

The generalization to the cases with m additive or multiplicative actions are the following.

Proposition 5 Called $P_{Ai,r} = [\alpha_i, \beta_i]$ the precondition intervals, for i=1,…,m and $E_{Ai,r}=[e_{min,i}, e_{max,i}]$ the effect intervals over r for i=1,…,m, we have that the action A_i's are simultaneously executable if $\alpha \leq \beta$, *where*

$\alpha = max\{\alpha_i - km_i : i=1,\ldots,m\}$, $\beta = min\{\beta_i - kp_i : i=1,\ldots,m\}$ *and* $kp_i = \sum_{j \neq i, e_{\max,j}>0} e_{\max,j}$ *and* $km_i = \sum_{j \neq i, e_{\min,j}<0} e_{\min,j}$.

Strategies on Resources

In this section we present the strategies that we have defined for achieving the goals over resources. These strategies are necessary to solve "*pure numerical problems*", i.e. problems with goals only on resources. The methods that implement these strategies are combined with the ones for solving logical goals, by evaluating the difficulties of resources and logical goals and selecting the most difficult goal to solve.

Strategy for numerical resources

For each time step *t* and for each resource *r*, we check if the condition $R_{rt} \in D_{rt}$ holds: the negative cases are the goals on resources. An action that can help solving a goal on resource *r* at time *t* can be chosen according to the following rules.

First, the algorithm searches for an action that can achieve the goal in one only step, preferring, in the case of many available options, the action situated at the level nearest to *t*. If such an action does not exist, we choose an increaser (actions which make R_{rt} bigger) or a decreaser (actions which make R_{rt} smaller) according to the case. In the detail, a first search is performed for all those actions *A* present at any time $\tau < t$ such that the updated value (if *A* would be executed) of R_{rt}, say R'_{rt}, verifies the condition $R'_{rt} \in D_{rt}$. Among those, the action *A* with the highest τ is selected, by performing the search starting from the time-step *t–1* and going backward: the first action found is chosen.

If the first search fails, the algorithm chooses an action that can permit us to come closer to the goal. Called $[\alpha,\beta]$ the interval D_{rt}, the algorithm selects an increaser *A*, if $R_{rt} < \alpha$, or a decreaser *A*, if $R_{rt} > \beta$, that minimizes $\min\{|\alpha - R'_{rt}|, |R'_{rt} - \beta|\}$.

Strategy for interval resources

When we work with resources as intervals using the "interval algebra" explained in a previous section, we have to handle with real intervals whose width can in general only grow, except when an assignment is performed. In fact it is obvious that any numerical operation between intervals produce as a result an interval which has a width larger than the original widths.

The following example can show this characteristic. If we have in the tank an amount of fuel that we do not know exactly, but that we know be between *10* and *15* liters (this is a case of incomplete knowledge in initial state) and we take from an another tank an unknown amount of fuel between *12* and *16* liters (example of non deterministic effects over resource), then the minimum amount of fuel in the first tank is *22* liters and the maximum is *31* liters. So, in the notations here used, from the rule $R_{rt} + E_{A,r} = R_{r,t+1}$, we will obtain *[10,15] + [12,16] = [22,31].* This means that we have at time step *t*, before the action application, a realized interval with width *5* and then, at the next time-step *t +1*, an interval with width *9*.

If you think that the width of realized interval represents, in some sense, the indetermination on resource value, we have that the larger the interval width, the larger the indetermination. Moreover note that if the width of the realized interval is large, it is more difficult that the solution conditions $R_{rt} \subset D_{rt}$ will hold. Let $|R|$ denote the width of interval *R=[a,b]*, i.e. $|R|=b-a$. The first control to do is on the widths of realized and desired intervals.

1) If $|R_{rt}| > |D_{rt}|$ an assignment which assigns an interval with width less than $|D_{rt}|$ is the only possible choice. If there are many such assignments, the algorithm chooses that one which assigns the interval with the least width. If there are no assignments with this property, a backtracking is necessary.
2) If $|R_{rt}| \leq |D_{rt}|$ the algorithm tries to solve this goal using a procedure similar to that described in the previous section. Now the choice criteria takes into account the distance between D_{rt} and R'_{rt} (which is the realized interval updated after the execution of the action to be evaluated) and their widths.

The previously described criteria can be implemented by defining two preference functions, one for the interval widths, and one for the distance between intervals, and by searching for an action that maximizes a linear combination of the functions. The first function is $f_W(A, D_{rt}) = \frac{|D_{rt}| - |R'_{rt}|}{|D_{rt}|}$ and gives to each action *A* a numerical positive score between 0 and 1.

The second function is a decreasing function of the distance between the middle points of the two intervals $f_D(A, D_{rt}) = \exp(-\frac{1}{2}|\gamma + \delta - \alpha - \beta|)$ where $[\alpha,\beta] = D_{rt}$ and $[\gamma,\delta] = R'_{rt}$.

Also this function gives to each action *A* a numerical positive score between 0 and 1.

Experimental results show that good values for the ratio c_1/c_2 of the coefficients of the linear combination $f(A, D_{rt}) = c_1 f_W(A, D_{rt}) + c_2 f_D(A, D_{rt})$ are between 1 and 2.

A theoretical justification is that f_W is slightly more influent for driving the search algorithm, because it can be useless to get closer to the desired interval if the width of the realized interval is too large.

Conclusion

RDDPlan a model of planning with interval resources based on propagation rules has been described. This models seems to be more adequate than existing models of planning in order to describe real world operators which uses resources because it does not require a complete knowledge of the quantity to be updated, but an interval boundary.

The main contribution to RDPPlan comes from [1], whose propagation and failure rules have been extended with interval management. Other related works share with it a similar planning graph structure with a different semantics for resources and no management of intervals [12,15]. Although an actual planner and strategies for resources have been implemented on the basis of the proposed model, RDPPlan can be considered a platform on which strategies and heuristics for planning with resources can be experimented. Further investigations and experiments are planned in order to develop more accurate heuristics and strategies which takes into account of resources, moreover, in order to provide a meaningful evaluation it will be also required the development of a set of significant benchmarks for planning domains with interval resources.

Finally it is worth investigating further extensions to the resources model more accurate with respect to the uncertainty in the real world e.g. intervals with given probability distribution over resources values and fuzzy quantities.

Bibliography

[1] M.Baioletti,S.Marcugini,A.Milani. DPPlan: an algorithm for fast solution extraction from a planning graph. *In Proc. of AIPS-00*, 2000.

[2] A.Blum,M.Furst. Fast planning through planning graph analisys. *Artificial Intelligence (90):281-300*, 1997.

[3] B.Bonet,H.Geffner. Planning as heuristic search. *Artificial Intelligence 129*, 2001.

[4] S.Chien et al. ASPEN automated planning and scheduling for space mission operation. *In Proc of SpaceOps2000*, 2000.

[5] K.Currie,A.Tate. O-Plan: The open planning architecture. *Artificial Intelligence (52):49-86*, 1991.

[6] M.Do,S.Kambhampati. Sapa: A domain indipendent heuristic metric temporal planner. *In Proc. of ECP-01*, 2001

[7] M.Fox,D.Long. PDDL 2.1: An extension to PDDL for expressing temporal planning domains. Forthcoming in JAIR special issue on 3rd International Planning Competition

[8] J.Hoffmann,B.Nebel. The FF planning system: Fast plan generation through heuristic search. *Journal of Artificial Intelligence Research (14): 253-302*, 2001.

[9] J.Hoffmann. Extending FF to numeric state variables. *In Proc. of ECAI-02*, 2002.

[10] A.Jonsson et al. Planning in the interplanetary space : theory and practice. *In Proc. of AIPS-00*, 2000.

[11] J.Koehler,B.Nebel,J.Hoffmann,Y.Dimopoulos. Extending planning graphs to an ADL subset. *In Proc. of ECP-97*, 1997.

[12] J.Koehler. Planning under resource constraints. *In Proc. of ECAI-98*, 1998.

[13] P.Laborie,M.Ghallab. Planning with sharable resource constraints. *In Proc of IJCAI-95*, 1995.

[14] X.Nguyen,S.Kambhampati,R.S.Nigenda. Planning graph as the basis for deriving heuristics for plan synthesis by state space and csp search. *ASU Technical Report*, 2002.

[15] J.Rintanen,H.Jungholt. Numeric state variables in constraint based planning. *In Proc. of ECP-99*, 1999.

[16] S.Wolfman,D.Weld. The LPSAT engine and its application to resource planning. *In Proc. of IJCAI-99*, 1999.

Authors information

Marco Baioletti – Dipartimento di Metodi Quantitativi, Univeristà degli Studi di Siena, P.zza S.Francesco 5, Siena, Italy

Alfredo Milani - Dipartmento di Matematica e Informatica, Università degli studi di Perugia, Via Vanvitelli, 06100 Perugia, Italy

Valentina Poggioni – Dipartimento di informatica e Automazione, Università degli Studi di Roma Tre, Via della Vasca Navale 79, 00146 Roma, Italy

PLANNING OF INTELLECTUAL ROBOT ACTIONS IN REAL TIME

N. Romanenko

Summary: *In article the mathematical model of the mobile robot actions planning at recognition of situations in extreme conditions of functioning is offered. The purpose of work is reduced to formation of a concrete plan of the robot actions by extrapolation of a situation and its concrete definition with the account a priori unpredictable features of current conditions.*

Key words*: the mobile robot, recognition of a situation.*

Introduction

Creation of the intellectual mobile robots, capable to adapt and plan the actions in conditions of aprioristic uncertainty of dynamically changing habitat, is one of the important strategic problems of modern techniques. Absence of preliminary environment formalization, and also presence of any way moving obstacles and purposes in it complicates the use of automatic control traditional methods. The given circumstance stimulates development of new control systems with presence on mobile robots (MR) board of situation recognition system on the basis of the multiprocessing computer with elements of artificial intelligence that provides adaptability of MR behavior in an environment.

Recognition of situations is the new area of cybernetics. The closest area is images recognition. But there is a basic distinction of these concepts: the image is "static", and the situation is dynamical, recognition of situations is always connected to a prediction (extrapolation) that usually does not happen in the theory of images recognition. At situation recognition there is no aprioristic classification as the number of possible situations is unlimited, but results are classified and have the final alphabet.

For MR control system we shall understand that the situation is a set of events, developing in time and space limited in radius of its action, and having the important consequences from the point of view of the chosen criterion function. The situation includes three basic components:

- the ground conditions fixed during the certain moment of time (presence of obstacles in a way);
- processes which can occur both with its condition, and with MR condition (dynamism of obstacles and the robot);
- result or possible consequences (planning of actions and forecasting).

To distinguish a situation by control system - means to develop decision about result of further MR movement on the basis of environment and proceeding process information.
Traditional mathematical models of MR management in extreme conditions of functioning becomes insufficiently as MR proper response to change of situations is not described, especially at occurrence of obstacles. The given problem was examined by many researchers, and there exist various ways of its decision [1-3].
Planning of actions is the major function of the mobile robots independently working in dynamic and uncertain environments. Scripts [4] are effective model of knowledge representation in such systems. Scripts represent the generalized description of sequence of MR actions in some stereotyped situation, allowing to achieve a required target condition. Formation of a concrete plan of action is carried out by a choice of one of possible situations and its concrete definition with the account a priori unpredictable features of current conditions. At functioning in the uncertain environment, time restrictions on decision-making are a priori unknown; therefore the robot should possess ability to adapt time of decisions to dynamics of processes occurring in the environment. Known models of representation and recognition of situations do not support such opportunity.

Formal representation of actions plan

For representation of robot actions plan we shall use hierarchical frame structure of the following kind. At the top-level plan FPm is set by the frame of a kind:

$$FP_m = (PN_m, S_0, Act_1, S_1, Act_2, S_2, ..., S_{n-1}, Act_n, S_n),$$

where PN_m - a name of *m* plan; S_0 - the current condition of environment; Act_i - the frame of i action of the plan; S_i – the frame of environment condition after performance *of i* action of a plan; S_n - the target condition of environment. Frames of environment conditions and actions of the robot contain slots «Type of the frame», «Name of the frame» and set of frames of the bottom level serving for representation of parameters of conditions and actions.

Parameters of environment conditions can be divided on the following groups:

- external world condition parameters, which value do not depend on robot actions;
- environment condition parameters and the robot in the environment, changeable as a result of robot actions;
- robot inwardness parameters, describing internal robot resources.

Let MR, moving in a priori unknown to it environment, to find out on the way an obstacle and as one of possible variants of actions to consider a detour of an obstacle on the right. In this case it is possible to allocate two stages of obstacle overcoming: turn to the right and alignment of movement. Accordingly plan of MR action «Obstacle overcoming» contains two frames of action with names "To the right" and "Directly" and three frames of condition: «Before an obstacle», "Alignment" and «Behind an obstacle». It is obvious, that success of realization of the given plan is defined by ability of the robot to make a detour of an obstacle and entrance on a target trajectory. The successful detour depends on MR maneuverability and speed. In its turn speed depends on capacity of engines and density of a ground in a detour place, i.e. is defined as internal robot opportunities (ability to provide required speed at turn), and external conditions (character of a ground on a trajectory).

Statement of decision-making task in real time on the basis of actions planning

Under generalized problem situation (GPS) we shall understand the generalized description of environment condition in which the robot is required to make some decision. GPS is identified by the name; for example, GPS «Obstacle overcoming» corresponds to an above-mentioned example.

Generally in some problem situation the robot has not unique variant of possible actions and, accordingly, a plan of action subject to the analysis. So for an above-mentioned example the robot can, along with the script «Detour on the right», to consider also the script «Detour on the left». Decision-making with use of actions planning in real time assumes definition of the most effective in the given situation way of actions for limited time. Thus the stock of time T_d for decision-making should be defined dynamically, proceeding from the analysis of the current situation.

Proceeding from the aforesaid, process of decision-making by the intellectual robot in system of real time on the basis of actions planning includes the following steps:

1) preliminary estimation of a situation and definition of the general stock of time T_d for decision-making;
2) definition of actions variants set possible in the given situation;
3) distribution of the general budget of time for tasks of various actions variants estimation;
4) concretization and estimation of actions planning efficiency;
5) actions variants comparison and choice of the best of them.

Let's consider all listed steps of decision-making.

1. Definition of the general stock of time.

Time restrictions for decision-making by the robot are caused by possible approach of events undesirable to the robot if it in due time will not undertake corresponding actions. Time of approach of events is determined by dynamics of environment processes (in particular, actions of mobile obstacles) and a priori is not known. Thus, the robot should possess ability to define dynamically the time of critical event approach on the basis of forecasting of possible consequences of the current situation with use of knowledge of laws of various environment processes.

Set of possible in the future critical events $\{CE_i\}$ can be put in conformity with every GPS. For example, in a situation with obstacle overcoming (the detour on the right), critical event for MR moving is presence of a wall or a hole to the right as an obstacle. The robot stops before the obstacle if movement is impossible owing to the appeared dynamic obstacle. Function $f_d^i(P_j, ..., P_k)$ from predicates of the current situation, calculating stock of time T_d caused by possible approach of the given critical event can be put in conformity to each critical event.

The kind of this function is known at a stage of MR control system construction, and its description is stored in corresponding slot of action plan. The valid value of a stock of time T_d is calculated dynamically on the basis of the current values of situation parameters.

2. Definition of a set of possible variants of actions.

Every GPS at a stage of construction of base of MR knowledge puts in conformity set of tactical variants of actions submitted by plans of action $\{FP1, \ldots, FPq\}$. Each such plan has slot "Precondition" in which predicate *PC (precondition)* is written down, determining additional conditions of the given plan applicability. The predicate is determined on parameters of the current situation (both external, and inwardnesses of the robot) and allows to exclude some plans from the further consideration. For example, the detour of an obstacle at the left can be impossible because of a reservoir, taking place there, movement can be stopped owing to a steep slope of a line, etc.

Calculation of the given predicate is realized by function $g(P_j,...,P_k)$. Computing complexity of this function should be low and the top estimation of its calculation time should be known. Calculation of predicate *PC* can demand gathering of the additional information for reception of facts, which are not contained at present in a database. As a result of definition of the validity of preconditions of all plans contained in $\{FP1,\ldots,FP_q\}$ the reduced set $\{FP_m\}$ of possible plans for the further analysis is formed.

3. Distribution of the general stock of time between tasks of action variants estimation.

As the estimation of efficiency of all tactical plans from set $\{FP_m\}$, should be executed in time T_d , it is necessary to allocate the general stock of time between corresponding tasks of a concretization.

Various plans of actions, generally, demand various time of a concretization and this time depends on parameters of the current situation and it is not known at a stage of knowledge base construction. Besides time of action plan concretization can vary over a wide range not only depending on an external situation, but also on internal resources of the robot. For example, the concretization of the script «Detour of an obstacle» assumes scanning a surface of road for definition of roughnesses and density of a ground with the purpose of definition of an optimum trajectory of movement. It is obvious, that time of this task decision depends on the area of the scanning, the

current condition of touch and MR processing resources, and other parameters which values are a priori unpredictable. At the same time, set of such parameters can be allocated beforehand and for each concrete plan FP_m the bottom estimation of decision time of task of concrete definition T_m^* can be expressed as function from these parameters:

$$T_m^* = h_m(P_l, ..., P_k)$$

These functions should have low computing complexity. Using the given functions, MR calculates the bottom estimation of total time of a concrete definition of all plans belonging to set $\{FP_m\}$ on the basis of the current situation:

$$T_\Sigma^* = \sum_m T_m^*$$

Then time allocated for concretization of *i* plan of action, will be determined as follows:

$$T_d^{(i)} = T_d * (T_i^* / T_\Sigma^*)$$

4. Concretization and estimation of actions planning efficiency.

Concretization of action plan Fp_i is reduced to a concretization of all of its steps - frames - actions. This task should be solved for limited time $T_d^{(i)}$ Tasks of search of optimum values of action plan parameters depend on the concrete mathematical models used for the description of corresponding steps of the plan, and can be the diversified. However, not narrowing the generality of consideration it is possible to count, that these tasks are under construction as a task of undefined time [5].

Thus the set $\{v_i^k\} = \{(T_i^k, Q_i^k)\}$ variants of its decision distinguished by time T_i and quality Q_i of received result should be put in conformity to each single task of step concretization. Use of a variant of undefined time creates a basis for formation of task decision for allocated time due to reduction in quality of the decision.

For MR working in the uncertain environment, variants of individual concretization tasks decision distinguished by time and quality of the decision should be formed dynamically in view of the current parameters of external conditions and internal resources of the robot. With this purpose in everyone frame - action of the plan slot is added, containing function $u_S(P_l, ..., P_k)$, forming set $\{v_i^k\} = \{(T_i^k, Q_i^k)\}$ for a concrete situation (index *k* - number of variant of tasks decision). Function $u_S(P_l, ..., P_k)$ should have low computing complexity.

The received sets $\{v_i^k\}$ serve as the initial data for distribution of time $T_d^{(i)}$ allocated for concretization of the given plan of actions, between individual tasks of separate steps concretization.

Conclusion

Use of suggested method of MR actions planning in uncertain environment allows to determine dynamically time of approach of critical event on the basis of forecasting of possible consequences of the current situation with use of various environment processes laws knowledge.

Tasks of search of optimum values of action plan parameters depend on the concrete mathematical models used for the description of corresponding steps of the plan, and can be the diversified.

At the same time, MR control systems with recognition of situations and planning of actions are characterized by great volume of the processed information and high complexity of used algorithms of processing of the information and decision-making. High reliability demands are also made for them. The specified characteristics can be achieved due to use of multiprocessing computing systems, for example as an artificial neural network. In hardware realization neural network is a network from set of simple processors, each of which has small local memory and communication connections with other processors.

Prototypes of such networks for MR control systems have been already in use now for forecasting of situations in financial sphere, images recognition, speech.

Literature

1. Popov E.V., Firdman G.R. Algorithmic Basics of Intellectual Robots and Artificial Intelligence. – M.: Nauka, 1976.
2. Homogeneous Managing Structures of Adaptive Robots / Kaljaev A.V., Chernuhin Yu.V., Noskov V.N., Kaljaev I.A. under A.V.Kaljaeva and J.V.Chernuhina edition.- Moscow: Nauka, 1990
3. Stenz A., Map-Based strategies for Robot Navigation in Unknown Environments/in proc. AAAI 96 Planning with incomplete information for Robot Problems
4. Artificial Intelligence. - In 3 books. B. 2. Models and Methods: Directory / Under D.A.Pospelov edition – M.: Radio I Svyaz, 1990.
5. Zilberstein S. Using anytime algorithms in intelligent systems, AI Magazine, 1996, v. 17, N 3, p.73-83.

Authors information:

Romanenko Nadezhda - Institute of Artificial Intelligence, B.Hmelnitsky avenue, 84, Donetsk - 83050, Ukraine
e-mail: rni@iai.donetsk.ua

STABILITY OF AN OPTIMAL SCHEDULE FOR A JOB-SHOP PROBLEM WITH TWO JOBS

Yu. N. Sotskov, N. Yu. Sotskova

***Abstract**: The usual assumption that the processing times of the operations are known in advance is the strictest one in scheduling theory. This assumption essentially restricts practical aspects of deterministic scheduling theory since it is not valid for the most processes arising in practice. The paper is devoted to a stability analysis of an optimal schedule, which may help to extend the significance of scheduling theory for decision-making in the real-world applications. The term stability is generally used for the phase of an algorithm, at which an optimal solution of a problem has already been found, and additional calculations are performed in order to study how solution optimality depends on variation of the numerical input data.*

***Keywords**: Job shop problem, geometric approach, stability analysis*

Introduction

The problem under consideration is to minimize the value of the given objective function of completion times of n jobs J = {1, 2, …, n} processed on m machines M = {1, 2, …, m}. First, we assume that processing time $t_{j,k}$ of job $j \in J$ on machine $k \in M$ (i.e., processing time of operation $O_{j,k}$) is known before scheduling. Operation preemptions are not allowed. This problem is denoted as $J||\Phi$ where Φ defines objective function.

Let $C_{i,k}$ denote the completion time of the job in position i on machine $k \in M$. We assume that objective function $\Phi(C_{1,m}, C_{2,m}, \ldots, C_{n,m})$ is non-decreasing function of job completion times. Such a criterion is called regular.

For the job-shop problem $J|n=2|C_{max}$ with two jobs and makespan objective function $C_{max} = \max\{C_{1,m}, C_{2,m}, \ldots, C_{n,m}\}$, the geometric algorithm was proposed by Akers and Friedman [1] and developed by Brucker [2], Szwarc [7], Hardgrave and Nemhauser [4]. Sotskov [5] generalized the geometric algorithm for the problem $J|n=2|\Phi$ with any given regular criterion. Sotskov [6] proven that both problems $J|n=3|C_{max}$ and $J|n=3|\sum_{i=1}^{n} C_{i,m}$ are binary NP-hard. Hereafter, the criterion $\sum C_{i,k}$ means minimization of total completion time $\sum_{i=1}^{n} C_{i,k}$.

Geometric algorithm

For simplicity, we describe geometric model for the case of a flow-shop problem F|n=2|Φ, i.e., when all n jobs have the same technological route through m machines, namely, (1, 2, …, m).

Let $TM_{j,k}$ denote the sum of the processing times of job $j \in J = \{1, 2\}$ on a subset of k machines $\{1, 2, ..., k\} \subseteq M$: $TM_{j,k} = \sum_{i=1}^{k} t_{j,i}$, $1 \le k \le m$. It is assumed that $TM_{1,0} = TM_{2,0} = 0$. We introduce a coordinate system xy on the plane, and draw the rectangle H with corners (0, 0), ($TM_{1,m}$, 0), (0, $TM_{2,m}$) and ($TM_{1,m}$, $TM_{2,m}$). In the rectangle H, we draw m rectangles H_k, $k \in \{1, 2, …, m\}$, with corners ($TM_{1,k-1}$, $TM_{2,k-1}$), ($TM_{1,k}$, $TM_{2,k-1}$), ($TM_{1,k-1}$, $TM_{2,k}$), ($TM_{1,k}$, $TM_{2,k}$). We denote south-west corner ($TM_{1,k-1}$, $TM_{2,k-1}$) of the rectangle H_k as SW_k, north-west corner ($TM_{1,k-1}$, $TM_{2,k}$) as NW_k, south-east corner ($TM_{1,k}$, $TM_{2,k-1}$) as SE_k, and north-east corner ($TM_{1,k}$, $TM_{2,k}$) as NE_k. Obviously, point (0, 0) is SW_1 and point ($TM_{1,m}$, $TM_{2,m}$) is NE_m.

We use Chebyshev's metric, i.e., the length d[(x, y), (x', y')] of a segment [(x, y), (x', y')] connecting points (x, y) and (x', y') in the rectangle H is calculated as follows:

$d[(x, y), (x', y')] = \max\{|x – x'|, |y – y'|\}$.

The length $d[(x_1, y_1), (x_2, y_2), …, (x_r, y_r)]$ of a continuous polygonal line $[(x_1, y_1), (x_2, y_2), …, (x_r,y_r)]$ is equal to the sum of the lengths of its segments.

Since $\Phi(C_{1,m}, C_{2,m})$ is a non-decreasing function, the search for the optimal schedule can be restricted to set S of schedules in which at any time of the interval $[0, \max\{C_{1,m}, C_{2,m}\}]$ at least one job is processed. A schedule from set S can be suitably represented within the rectangle H on the plane xy as a trajectory (continuous polygonal line) $\tau = [SW_1, (x_1, y_1), (x_2, y_2), …, (x_r, y_r), NE_m]$ where either $x_r = TM_{1,m}$ or $y_r = TM_{2,m}$.

Let a point (x, y) belong to the trajectory τ and let d be the length of the part of trajectory τ from the point SW_1 to the point (x, y). The coordinate x (coordinate y) of point (x, y) defines the state of processing job 1 (job 2) as follows.

If $SW_u \le x \le SE_u$ and $SW_v \le y \le NW_v$, $u \in M$, $v \in M$, then job 1 (job 2) is completed on the machines 1, 2,…, u-1 (on the machines 1, 2,…, v-1) at time d. Moreover at time d, job 1 (job 2) has been processed on machine u (machine v) during $x - SW_u$ (during $y - SW_v$) time units.

Since a machine cannot process more than one job at a time and operation preemptions are not allowed, each straight segment [(x, y), (x', y')] of a trajectory τ may be either

- horizontal (when only job 1 is processed) or
- vertical (when only job 2 is processed) or
- diagonal with slope of 45^0 (when both jobs are processed simultaneously).

It is clear that a horizontal segment (vertical segment) can only pass along south boundary (west boundary) of the rectangle H_k, $k \in M$, or along north (east) boundary of the rectangle H. The diagonal segment of trajectory τ can only pass either outside rectangle H_k or through point NW_k or point SE_k.

Sotskov [5] proven that problem J|n=2|Φ of finding the optimal schedule or, in other words, of finding the optimal trajectory, can be reduced to the shortest path problem in the digraph (V, A) constructed by the following Algorithm 1. Again for simplicity, we describe this algorithm for the case of a flow-shop problem F|n=2|Φ, when all n jobs have the same technological route through m machines.

Vertex set V of the digraph (V, A) is a subset of set

$$V^0 = \{SW_1, NE_m\} \cup \{NW_k, SE_k : k \in M\} \cup \{(x_k, TM_{2,m}), (TM_{1,m}, y_k) : k \in M\}.$$

Algorithm 1

1. Set $V = \{SW_1, SE_1, NW_1, NE_m\}$ and $A = \{(SW_1, SE_1), (SW_1, NW_1)\}$.
2. Take vertex $(x, y) \in V \setminus \{NE_m\}$ with zero outdegree. If $(x, y) = SE_k$, go to step 3. If $(x, y) = NW_k$, go to step 4. If set $V \setminus \{NE_m\}$ has no vertex with zero outdegree, STOP.
3. Draw a diagonal line with slop 45^0 starting from vertex SE_k until either east boundary $[(TM_{1,m}, 0), NE_m]$ of the rectangle H is reached in some vertex $(TM_{1,m}, y_k)$ or open south boundary (SW_h, SE_h) of the rectangle H_h, $k+1 \le h \le m$, is reached. In the former case, set $V: =V \cup \{(TM_{1,m}, y_k)\}$ and $A: =A \cup \{(SE_k, (TM_{1,m}, y_k)), ((TM_{1,m}, y_k), NE_m)\}$. In the latter case, set $V: =V \cup \{SE_h, NW_h\}$ and $A : =A \cup \{(SE_k, SE_h), (SE_k, NW_h)\}$. Go to step 2.
4. Draw a diagonal line with slope 45^0 starting from vertex NW_k until either north boundary $[(0, TM_{2,m}), NE_m]$ of the rectangle H is reached in some vertex $(x_k, TM_{2,m})$ or open west boundary (SW_h, NW_h) of the rectangle

H_h, $k+1 \le h \le m$, is reached. In the former case, set V: = V∪{(x_k, $TM_{2,m}$)} and A: = A∪{(NW_k, (x_k, $TM_{2,m}$)), ((x_k, $TM_{2,m}$), NE_m)}. In the latter case, set V: = V∪{SE_h, NW_h} and A: = A∪{(NW_k, SE_h), (NW_k, NW_h)}. Go to step 2.

In order to find the optimal path (i.e., optimal schedule) for the problem J|n=2|Φ we can use the following Algorithm 2, where the length of arc ((x, y), (x', y')) ∈ A is assumed to be equal to the length of the polygonal line constructed by Algorithm 1 with origin in the point (x, y) and with end in the point (x', y').

Algorithm 2

1. Construct the digraph (V, A) using Algorithm 1 and find all border vertices in the digraph (V, A), i.e., the vertices (x, y) either of the form (x_k, $TM_{2,m}$) or of the form ($TM_{1,m}$, y_k).
2. Construct the set of trajectories corresponding to the shortest paths in the digraph (V, A) from the vertex SW_1 to each of the border vertices.
3. Find an optimal trajectory (optimal path in (V, A)) in the set constructed at step 2 that represents a schedule with minimal value of the objective function Φ.

It was proven that Algorithm 2 takes O(m log m) time for problem F|n=2|Φ and its generalization for problem J|n=2|Φ takes $O(m^2 \log m)$ time (see Sotskov [5, 6]).

Example

Next, we demonstrate the geometric algorithm for the problem Fk|n=2|Φ using five machines and processing times given in Table 1. We call this example as Example 1.

Machine m	m=1	m=2	m=3	m=4	M=5
$t_{1,m}$	2	20	8	4	12
$t_{2,m}$	16	4	6	2	4

Table 1. Processing times of two jobs

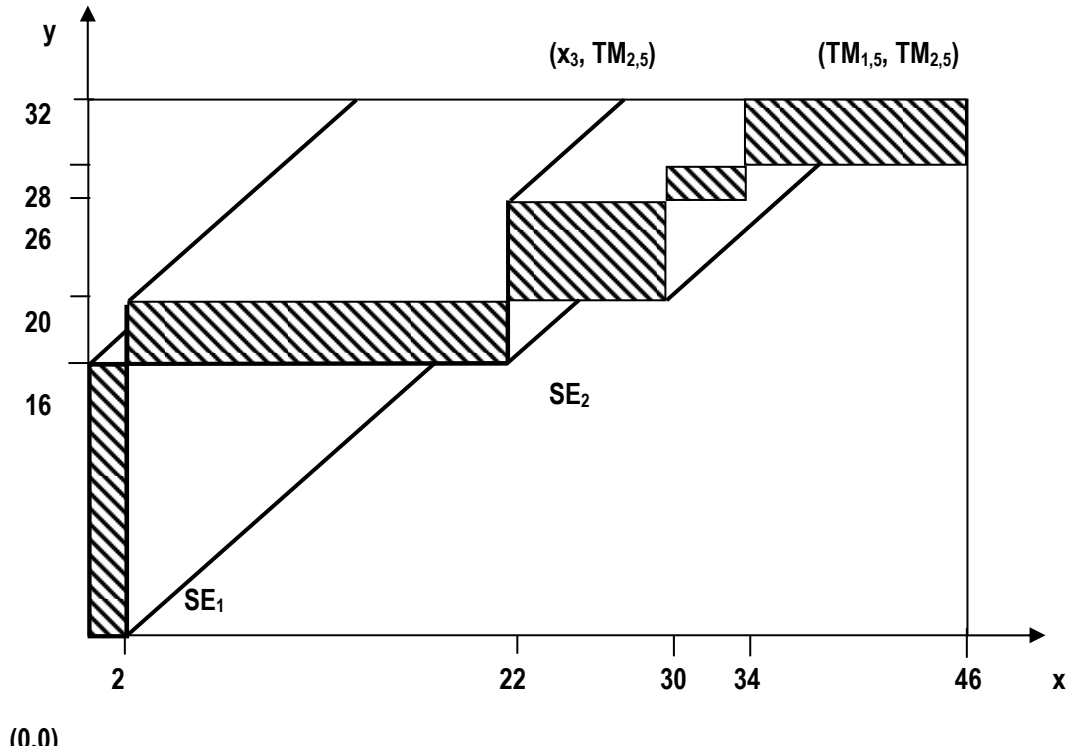


Figure 1. Trajectories representing active schedules for Example 1

In Fig. 1, the rectangles R_m for Example 1 are shaded.

For Example 1, we see that the shortest path ((0, 0), SE_1, SE_2, NW_3, (x_3, $TM_{2,5}$)) from vertex (0, 0) to the border vertex (x_3, $TM_{2,5}$) in the digraph (V, A) constructed by Algorithm 1 specifies trajectory [(0, 0), (2, 0), (18, 16), (22, 16), (22, 26), (28, 32)] in the rectangle R. Using Algorithm 2 we indicate that the schedule represented by this trajectory is optimal for the problem F5|n=2|$\Sigma C_{i,5}$ with total completion time criterion.

Stability analysis

In what follows, we consider stability of an optimal schedule with respect to possible variations of the given vector t = ($t_{1,1}$, $t_{1,2}$, … , $t_{1,m}$, $t_{2,1}$, $t_{2,2}$, …, $t_{2,m}$) of operation processing times.

Let (V_t, A_t) denote the digraph (V, A) constructed by Algorithm 1 for the problem F|n=2|Φ with vector t of operation processing times. Let $\mathbf{P_t}$ be set of all shortest paths from vertex SW_1 to the border vertices in the digraph (V_t, A_t). As follows from Algorithm 1, the same path may belong to sets $\mathbf{P}_t$ constructed for different vectors t of operation processing times (since for any vector t we have $V_t \subseteq V^0$). Notation $s_u(t)$ is used for a schedule defined by path $\tau_u \in \mathbf{P}_t$. The objective function value calculated for schedule $s_u(t)$ is denoted as $\Phi(s_u(t))$.

A schedule is called active if none of the operations can start earlier than in this schedule, provided that the remaining operations could start no later. It is known (see Giffler and Thompson [3]) that a set of active schedules is dominant (i.e., it contains at least one optimal schedule) for any regular criterion. The following claim may be proven by induction with respect to number of machines m.

Theorem 1: *If $\boldsymbol{P_t}$ is set of all shortest paths from vertex SW_1 to the border vertices in the digraph (V_t, A_t), then set $\boldsymbol{P_t}$ defines all active schedules for the problem F|n=2|Φ with operation processing times defined by vector t.*

Let $\mathbf{R}^{2m}$ be space of non-negative 2m-dimensional real vectors t = ($t_{1,1}$, $t_{1,2}$, … , $t_{1,m}$, $t_{2,1}$, $t_{2,2}$, …, $t_{2,m}$) with Chebyshev's metric

$d(t, t^0) = \max\{|t_{i,j} - t^0_{i,j}| : i \in \{1, 2\}, j \in \{1, 2, \ldots, m\}\}$

where $t^0 = (t^0_{1,1}, t^0_{1,2}, \ldots, t^0_{1,m}, t^0_{2,1}, t^0_{2,2}, \ldots, t^0_{2,m}) \in \mathbf{R}^{2m}$. Let path $\tau_u \in \mathbf{P}_t$ be optimal for the problem F|n=2|Φ with operation processing times defined by vector t. If for any small positive real number $\varepsilon > 0$ there exists vector $t^0 \in \mathbf{R}^{2m}$ such that $d(t, t^0) = \varepsilon$ and path τ_u is not optimal for the problem F|n=2|Φ with operation processing times defined by vector t^0, then optimality of path τ_u is *not stable*. Otherwise, optimality of path τ_u is *stable*.

Let $\delta(\tau_u)$ denote the set of all operations $O_{j,k}$, $j \in \{1, 2\}$, which are processed by machine $k \in M$ in such a way that at the same time job $i = 3 - j$ waits since operation $O_{i,k}$ (which is ready to be processed) needs the same machine k. Obviously, if $O_{1,k} \in \delta(\tau_u)$ (respectively, $O_{2,k} \in \delta(\tau_u)$), then trajectory defined by path τ_u includes a horizontal segment [(x, y), SE_k] (vertical segment [(x, y), NW_k]).

Theorem 2: *Let path $\tau_u \in \boldsymbol{P_t}$ be optimal for the problem F|n=2|Φ where Φ is continuous increasing function of job completion times. Optimality of path τ_u is stable if and only if set $\boldsymbol{P_t}$ does not contain another optimal path for the problem F|n=2|Φ with operation processing times defined by vector t.*

Proof: *Sufficiency.* Since set of active schedules is dominant for any regular criterion, it is sufficient to compare schedule $s_u(t)$ with other active schedules. So due to Theorem 1, we have to compare path τ_u with other paths $\tau_v \in \mathbf{P_t}$, $\tau_v \neq \tau_u$. Since path τ_u is unique optimal path, we get inequality $\Phi(s_v(t)) - \Phi(s_u(t)) > 0$. Since Φ is increasing function, in order to overcome the difference $\Phi(s_v(t)) - \Phi(s_u(t))$ for the new vector t^0 of operation processing times, we have to increase the processing times for operations from the set $\delta(\tau_u)$ or (and) to decrease the processing times for operations from the set $\delta(\tau_v)$. Since Φ is continuous function, we can reach equality $\Phi(s_v(t^0)) - \Phi(s_u(t^0)) = 0$ only if $d(t, t^0) > 0$. Thus, optimality of path τ_u is stable.

Necessity. Let equality $\Phi(s_w(t)) = \Phi(s_u(t))$ hold. Since optimal paths τ_w and τ_u are different, either set $\delta(\tau_w) \setminus \delta(\tau_u)$ or set $\delta(\tau_u) \setminus \delta(\tau_w)$ is not empty. In the former case (we call it as case (a)), there exists at least one operation $O_{j,k} \in \delta(\tau_w) \setminus \delta(\tau_u)$ such that trajectory defined by path τ_w includes some segment of a boundary of rectangle H_k while trajectory defined by path τ_u does not include a segment of this boundary. In the latter case (we call it as case (b)), there exists at least one operation $O_{i,r} \in \delta(\tau_u) \setminus \delta(\tau_w)$ such that trajectory defined by path τ_u includes some

segment of a boundary of rectangle H_r while trajectory defined by path τ_w does not include a segment of this boundary. Note that Φ is increasing function of job completion times.

Therefore, if in the case (a) we subtract any small positive value $\varepsilon > 0$ from the value $t_{j,k}$ with remaining the same all other components of the vector t, then we get such a vector t^0 of operation processing times that inequality $\Phi(\tau_w(t^0)) < \Phi(\tau_v(t^0))$ holds. On the other hand, if in the case (b) we add any small positive value $\varepsilon > 0$ to the value $t_{i,r}$ with remaining the same all other components of the vector t, then we get such a vector t^* of operation processing times that inequality $\Phi(\tau_w(t^*)) < \Phi(\tau_v(t^*))$ holds. Since value ε can be as small as desired, we conclude that optimality of path τ_u is not stable in both cases (a) and (b).

■

Returning to the Example 1, we see that the shortest path ((0, 0), SE_1, SE_2, NW_3, (x_3, $TM_{2,5}$)) from vertex (0, 0) to the border vertex (x_3, $TM_{2,5}$) in the digraph (V, A) is stable since set $\mathbf{P_t}$ does not contain another optimal path for the problem F|n=2|Φ with operation processing times defined by vector t given in Table 1.

Conclusion

Both Theorems 1 and 2 will be correct if flow shop problem F|n=2|Φ will be replaced by job shop problem J|n=2|Φ.

To test whether optimality of the path $\tau_u \in \mathbf{P}_t$ is stable takes O(m log m) time for problem F|n=2|Φ and O(m^2 log m) time for problem J|n=2|Φ. Indeed, we can use Algorithm 2 for the vector t of the operation processing times and construct optimal paths with different border vertices. Number of the optimal paths which have to be tested due to Theorem 2 is restricted by the number of border vertices asymptotically restricted by O(m) for problem F|n=2|Φ and by O(m^2) for problem J|n=2|Φ.

It is easy to convince that for the above sufficiency proof of Theorem 2 we can replace increasing function Φ by non-decreasing function Φ. It should be noted that the most objective functions considered in classical scheduling theory are continuous non-decreasing functions of job completion times, e.g., makespan C_{max}, total completion time $\sum_{i=1}^{n} C_{i,m}$, maximal lateness $L_{max} = \max\{C_{i,m} - D_i : i \in J\}$ and total tardiness $\sum_{i=1}^{n} T_{i,m} = \sum_{i=1}^{n} \max\{0, C_{i,m} - D_i : i \in J\}$ where D_i denotes the given due date for a job i. However, function $\Phi = \sum_{i=1}^{n} sign(\max\{0, C_{i,m} - D_i\})$ equaled to the number of late jobs is not continuous, and so sufficiency of Theorem 2 may be violated in the break points of such a function Φ.

The research of the first author was supported by INTAS (Project 00-217).

Bibliography

[1] S.B. Akers and J. Friedman, A non-numerical approach to production scheduling problems, *Operations Research* 3, 1955, 429 - 442.

[2] P. Brucker, An efficient algorithm for the job-shop problem with two jobs, *Computing* 40, 1988, 353 - 359.

[3] B. Giffler and G.L. Thompson, Algorithms for solving production-scheduling problems, *Operations Research* 8, N 4, 1960, 487 - 503.

[4] W.W. Hardgrave and G. Nemhauser, A geometric model and graphical algorithm for a sequencing problem, *Operations Research* 11, N 6, 1963, 889 - 900.

[5] Y.N. Sotskov, Optimal scheduling two jobs with regular criterion, In: *Design Processes Automating,* Institute of Engineering Cybernetics, Minsk, 1985, 86 - 95 (in Russian).

[6] Y.N. Sotskov, The complexity of shop-scheduling problems with two or three jobs, *European Journal of Operational Research* 53, 1991, 326 - 336.

[7] W. Szwarc, Solution of the Akers-Friedman scheduling problem, *Operations Research* 8, 1960, 782 - 788.

Author information

Yuri N. Sotskov - United Institute of Informatics Problems of National Academy of Sciences of Belarus, Surganov Str. 6, 220012 Minsk, Belarus; e-mail: sotskov@newman.bas-net.by

Nadezhda Yu. Sotskova - Hochschule Magdeburg-Stendal, Fachbereich Wasserwirtschaft, PSF 3680, D-39011 Magdeburg, Germany; OR Soft Jänicke GmbH, Geusaer Str. 104, FH, D-06217 Merseburg, Germany; e-mail: nadezhda.sotskova@orsoft.de

КОМПЬЮТЕРНАЯ ПОДДЕРЖКА ПРИ СОСТАВЛЕНИИ ПРОИЗВОДСТВЕННЫХ РАСПИСАНИЙ

Н. Ю. Сотскова, В. Енике, В. Темельт

***Аннотация**: При планировании в многостадийных системах обслуживания производства необходимо распределить (а затем упорядочить) заданные требования между заданными приборами в соответствии с технологическими маршрутами (рецептурами) и с учетом принятого критерия оптимальности. В системе SCHEDULE++ можно реализовать такие процессы автоматически или в диалоговом режиме на основе точных или эвристических алгоритмов. При этом из плановых заказов строятся технологические заказы. Если некоторые из выбранных приборов (ресурсов) не освобождаются к необходимому моменту времени, то возникший конфликт изображается на диаграмме Ганта. Для разрешения такого конфликта планировщику необходимо изменить те или иные параметры технологического заказа (например, размер партии или дату поставки конечной продукции) и согласовать необходимые изменения с клиентами. Поиск множества допустимых (в конкретном интервале планирования) технологических заказов включается в интерактивном режиме в процесс составления расписания. В статье описываются основные особенности решения задачи планирования производства с использованием автоматизированной системы SCHEDULE++.*

***Ключевые слова**: Календарное планирование производства, эвристические алгоритмы.*

Введение

Планирование производства является одной из основных функций, которые постоянно реализуются на предприятиях. Цель планирования производства состоит в обеспечении необходимых объемов производимых продуктов своевременно и с наименьшими затратами. Для оптимального планирования необходимо максимизировать использование производственных мощностей, сократить количество переналадок приборов, минимизировать складские запасы различных видов сырья, полуфабрикатов, промежуточных продуктов и конечной продукции. Одна из задач планирования производства состоит в том, чтобы ответить на вопрос, может ли производиться и каким образом множество указанных продуктов к заданному сроку или в заданном интервале времени. Можно ли выполнить указанное множество заказов при использовании одного или несколько ресурсов предприятия в заданные сроки?

Реальное планирование производства, как правило, требует от планировщика компромиссов, которые связаны с разрешением возникающих конфликтных ситуаций. Конфликт может быть разрешен оптимальным образом, если лицу, принимающему решение, предоставлена по возможности в наглядном виде вся информация о сложившейся ситуации на производстве и если достоверно известны приемлемые производственно-экономические параметры. Система SCHEDULE++ предоставляет такую информацию пользователю и позволяет эффективно организовать поиск путей разрешения конфликтных ситуаций.

Планирование в многостадийных системах обслуживания

Планирование современного производства нельзя рассматривать как изолированный процесс. Планирование должно включаться в контур регулирования всей работы предприятия и должно охватывать материально-техническое снабжение, координацию цепей поставок, техническое обслуживание оборудования, потребительское планирование (планирование потребностей в материалах), стратегическое планирование (среднесрочное и долгосрочное планирование), оперативное планирование (краткосрочное), календарное планирование (составление расписания), логистику, наблюдение за клиентами и т.д. Все указанные функции реализуются сотрудниками предприятия, которые используют для своей деятельности вспомогательные компьютерные средства с запрограммированными специфическими алгоритмами планирования производства, предназначенными для моделирования, имитации, оптимизации и т.д. В качестве одного из таких вспомогательных средств может эффективно использоваться система SCHEDULE++. Типичный сценарий планирования на основе системы SCHEDULE++ изображен на рис. 1.

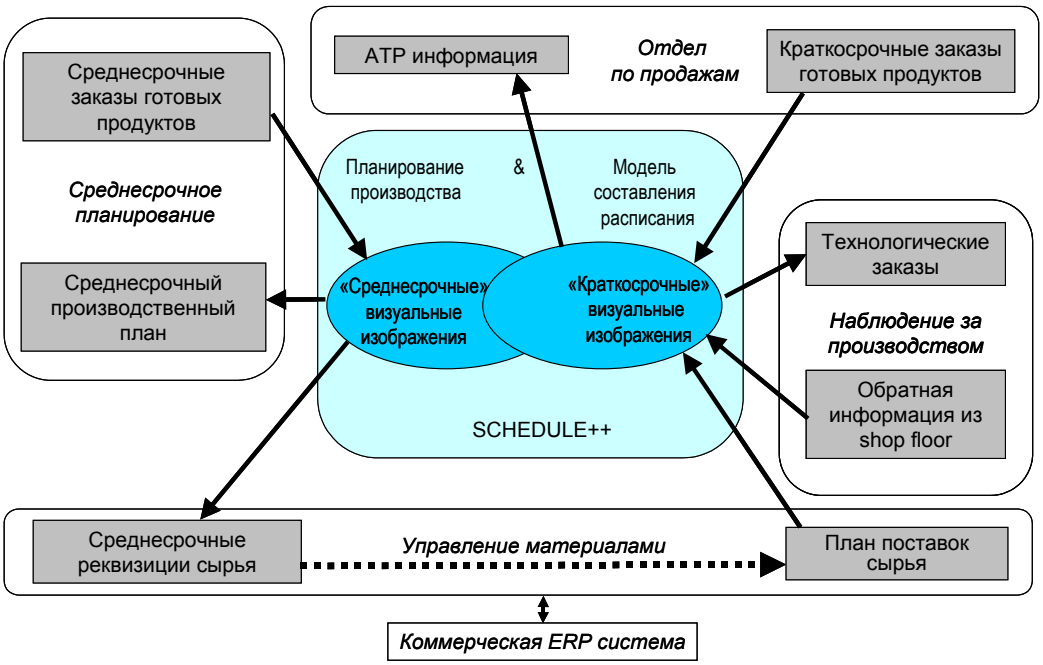


Рис. 1. Общий сценарий планирования

Стратегическое планирование. С одной стороны, на предприятии необходимо проводить долгосрочное и среднесрочное планирование, т.е. стратегическое планирование на длительный период времени (квартал, год или несколько лет). При таком планировании невозможно получить точную информацию о поступающих заказах, поэтому стратегическое планирование, как правило, проводится на основе информации о реальном производстве за прошлые месяцы или годы. При этом планировщику и менеджеру необходимо хотя бы приблизительно знать, какими же производственными мощностями они могут располагать в будущем, т.е. каковы возможности предприятия для выполнения намеченных заказов. Клиенты, как правило, первоначально высказывают предполагаемые размеры своих заказов. После получения такого «заказа» от клиента планировщик строит плановые заказы, на основе которых должна быть получена ATP (Available-To-Promise) информация о возможности выполнения данного заказа клиента с приблизительными временными оценками (на такой-то неделе заказ может быть выполнен). На этом этапе проводится также планирование потребностей в материалах, т.е. осуществляется заказ необходимого сырья и материалов.

По ряду причин решение конкретной задачи планирования может оказаться невозможным (например, из-за недоступности ресурсов или недостаточной мощности ресурсов, из-за слишком больших размеров заказанной клиентом партии и т.д.). В такой ситуации должна быть осуществлена некоторая модификация построенного плана. В частности, необходимо найти возможность сократить какой-либо другой производственный процесс, чтобы освободить производственные мощности (ресурсы) для выполнения

более срочного заказа. Можно также объединить повторные технологические заказы, сократить количество переналадок приборов (машин). При этом должны быть учтены возможные побочные эффекты, связанные с модификацией плана производства. Планировщик и менеджер должны довести заказ клиента до реально допустимого, согласовав временные и количественные условия выполнения заказа с клиентом. Следует отметить, что комбинация непрерывных и дискретных шагов производства нуждается в некотором особом способе принятия решений для того, чтобы обеспечить наилучшую производительность предприятия в целом. По этой причине при реализации алгоритмов управления работой на производстве важно знать обо всех существенных числовых параметрах.

Оперативное планирование. После заключения с клиентами договоров о производстве строится подробный краткосрочный оперативный план (с периодом до 10 дней). К этому моменту времени планировщики уже должны располагать достаточно полной информацией о требованиях клиентов, о состоянии производственных мощностей и о технологических рецептурах производства. На основе такой достоверной информации составляются технологические заказы. Процесс обслуживания технологических заказов в системе производства включает, как правило, несколько операций. На последнем этапе планирования производства (календарное планирование) необходимо распределить операции по приборам (ресурсам), т.е. составить расписание. При этом решаются комбинаторные задачи из области теории расписаний (в немецком языке используется термин «Maschinenbelegungsproblem», в английском – «scheduling problem»). В качестве расписания принято рассматривать совокупность кусочно-постоянных непрерывных слева функций, которые указывают какие требования обслуживаются какими приборами в соответствующие интервалы времени. В системе SCHEDULE++ расписание можно представить в виде диаграммы Ганта (Gantt Chart). Пример такого представления расписания изображен на рис. 2. Поскольку вся необходимая информация, как правило, известна к моменту составления расписания, то при краткосрочном планировании необходимость принятия оперативных решений возникает только том в случае, когда какой-то прибор (ресурс) выйдет из строя, сломается транспортное средство и т.п.

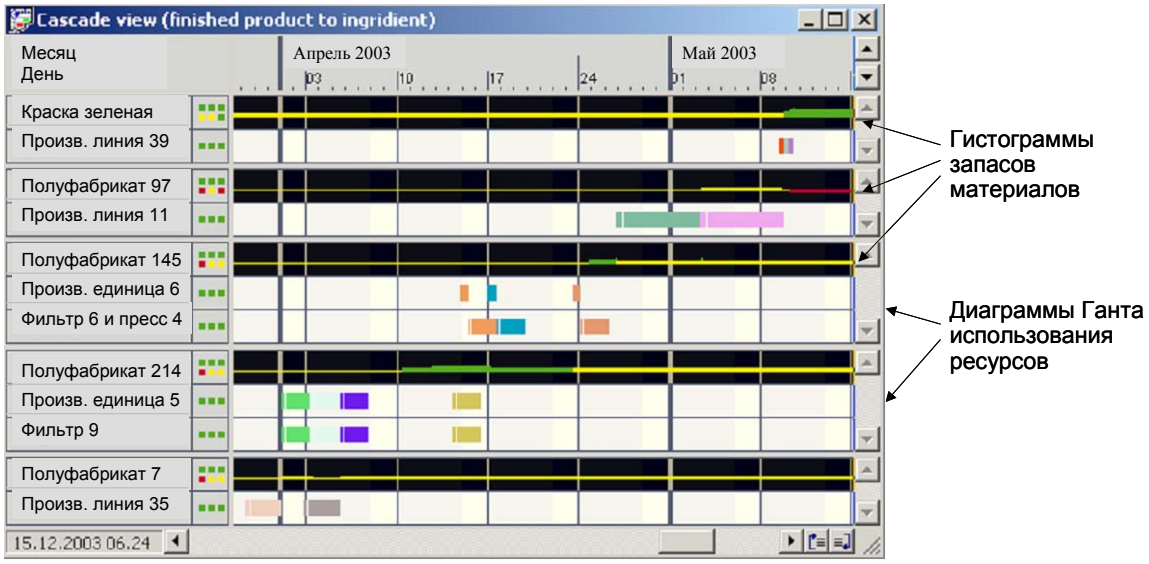


Рис. 2. Графическое представление ситуации планирования

Основываясь на имеющейся информации, с помощью системы SCHEDULE++ можно проводить как краткосрочное, так и долгосрочное планирование (рис. 1). Опишем более подробно модели данных, которыми обычно оперируют на предприятии.

Модель данных

Основные данные. Такие объекты данных как материалы, ресурсы и технологические рецептуры являются основными данными. Все такие объекты данных получают идентификационный номер и сопровождаются структурированными сведениями. В качестве ресурсов могут рассматриваться производственные линии, станки, резервуары, танкеры, пресса, фильтры, железнодорожные пути, суда, склады, рабочая сила и т.п. Проще всего представить материалы и ресурсы в виде Excel таблиц. Более

сложными являются технологические рецептуры, которые представляют собой описание, какие материалы, в каких количествах и когда назначаются в производство, какие ресурсы как долго и для обработки каких материалов пригодны. Каждому полуфабрикату и готовому продукту соответствует технологическая рецептура. Рецептура описывает весь производственный процесс изготовления продукта и включает в себя:

- списки необходимых материалов (сырья, полуфабрикатов, упаковочного материала);
- списки ресурсов, которые должны быть задействованы (иногда одна и та же операция может быть выполнена несколькими идентичными или различными приборами);
- длительности выполнения операций;
- промежутки времени между операциями (что очень важно, например, в химическом производстве, т.к. полуфабрикат зачастую нельзя поместить на склад из-за продолжающихся в нем химических реакций);
- длительности переналадок приборов (матрицы переналадочных интервалов, где каждой паре материалов соответствует время необходимой переналадки, которое может варьироваться в зависимости от последовательности материалов от нескольких минут до нескольких недель).

Динамические данные. К динамическим данным относятся:

- балансовые потоки материалов (т.е. заказы клиентов, включающие имя клиента, материал, количество и срок поставки; заказы на поставку, характеризующиеся поставщиком, материалом, количеством и сроком поставки);
- технологические заказы (описание процессов обработки материалов с использованием ресурсов, строго следующее технологическим рецептурам, с конкретными временными и пространственными характеристиками);
- заводские запасы материалов (количество материалов к определенной дате, которое рассчитывается на основе последней инвентаризации, закупок, сбыта, производства или расхода на основе технологических заказов между датой инвентаризации и текущей датой).

Строгие и нестрогие данные. При описании модели планирования «строгая» информация о потоках материалов и использовании ресурсов характеризуется сведениями о дате, сроке, количестве и способе использования. Эти данные описывают в каждый момент времени конкретное состояние модели планирования как текущее, так и будущее. Дополнительно имеется и ряд «нестрогих сведений», которые также поступают в модель планирования и выражают предположение по поводу будущей ситуации планирования (например, заказы клиентов, плановые заказы и независимые плановые требования).

Плановые заказы по сути представляют собой метки-заполнители для возможных технологических заказов, которые еще не проверены по всем планово допустимым параметрам (т.е. находятся на этапе среднесрочного или долгосрочного планирования). Такие заказы используются в иерархии планирования как указания планировщику о том, что именно он должен вкладывать в технологический заказ. Плановыми заказами предварительно оцениваются потоки материалов с конкретными временными и количественными характеристиками. По существу, в плановом заказе выражено только предположение о сбыте продукта (в такой-то момент может возникнуть необходимость в таком-то количестве продукта).

Ограничения. При построении расписания должны соблюдаться все условия и ограничения, вытекающие из постановки рассматриваемой задачи, т.е. расписание должно быть допустимым. Во-первых, необходимо учитывать различные ограничения на использование ресурсов (например, рабочий режим, квалификации рабочих, ограниченный объем резервуара, ограниченное потребление воды, пара, электроэнергии, ограничение на количество отходов и т.д.) и на использование материалов (огнеопасные или взрывчатые вещества, химические полуфабрикаты, отходы разного типа). Во-вторых, определяются связи между отдельными видами основных и динамических данных (например, матрицей переналадок приборов, сетью заказов). В отношении методов расчета для динамических данных могут предусматриваться те или иные особые условия (например, соблюдение «строгих» ограничений для определенных ресурсов в течение определенных промежутков времени, или наоборот только уведомление планировщика о нарушении некоторых «нестрогих» ограничений).

Пример многостадийной системы

Как типичный пример многостадийной системы обслуживания рассмотрим многопрофильный химический завод, на котором производятся партиями или полунепрерывно различные продукты (например, особо

чистые химикаты, краски). На завод поступает сырье, которое соединяется в реакторах, где проводятся химические реакции, в результате которых возникает некоторый полуфабрикат. Полученный продукт закрепляется в щелочи, фильтрируется, прессуется и затем упаковывается в различных формах. Резервуар щелочи должен сохраняться и при необходимости использоваться повторно. Иногда продукт может производиться различными способами в зависимости от используемых устройств, но, разумеется, строго в соответствии с технологической рецептурой.

Многоступенчатый производственный процесс характеризуется описанием потоков материалов и использованием ресурсов. Рисунок 2 демонстрирует план производства зеленой краски. В левой части рис. 2 указаны названия материалов (полуфабрикатов и готового продукта) и задействованных ресурсов (производственных единиц, линий, фильтров, прессов). В правой части рис. 2 указаны гистограммы запасов материалов и графики Ганта использования ресурсов, иллюстрирующие начальные и конечные моменты выполнения операций.

Интегрированная система SCHEDULE++

Опишем кратко основную внутреннюю структуру системы SCHEDULE++, схематично представленную на рис. 3.

Рис. 3. Основа SCHEDULE++

Алгоритмы SCHEDULE++. Вызов алгоритмов планирования осуществляется по умолчанию в пределах работы системы SCHEDULE++. Для того, чтобы управлять основными элементами процесса планирования, алгоритмы используют различные параметры, такие как времена загрузки, мощности ресурсов, готовность материалов и т.д. Ограничения для алгоритма планирования включают заданные временные горизонты, которые диктуются заказчиками готовой продукции. Таким образом, система SCHEDULE++ позволяет организовать очень детальное планирование в ближайшем будущем и в то же время позволяет планировать предварительное размещение заказов в далеком будущем. Фактические параметры некоторого технологического заказа могут быть выбраны в рамках системы SCHEDULE++ так, чтобы наилучшим образом соответствовать ожидаемым изменениям производственной ситуации. Могут быть использованы формулы, определяющие эффективную продолжительность шага технологического заказа относительно определенного ресурса, изменений производительности прибора, поставок сырья и материалов к следующим этапам процесса обработки (к следующим шагам) или относительно каких-либо других факторов. Сложные задачи составления расписаний, как правило, сопровождаются целым набором ограничительных параметров. При необходимости, специальные требования и процедуры программируются дополнительно. Таким образом, в системе SCHEDULE++ предусматривается любая разумная комбинация основных задач планирования наряду с генерацией визуальных представлений данных.

Составление расписания осложняется из-за того, что в большинстве случаев несколько заказов должны быть выполнены одновременно с использованием одних и тех же свободных приборов. Составление расписания еще более усложняется, когда в рецептурах планирования учитывается буфер диспозиции (переменные или константные величины временного интервала между процессами) и если имеются другие ограничения математической модели.

При составлении оптимальных расписаний возникают комбинаторные задачи оптимизации, для которых уже для задач средней размерности не существует алгоритмов, которые дают на PC (personal computer)

«оптимальный» результат за приемлемое время. Более того, построение не только оптимального расписания, но и просто допустимого расписания является в общем случае NP-трудной задачей. Поэтому при решении задач большой размерности приходится довольствоваться эвристическими алгоритмами, которые строят допустимые решения. Как показывает практика, использование таких алгоритмов позволяет находить довольно эффективные решения, близкие к оптимуму.

Система SCHEDULE++ основывается, в основном, на алгоритмах обратного шага (backtracking algorithms). Последовательное распределение требований по приборам в соответствии с технологическими рецептурами идет до тех пор, пока не встречается конфликт. При возникновении конфликта алгоритм делает обратный шаг (или несколько шагов назад до «развилки») и выбирается альтернативная возможность последовательной «загрузки» имеющихся ресурсов заданными требованиями. Так продолжается до построения допустимого решения. При этом вышеупомянутые ограничительные условия учитываются в системе SCHEDULE++ автоматически. Задача составления расписания может быть упрощена, если упростить модель планирования, например, за счет объединения многостадийных технологических заказов в одностадийные при рассмотрении только ресурса «узкого места» (bottleneck resource). Однако в таком случае обычно приходится искать компромисс между издержками вычислений и точностью полученных результатов.

Визуализация данных. Основные функциональные возможности системы SCHEDULE++ состоят в визуальном представлении данных производства (для одного или нескольких заводов). Это позволяет представить в сжатом виде обзор всей ситуации производства или цепи поставок и дает возможность опытным планировщикам и координаторам поставок принять необходимые решения достаточно быстро и по возможности эффективно. В этом заключено существенное отличие системы SCHEDULE++ от многих известных ERP (Enterprise Resource Planning) систем, таких как система SAP R/3 [1], которые обычно работают с отдельными модулями и этапами планирования, что затрудняет получение общей картины производства в целом.

Типичное представление информации о построенном расписании представлено на рис. 2. Подобные возможности визуальных представлений полученных решений позволяют выделить наиболее существенные «пробелы» (конфликты использования ресурсов, длительные простои из-за переналадок приборов или недостающее количество сырья, материалов, готовой продукции) и используются для улучшения построенного расписания. Отметим, что часто бывает трудно формализовать сложные идеи и правила разрешения возникающих конфликтов и запрограммировать такие правила для реализации на PC. Поэтому процесс устранения недостатков построенного плана часто предоставляется опытному планировщику. При этом рабочее место специалиста не замещается компьютером, что очень важно при возрастающей индустриализации и безработице, например, в Германии. Система SCHEDULE++ используется, в основном, как вспомогательное средство планирования. Основываясь на подробных графических представлениях производственной ситуации, планировщик может создавать и модернизировать объекты данных (технологические заказы, заказы на поставку, закупки реквизитов и т.п.). Это может быть сделано в оперативном режиме или в режиме имитации. Оперативный режим означает, что все изменения будут немедленно записаны назад в систему ERP. В режиме имитации можно производить несколько манипуляций данными без какого-либо взаимодействия с основной системой ERP. Все это позволяет вычислять и оценивать полный сценарий производства на предприятии. Если найдено удовлетворительное решение, то все изменения можно записать назад в систему ERP по соответствующему запросу пользователя.

Связи основных и динамичных данных. Информация об использовании какого-то ресурса в прошлом, в настоящем и (или) в будущем документируется в списке использования ресурса, в котором фиксируются все процессы по технологическим заказам, загружающим данный ресурс. Списки использования ресурсов можно представить в системе SCHEDULE++ в виде диаграмм Ганта.

Список потребностей (запасов) материала SRLM (Stock/Requirements List for a Material) позволяют отслеживать динамику доступности материалов с учетом времени. В списке SR фиксируются все внешние и внутренние заготовки (закупки, производство) и все внешние и внутренние расходы (использование, сбыт, потери) по отношению к этому материалу, в соответствии со спецификациями закупочных заказов, технологических заказов и датами поставок клиентам. На основании плановых заказов в список заносятся также запланированные поступления материалов и резервирование. Списки SR можно представить в системе SCHEDULE++ как гистограммы (рис. 2). В ходе построения гистограмм запасов алгоритмы обращаются как к основным данным, так и к динамическим данным.

MRCP. Одно из преимуществ системы SCHEDULE++ состоит в способности одновременного планирования материалов и мощности MRCP (Material Requirement and Capacity Planning). MRCP означает, например, что если какой-либо производственный заказ будет перемещен в графическую диаграмму Ганта, то автоматически будут задействованы все ресурсы, необходимые для производства этого заказа, будет также рассчитано количество вовлеченных в этот процесс материалов. Все эти изменения будут немедленно показаны на гистограммах, подобных рис. 2. Кроме того, в системе SCHEDULE++ поддерживается планирование большого количества заказов с различными стратегиями.

Одновременная иллюстрация загруженности ресурсов и наличия материалов предоставляет пользователю существенные преимущества. Можно, например, отслеживать дефициты материалов и конфликты ресурсов. Дополнительно можно просматривать составы различных складских помещений. Разумеется, при этом можно рассматривать и материалы, для которых не ведутся никакие списки потребностей и не ведутся инвентарные списки (например, промежуточные материалы, средства для очистки, катализаторы).

Динамические деревья потока материалов. Непрерывный учет материалов основывается на информации о временных длительностях процессов закупок, поставок и производства. При планировании кампаний (production campaign), в которых объединены по некоторым критериям различные технологические заказы, требуется анализировать проблему пересечения независимых технологических заказов по нескольким этапам полунепрерывных процессов потоков материалов. Основная идея решения задачи состоит в том, чтобы использовать списки материальных запасов и материальных требований MM-IM (Material Management and Inventory Management) в существующей системе PP (Production Planning). На основе такого MM-IM списка в рамках системы SCHEDULE++ был разработан алгоритм, строящий динамическое дерево потока материалов, который может быть использован в ходе составления расписания и координации поставок сырья и материалов.

Алгоритм был разработан для двух возможных «направлений» движения по такому дереву: движение «назад» (consumption pegging) и движение «вперед» (production pegging). В случае движения «назад» алгоритм ищет все последующие операции с выбранным материалом (использование материала как полуфабрикат при дальнейшем производстве различных готовых продуктов, размещение на складах, сбыт в цепи поставок и т.д.). В случае движения «вперед» алгоритм отслеживает полное производство данного материала или закупку в цепи поставок. Динамическое дерево потока материала можно начинать строить с заказа клиента, с заказа на поставку или с технологического заказа в зависимости от выбранного «направления» движения.

Возможно также динамическое представление материального потока на диаграммах Ганта и гистограммах наличия материалов (cascade view): «назад» (ingridient to finished products) и «вперед» (finished product to ingridients). Эти диаграммы визуализируют MM-IM списки и использование ресурсов, вовлеченных в процесс производства некоторого готового изделия. Можно также объединять информацию, полученную от многочисленных клиентов и систем PP.

Интегрированная система. В конечном счете, модель планирования описывает настоящие и будущие потоки материалов и использование ресурсов. Задача всех участвующих в процессе производства и планирования сотрудников (из отделов планирования, производства, покупки, продаж) состоит в обеспечении и поддержке актуальной информации относительно потоков материалов и технологических заказов на основании операций перепланирования, информации с цехов (уровня shop floor). Если каждый сотрудник занимается своим делом, то можно довольно быстро определить, какое воздействие имеет решение одного из участников на всю ситуацию в целом. Этим подчеркивается необходимость функционирования интегрированной системы поставок, планирования и производства, т.е. необходимость интегрировать планирование в процессы заводского хозяйства. Иными словами, самостоятельные системы планирования следует объединять с другими системами управления предприятием.

Справка о SCHEDULE++

Информационная система планирования SCHEDULE++ является высоко специализированной надстройкой к системе SAP R/3 (и другим ERP системам). Этот программный продукт работает с копией основных и динамические данных SAP R/3 без использования каких-либо собственных форматов данных. Преимущество системы SCHEDULE++ состоит в расширении функциональных возможностей логистических модулей PP, PP-PI (Production Planning in Process Industry), PM (Plant Maintainance), MM-IM,

SD (Sales and Distribution) системы ERP. В виде стандарта предлагается использовать широкий диапазон визуального представления и функциональных возможностей системы SCHEDULE++, требуемых для решения различных задач планирования. Система SCHEDULE++ может быть инсталлирована на обычном персональном компьютере с операционными системами MS-Windows NT / 2000 / XP. При этом все необходимые основные и динамические данные системы SAP R/3 считываются в RAM (Random Access Memory) пользовательского PC. Это позволяет производить быструю обработку доступных данных эвристическими алгоритмами при построении приемлемого решения. Для пользователей SAP R/3 типичный цикл решения задачи планирования на основе системы SCHEDULE++ состоит из следующих этапов:

- загрузка SAP R/3 данных,
- автоматическое построение начального решения,
- улучшение начального решения (вручную или автоматически),
- запись решений планирования назад в систему SAP R/3.

Система SCHEDULE++ может работать не только как надстройка к ERP системам, но и как самостоятельная система. При этом основные и динамические данные могут быть считаны автоматически или в диалоговом режиме из базы данных предприятия (из баз данных SAP R/3, Oracle, MS Excel). Необходимые браузеры, таблицы, визуальные изображения и пользовательский язык возможно конфигурировать с учетом требований пользователей. SCHEDULE++ может использоваться как в машиностроении, так и в технологической и обрабатывающей промышленности. В Интернете по адресам [2] предоставляется более полная информация о системе SCHEDULE++.

Заключение

Система SCHEDULE++ является компьютерной поддержкой при построении решений в области календарного планирования, составления производственных расписаний, координации цепей поставок (intelligent decision making system for planning, scheduling and supply chain coordination) и демонстрирует всю цепочку Knowledge-Dialogue-Solution. С помощью системы SCHEDULE++ можно обрабатывать и визуально представлять основные и динамические данные, моделировать производство с использованием запрограммированных алгоритмов. В конечном итоге человек оставляет за собой право принятия решений при фактическом планировании производства. В диалоговом режиме (Dialogue) планировщик устраняет конфликты и улучшает построенное эвристическое решение, в результате чего получается окончательный календарный план (Solution). При этом используется интерактивный режим для принятия решения, когда планировщик корректирует эвристическое решение, полученное компьютером, с учетом своего опыта, знания и понимания актуальной производственной ситуации. В системе SCHEDULE++ накопленный опыт производства прошлых лет (Production Knowledge) используется для построения прогнозов на производство в будущем.

Литература

[1] М. Ребшток, К. Хильдебранд. SAP R/3 Менеджмент. Минск, Изд-во: «Новое знание», 2001.
[2] http://www.orsoft.de; http://www.miniapps.net; http://www.x-apps.de

Информация об авторах

Nadezhda Sotskova - Hochschule Magdeburg-Stendal, Fachbereich Wasserwirtschaft, PSF 3680, D-39011 Magdeburg, Germany; OR Soft Jänicke GmbH, Geusaer Str. 104, FH, D-06217 Merseburg, Germany; e-mail: nadezhda.sotskova@orsoft.de
Winfried Jänicke - OR Soft Jänicke GmbH, Geusaer Str. 104, FH, D-06217 Merseburg, Germany; e-mail: marketing@orsoft.de
Wolfgang Thämelt - OR Soft Jänicke GmbH, Geusaer Str. 104, FH, D-06217 Merseburg, Germany; e-mail: wolfgang.thaemelt@orsoft.de

AUTHOR INDEX

Astrakhan State Technical University

with the Association of

the Universities of the Caspian Sea States,
Astrakhan Regional Authorities,
the Scientific-methodical Council of Electrotechnics and Electronics
of the Ministry of Education of the Russian Federation,
the Academy of Electrotechnical sciences of the Russian Federation,
the International Academy of Open Education,
Institute of Informatization of Russian Academy of Education,
Moscow State University of Economics, Statistics and Informatics (Astrakhan Branch)
ITHEA - FOI Institute of Information Theories and Applications (Bulgaria),
Fachhochschule Konstanz Hochschule fuer Technik Wirtschaft und Gestaltung
University of Applied Science (Germany),
Mount Wachusett Community College (USA)

WILL PRESENT
in Astrakhan on October 6-11, 2003

the Sixth international scientific-methodical conference
NEW INFORMATION TECHNOLOGIES in ELECTROTECHNICAL EDUCATION
NITE-2003

THEMES of the CONFERENCE

SECTION 1. Elaboration of electronic teaching aids in the sphere of electrotechnical subjects.

SECTION 2. Application experience of information technologies in the process of teaching electrotechnical subjects.

SECTION 3. Scientific and methodical foundations of the organization of open education

WE INVITE YOUR PARTICIPATION in the CONFERENCE !

For the participation in the conference it is necessary to send papers in Russian or English, 3-5 pages, type size 12 and the information on the author before July 15, 2003 by E-mail: **zain@astranet.ru**

The papers will be published by the beginning of the conference
The papers of participants from Foreign countries are published free of charge (gratis)
The participation in the conference without reports is possible.

Our address:
Russia, 414025, Astrakhan, Tatichev str., 16,
Astrakhan State Technical University, to the Chief of Electrical department Larisa Zaynutdinova
E-mail: zain@astranet.ru
Fax: (851-2) 25-64-27 or (851-2) 25-73-68

The General Sponsor

FOI Bulgaria

Since 1990 FOI Bulgaria has been established as a strong net of companies for service of large scale of users all over Bulgaria. The main activities of FOI Bulgaria are scientific research, design and implementation of large scale of systems for information service.

FOI Bulgaria has become one of the most qualitative structures for scientific research and high quality information service in Bulgaria.

FOI Bulgaria is well known by its:

- Institute for Information Theories and Applications;
- FOI Commerce Group;
- Software "Complex FOI" for information service of accounting and administrative activities.

ITHEA

FOI ITHEA

Institute for Information Theories and Applications

Sofia, 1000, P.O. Box 775, Bulgaria Phone/Fax: (++359 2) 920 19 69,
e-mail: foi@nlcv.net, www.foibg.com

ITHEA main scope of activities

- Scientific projects and publications
- International Conferences "Information Theories and Applications" (ITA)
- International Journal "Information Theories and Applications" (IJ ITA)
- International Prize "ITHEA" for Information Theories and Application

Scientific projects and publications in the field of:

General Information Theory - Philosophy and methodology of the informatics;

Software engineering - New technologies for the software design; Program systems with AI, Pyramidal information systems, Very large data & knowledge bases

Business informatics - Information modelling of the business processes and based on it applied software design;

Informatics IN and OF the Education for Adults - Information research, models and service*;*

Special Applied systems - Multyagent information systems for the government or private organisations.

International Conferences "Information Theories and Applications" (ITA)

ITA conferences are the successor of the series of workshops and conferences organised by unique international scientific group. The beginning is in the workshops organised within 1986-1990 by the International Workgroups (IWG) researching the problems of databases and artificial intelligence. As a result of tight relation between these problems in 1990 in Budapest appeared the scientific group of data base intellectualisation (IWGDBI) integrating the feasibilities of information bases with the creative process support tools. For more than ten years the IWGDBI and ITHEA have organised many scientific events which took place in several East European countries.

International Journal
"Information Theories and Applications"
ISSN 1310-0513 www.foibg.com e-mail: foi@nlcv.net

The progress in Information and Computer Sciences is fuelled by the results of research work and the accumulation of practical experience. The concept "Information Theories & Applications" (ITA) represents the synthesis of this knowledge. The field of ITA is progressing rapidly and is constantly creating new challenges for professionals involved in it.

It is clear that there is continuing need for international forums for exchange of knowledge, experience and creative inspiration among ITA professionals in order to share and stimulate solutions. The "International Journal on Information Theory and Applications" (IJ ITA) is the successor of the scientific co-operation organised within 1986-1992 by international workgroups (IWG) researching the problems of data bases and artificial intelligence. As a result of tight relation between these problems in 1990 in Budapest appeared the scientific group of Data Base Intellectualisation (IWGDBI) integrating the possibilities of databases with the creative process support tools. Heads of the IWGDBI are R.Kirkova (Bulgaria) and V.Gladun (Ukraine).

IJ ITA has been established in 1993 as independent scientific media for publishing original and non-standard ideas. For ten years, IJ ITA became as well-known international journal. Till now, more than 200 papers from more than 400 authors have been published in nine volumes, which are separated in 61 numbers. IJ ITA authors are widespread in 26 countries all over the world: *Bulgaria, Canada, Czech Republic, Egypt, France, Germany, Greece, Hungary, Ireland, Israel, Italy, Japan, Lithuania, Netherlands, Poland, Portugal, Romania, Russia, Scotland, Senegal, Spain, Sultanate of Oman, Turkey, UK, Ukraine, USA.*

IJ ITA major topics of interest include, but are not limited to:

INFORMATION THEORIES

General Information Theory
Philosophy and Methodology of Informatics
Abstract Information Models
Artificial intelligence: Knowledge discovery, Knowledge acquisition and formation, Distributed artificial intelligence, Models of plausible reasoning, AI Planning and Scheduling
Natural language processing
Neuroinformatics
Theory of Computation
Cognitive science
Cognitive graphics
Information models of business activities
Statistical methods
Software engineering and Quality of the programs

APPLICATIONS

Computing
Hyper technologies
Object and Cell oriented programming
Program systems with artificial intelligence
Intellectualisation of data processing
Business Informatics
Information systems: Pyramidal information systems, Intelligent information systems, Very large information systems, Multimedia systems, Business information systems, Graphics systems, Communication systems, Statistical systems, Special applied systems
Computer art and Computer music

Founder and Editor in chief of IJ ITA is Krassimir Markov.

During the years, main co-editors of IJ ITA have been *R.Kirkova, V.Gladun, P.Barnev, Kr.Ivanova*. The IJ ITA Editorial Board also includes the members of the IJ ITA International Conferences Program Committees.

IJ ITA Publisher is FOI-COMMERCE Co., Sofia, Bulgaria.

IJ ITA official language is English.

Subscription for one year:
- for libraries and organisations: EURO 60.
- individual subscription: EURO 20.

Papers accepted by the editorial board of the IJ ITA are published in the following order and publishing fees:
- invited papers - free of charge;
- papers submitted by IJ ITA individual subscribers - EURO 3 per page A4;
- papers submitted from other sources - EURO 7 per page A4.

International Journal on INFORMATION THEORIES & APPLICATIONS
ISSN 1310-0513
Edited by Institute for Information Theories and Applications "FOI ITHEA"
Editor in chief: Krassimir Markov
Publisher: FOI-COMMERCE: Sofia, 1000, P.O.B. 775, Bulgaria; e-mail: foi@nlcv.net

International Prize "ITHEA"

International Prize "ITHEA" is aimed to mark achievements in the field of the information theories and applications.

Prize "ITHEA" is established by FOI Institute of Information Theories and Applications.

Every year, an International Scientific Jury selects the works to be awarded by Prize ITHEA in following divisions: General Information Theory; Software Engineering; Artificial Intelligence; Business Informatics; Computer Art; Special Applied Systems.

The awarded scientists till 2002:

1995	*Sandansky*	K. Bankov, P. Barnev, G. Gargov, V. Gladun, R. Kirkova, S. Lazarov, S. Pironkov, V. Tomov
1996	*Sofia*	T. Hinova, K. Ivanova, I. Mitov, D. Shishkov, N. Vashchenko
1997	*Yalta*	Z. Rabinovich, V. Sgurev, A. Timofeev, A. Voloshin
1998	*Sofia*	V. Jotsov
1999	*Sofia*	L. Zainutdinova
2000	*Varna*	I. Arefiev, A. Palagin
2001	*St.Peterburg*	N. Ivanova, V. Koval
2002	*Primorsko*	A. Milani, M. Mintchev

ADUIS

ASSOCIATION OF DEVELOPERS AND USERS OF INTELLIGENT SYSTEMS

offers

to businessmen, engineers, sociologists, managers - all who use data bases - collaboration in forming analytical information.

- Association has long-term experience in collaboration with teams, working in different fields of *research and development*. Methods and programs created in Association were used for revealing regularities, which characterize chemical compounds and materials with desired properties. Some thousands of high precise prognoses have been done in collaboration with chemists and material scientists of Russia and USA.
- Association can help *businessmen* to find out conditions for successful investment taking into account region or field peculiarities as well as to reveal user's requirements on technical characteristics of products being sold or manufactured.
- *Physicians* can be equipped with systems, which help in diagnosing or choosing treatment methods, in forming multi-parametric models that characterize health state of population in different regions or social groups.
- *Sociologists, politicians, managers* can obtain the Association's help in creating generalized multi-parametric "portraits" of social groups, regions, enterprise groups. Such "portraits" can be used for prognostication of voting results, progress trends, and different consequences of decision making as well.
- Association provides a useful guide in *technical diagnostics, ecology, geology, and genetics.*

ADUIS has at hand a broad range of high-efficiency original methods and program tools for solving analytical problems, such as knowledge discovery, classification, diagnostics, prognostication.

ADUIS unites the creative potential of highly skilled scientists and engineers.

For contacts:

V.M.Glushkov Institute of Cybernetics of NAS of Ukraine

Prospekt Academika Glushkova, 40, 03680 GSP, Kiev 187, Ukraine

Tel. (380+44) 266 22 60, Fax: (380+44) 266 33 48

Email: glad@aduis.kiev.ua