XII-th International Conference

Knowledge-Dialogue-Solution

June 20-25, 2006, Varna (Bulgaria)

P R O C E E D I N G S

FOI-COMMERCE

SOFIA, 2006

Gladun V.P., Kr.K. Markov, A.F. Voloshin, Kr.M. Ivanova (editors)

Proceedings of the XII-th International Conference "Knowledge-Dialogue-Solution" – Varna, 2006

Sofia, FOI-COMMERCE – 2006

ISBN-10: 954-16-0038-7
ISBN-13: 978-954-16-0038-2
First Edition

The XII-th International Conference "Knowledge-Dialogue-Solution" (KDS 2006) 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 ITHEA
Sofia-1090, P.O.Box 775, Bulgaria
e-mail: foi@nlcv.net
www.foibg.com

ISBN-10: 954-16-0038-7
ISBN-13: 978-954-16-0038-2
C\o Jusautor, Sofia, 2006

PREFACE

The scientific Twelfth International Conference “Knowledge-Dialogue-Solution” took place in June, 20-25, 2006 in Varna, Bulgaria. This volume includes the papers presented at the conference. Reports contained in the Proceedings correspond to the scientific trends, which are reflected in the Conference name.

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

Now we can affirm that the 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.

KDS 2006 topics of interest include, but are not limited to:

Cognitive Modelling	Knowledge Engineering
Data Mining and Knowledge Discovery	Logical Inference
Decision Making	Machine Learning
Informatization of Scientific Research	Multi-agent Structures and Systems
Intelligent NL Text Processing	Neural and Growing Networks
Intelligent Robots	Philosophy and Methodology of Informatics
Intelligent Technologies in Control and Design	Planning and Scheduling
Knowledge-based Society	Problems of Computer Intellectualization

The organization of the papers in KDS-2006 is based on specialized sessions. They are:

1. Philosophy and Methodology of Informatics
2. Neural and Growing Networks
3. Ontologies
4. Decision Making
5. Mathematical Foundations of AI
6. Intelligent Systems
7. AI and Education

The official languages of the Conference are Russian and English.

The Program Committee recommends some of the accepted papers for free publishing in English in the International Journal on Information Theories and Applications (IJ ITA).

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

We appreciate the contribution of the members of the KDS 2006 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 Kr.Ivanova, I.Mitov, L. Svyatogor, N.Fesenko and V.Velichko.

V.P. Gladun, A.F. Voloshin, Kr.K. Markov

CONFERENCE ORGANIZERS

National Academy of Sciences of Ukraine
Association of Developers and Users of Intelligent Systems (Ukraine)
International Journal “Information Theories and Applications”
V.M.Glushkov Institute of Cybernetics of National Academy of Sciences of Ukraine
Institute of Information Theories and Applications FOI ITHEA (Bulgaria)
Institute of Mathematics and Informatics, BAS (Bulgaria)
Institute of Mathematics of SD RAN (Russia)

PROGRAM COMMITTEE

Victor Gladun (Ukraine)
Alexey Voloshin (Ukraine)
Krassimir Markov (Bulgaria)

Igor	Arefiev (Russia)	Genady	Osipov (Russia)
Frank	Brown (USA)	Alexander	Palagin (Ukraine)
Vladimir	Donskoy (Ukraine)	Vladimir	Pasechnik (Ukraine)
Alexander	Eremeev (Russia)	Zinoviy	Rabinovich (Ukraine)
Natalia	Filatova (Russia)	Alexander	Reznik (Ukraine)
Constantine	Gaindric (Moldova)	Vladimir	Ryazanov (Russia)
Tatyana	Gavrilova (Russia)	Galina	Rybina (Russia)
Krassimira	Ivanova (Bulgaria)	Vasil	Sgurev (Bulgaria)
Vladimir	Jotsov (Bulgaria)	Vladislav	Shelepov (Ukraine)
Julia	Kapitonova (Ukraine)	Anatoly	Shevchenko (Ukraine)
Vladimir	Khoroshevsky (Russia)	Ekaterina	Solovyova (Ukraine)
Rumyana	Kirkova (Bulgaria)	Vadim	Stefanuk (Russia)
Nadezhda	Kiselyova (Russia)	Tatyana	Taran (Ukraine)
Alexander	Kleshchev (Russia)	Valery	Tarasov (Russia)
Valery	Koval	Adil	Timofeev (Russia)
Oleg	Kuznetsov (Russia)	Vadim	Vagin (Russia)
Vladimir	Lovitskii (GB)	Jury	Valkman (Ukraine)
Vitaliy	Lozovskiy (Ukraine)	Neonila	Vashchenko (Ukraine)
Nadezhda	Mishchenko (Ukraine)	Stanislav	Wrycza (Poland)
Ilia	Mitov (Bulgaria)	Nikolay	Zagoruiko (Russia)
Xenia	Naidenova (Russia)	Jury	Zaichenko (Ukraine)
Olga	Nevzorova (Russia)	Arkady	Zakrevskij (Belarus)

TABLE OF CONTENTS

Ontologies

Decision Making

Mathematical Foundations of AI

Intelligent Systems

AI and Education

INDEX OF AUTHORS

СОЗНАНИЕ, ПОДСОЗНАНИЕ И ЭМОЦИИ, ДУША И KDS

Зиновий Л. Рабинович, Юрий А. Белов

***Аннотация**: В докладе обсуждаются некоторые нейрофизиологические и другие феномены, интерпретация которых хоть и косвенно, но убедительно подкрепляет разрабатываемые авторами концептуальные представления в части организации памяти в мозге человека и протекающих в ней процессов.*

***Ключевые слова**: память, душа, сознание, подсознание, эмоции, решение проблем.*

Вступление

Прежде всего, отметим, что термины в названии доклада подразумеваются авторами работы как энциклопедические (взятые из «Энциклопедического словаря» издания 1981 г.), т.е. обычные, обиходные, не подверженные каким бы то ни было модернистическим толкованиям.

Это тем более важно, что как весьма часто встречается, все споры относительно природы, как естественных механизмов подсознания, сознания и души (зачастую вообще идеалистически отделяемой от ее естественной сущности), как раз и возникают из-за разного их понимания и толкования. В указанном смысле сосредоточением понятий, выраженными данными терминами, является именно мозг, изучение которого названо в науковедческой литературе первой из трех главных научных проблем современности (остальные две – государство и вселенная). И это изучение подвержено двум фундаментальным целям, а именно: разностороннему познанию и, соответственно, прикладному использованию, в том числе и в искусственном интеллекте. С нашей точки зрения, именно такое бионическое использование представляет самый существенный интерес для проблематики KDS (Knowledge, Dialog, Solution) – поскольку оно заключается в рациональном заимствовании для информационных механизмов и технологий, определенных свойств и особенностей, механизмов мышления, как обработки информации в памяти мозга (далее – Память). Вместе с тем проблематика KDS сама по себе также представляет существенный интерес и для познания механизмов человеческого мышления. Действительно, человеческое мышление определяется и как обработка чувственной информации, и как работа над знаниями, представленными в языке, что уже является определяющим атрибутом осознаваемого мышления. Обо всем этом будет сказано дальше, здесь мы лишь особо отметим, что процессы, протекающие в душе человека, как в его внутреннем мире (по энциклопедическому определению) как раз и состоят из сочетания процессов неосознаваемого и осознаваемого мышления, т.е. работой над знаниями, что и охватывается первым символом K в аббревиатуре KDS. Далее мышление, как таковое, происходит в двухстороннем общении с внешним по отношению к мыслящей субстанции средой, иначе говоря, – в диалоге с ней, что и покрывается вторым символом D в KDS. И, наконец, третий символ S, подразумевающий решение проблем, который как раз и является самой сутью человеческого мышления (в том числе его главной для искусственного интеллекта составляющей, направленной на решение конкретных проблем). Таким образом, подведение под проблематику KDS изучения механизмов мышления, как одной из главных проблем кибернетики (в ее классическом, традиционном смысле) оказывается весьма полезным с учетом еще и того чрезвычайного обстоятельства, что проблематика KDS уже оснащена эффективными математическими формализмами – теорией представлений и обработки знаний, теорией семантических сетей, теорией возможности и ее применениями при решении

задач моделирования и принятия нечетких решений, нечеткого оптимального оценивания, теоретико-возможностного анализа, обработки и интерпретации эксперимента и т.д.

В докладах авторов на предыдущей конференции KDS были изложены концептуальные представления об опознании образов и решении проблем в памяти человека [1], как гипотетические, но, в тоже время, подкрепляемые математическим компьютерно-модельным исследованием когнитивных процессов в памяти мозга [2]. При этом своем исследовании авторы опирались и использовали экспериментальные сведения, которые им были любезно предоставлены нейрофизиологом докт. биол. наук, доцентом МГУ им. М.В. Ломоносова Г.С. Воронковым, с которым их связывает давнее творческое научное сотрудничество.

В настоящем докладе авторы дополнительно приводят некоторые нейрофизиологические феномены (в том числе и обнаруженные совсем недавно), интерпретация которых хоть и косвенно, но убедительно подкрепляют наши концептуальные представления в части организации памяти в мозге человека и протекающих в ней процессов, что позволяет, по нашему мнению, перевести их из категории правдоподобно-гипотетических в категорию достоверных.

Сознание, подсознание, эмоции и душа

Не повторяя в целом упомянутое выше концептуальное представление, напомним лишь его основную сущность в той мере, которая необходима для дальнейшего понимания и изложения предмета доклада. Итак, что же собой представляет память мозга (Память), как вместилище накопленных знаний, подсознания, сознания и души в целом? Как известно, Память – лишь часть нейронной сети мозга, а, именно, та часть, в которой образуются следы поступающей информации, воспроизводимой при их возбуждении. Остальная же часть нейронной сети всего организма лишь поставляет информацию в Память, соответственно преобразуя для этого входные сигналы из внешней среды, но никак их не запоминая. Поскольку поставляемая в Память информация обладает колоссальной дискретностью, она для возможности ее отображения должна быть обобщена, иначе говоря, структурирована.

Обобщение информации осуществляется образуемыми в базисе нейронов и связей между ними иерархическими пирамидальными структурами, основания которых составляют все поступающие сенсорные данные, а верхушки их – собственно обозначения. Таким образом, пирамида образа состоит из иерархии подобразов, которые могут входить и в другие образы, ассоциативно с ним связанные. Направленность связей определяет распространение возбуждения, т.е. указанное обобщение осуществляется направлением возбуждения снизу-вверх. Но наличие в пирамиде образа только таких связей является недостаточным, и его построение, как запомненной в памяти структуры, требует еще наличия в пирамиде образа и обратных, т.е. нисходящих связей от самой верхушки ее вплоть к верхушкам подобразов до основания.

Это требование исходит из фундаментальной гипотезы, утверждающей, что воспроизведение запомненного образа осуществляется возбуждением тех же компонент Памяти, которыми был образ, воспринят извне, т.е. представляющих основание его пирамиды.

Это совершено конкретное утверждение, которое, прежде всего, исходит из здравого смысла в менее конкретном, т.е. более общем виде, несвязанным с пирамидальными структурами образов уже встречается в литературе последнего времени, как объясняющее обнаруженное наличие и роль обратных связей от «верхних» отделов коры мозга к «нижним» его отделам, воспринимающим внешнюю исходную информацию.

В целом же, приведенная гипотеза (названная авторами главной) определяет границу в нейронной сети мозга между «непамятью» и самой Памятью, которая таким образом начинается там, где кончаются обратные нисходящие связи. В этом смысле память мозга подобна схемной (аппаратной) памяти ЭВМ, образуемой замыканием выходов на входы (петлями внутри памяти). Возбуждение выходов в Памяти, т.е. верхушек пирамид, распространяясь по обратным связям, приводит к возбуждению составляющих их компонент, вплоть до самых оснований, что и есть, согласно данной гипотезе, явлением воспоминания, как действия, так называемого, умственного взора. Таким образом, умственный взор выхватывает из множества граничных компонент Памяти определенную их комбинацию, обозначенную верхушкой пирамиды образа. И это воспоминание определяет и узнавание предъявляемого образа – поскольку, если

в Памяти имеет месть его полная пирамида (со всеми восходящими и нисходящими связями), то вначале, по предъявлению образа возбуждение передается по восходящим связям, но по мере его продвижения к вершине пирамиды, оно уже начинает передаваться по нисходящим связям к ее основанию. Таким образом, процесс узнавания образа характеризуется динамически реализуемыми его композицией и несколько отставаемой по фазе декомпозицией. Следовательно, по мере прохождения этого процесса, все составные компоненты образа кроме лишь его самой верхушки возбуждаются дважды, и всплеск второго возбуждения как раз и означает узнавание образа. Если же образ в Памяти не запомнен, то при его предъявлении возникает лишь процесс композиции, как формирование его в Памяти по прямым (восходящим) связям (видит мозг, а не только глаз), а процесса декомпозиции не будет, поскольку обратные (нисходящие) связи отсутствуют – следовательно повторного возбуждения компонент принимать образ не будет, т.е. образ, в целом, будет воспринят, как новый, незнакомый (но именно – в целом).

Перевод же такого образа в запомненный потребует повторных предъявлений его, с тем, чтобы образовались обратные связи путем проторения генетически заложенных связей или их прорастания (что, как известно, обнаружено в нейрофизиологии). В результате такого обучения и формируется Память. Приведенное выше авторами работы представление является, по сути, интерпретацией уже известных в нейрофизиологии и психологии фактов, но содержит некоторые умозрительные, т.е. гипотетические заключения и выводы, которые связывают эти факты причинно-следственными отношениями, что и обуславливает убедительность этих заключений. Вместе с тем можно привести и некоторые косвенные доказательства истинности приведенных нами выше представлений, которые не поддаются прямой экспериментальной проверке. Так, например, активная фаза сна, во время которой и существуют сновидения, обуславливается быстрыми движениями глаз (БДГ), которые возникают и управляются специальным центром (четверохолмием), находящимся на таламусе, на который и поступает как первичная зрительная информация, так и обратные связи от коры, в которой она обрабатывается. Т.е. в таламусе содержится самый низжий слой Памяти, возбуждение компонент которого и осуществляется в активной фазе сна, сопровождаемое БДГ, как и наяву. Сновидения представляют собой возникающие образы – следовательно, возникновение сновидений как раз и подтверждает как наличие первичных компонент запомненных образов в нижнем слое Памяти, так и их воспроизведение, как составных частей в самых разнообразных комбинациях, представляющих сновидения. Указанные «наличие» и «воспроизведение» как раз и являются ключевыми моментами приведенной исходной гипотезы.

Однако сам процесс воспроизведения (как именно это возбуждение происходит) не подпадает под данную гипотезу и, собственно, относится уже к самому образованию всплеска возбуждения компонент образа, приводящего к его воспроизведению.

Гипотетическое предположение того, что этот всплеск происходит в результате декомпозиции образа нашло подтверждение при компьютерном моделировании процесса восприятия запомненного образа, проведенного с использованием реальных нейрофизиологических данных, которое и обнаружило колебательный процесс, возникающий только при наличии обратных декомпозиционных связей[2]. Но имеется и непосредственное фактическое подтверждение истинности гипотезы об узнавании образа, именно как результата этого всплеска. Это подтверждение обнаруживается рассмотрением явлений происходящих в параграфе (так называемом детекторе лжи). Испытуемому задают вопросы об предполагаемых известных ему событиях и контролируют точными приборами его физиологическое состояние. Если событие испытуемому известно, то значит оно структурно запечатлено в его Памяти. И вопрос об этом событии, т.е. введение в его Память образа события вызовет автоматическую реакцию его узнавания, выражаемую декомпозиционным всплеском. Если же событие неизвестно, т.е. его образа, запечатленного в Памяти нет, то и всплеска не будет. Происшедшее событие может быть запечатлено только в подсознании. Наличие же всплеска фиксируется приборами (в частности, проявляется изменением энцефалограммы). Таким образом, независимо от признания или отрицания испытуемым свершения предполагаемого следователем события, его истинность обнаружится наличием определенной физиологической реакции. Эта реакция косвенно указывает на наличие всплеска при узнавании вновь предъявляемого, но уже запечатленного в Памяти образа, что и является подтверждением истинности гипотезы об этом процессе.

Узнавание образов, как самая элементарная функция Памяти, неотделимая от самого этого понятия, свойственна любым организмам, обладающих ею. Но Память является тем замкнутым пространством,

содержащим в структурном виде разнообразные знания, в котором и осуществляется все мышление, понимаемое в широком смысле как работа над знаниями, включая их восприятие, анализ, накопление, синтез и т.п. При этом особо подчеркнем, что определяющим фактором природы мышления является представление знаний в двух формах – структурно-образной и языковой. Причем у Человека языковая форма столь существенно развита по сравнению с низшими, обладающими памятью организмами, что, возможно, это различие, как переход количества в качество, и делает человека Человеком. При этом подчеркнем, что эти две формы представления знаний локализуются в разных разделах пространства Памяти, которые, соответственно, названы «Сенсориумом» и «Языковой Системой» [3], а также и в объединяющей их общей высшей ассоциативной системе. Переходя теперь непосредственно к предмету доклада (как он указан в названии), прежде всего, отметим, что процессы мышления подразделяются на осознаваемые и неосознаваемые. Зададимся следующим вопросом: что же собой представляет осознаваемое мышление, как процесс работы со знаниями в Памяти, происходящий на уровне Сознания? Применительно к предмету доклада ответ на этот вопрос должен, прежде всего, исходить из использования той стороны многогранного понятия сознания, которое относится к его структурному воплощению в памяти мозга, причем, именно, человеческого (понятие это не снабженное специальными дополнениями, относится, именно, к Человеку). И это структурное воплощение должно обеспечивать возможность социального общения Человека, как пребывания его в мире людей. А средством такого информационного общения является Язык. В нем и выражены все знания, с которыми оперирует осознаваемое мышление. И эти знания как раз и заключены в языковой системе мозга Человека (и только такие знания попадают под энциклопедическое понятие «Знания»). Но как уже отмечалось выше, все объектные представления запомненные в чувственной памяти (Сенсориуме) также отнесены к Знаниям (хотя и не явным). В мыслительных процессах осуществляется путем передачи возбуждения обмен информацией между сенсорной и языковой системами мозга. Таким образом приходим к следующему заключению (которое и является полным ответом на поставленный вопрос), что процесс осознаваемого мышления осуществляется возбуждением динамических структур, состоящих из компонент и Сенсориума и Языковой системы. Причем, этот процесс – именно последовательный, поскольку возбуждение в языковой системе приводит к произнесению слов, хоть и умственному, но вызываемому всеми необходимыми для этого командами нервной системы (что также экспериментально установлено в нейрофизиологии).

Данный процесс на уровне сознания, собственно, и соответствует энциклопедическому понятию «мышление». Но мышлению в широком понимании этого термина, как, вообще говоря, информационному процессу работы со знаниями в Памяти (в какой бы форме они не представлялись), свойственны процессы в Сенсориуме, которые не зависят от процессов осознаваемого мышления, которое происходит обязательно с участием Языковой системы. Процесс мышления в сенсориуме, не будучи связанным с последовательным произношением слов может быть в целом глубоко распараллеленным или, более строго говоря, распределенным. В мышлении неосознаваемом (т.е. без участия в нем языковой системы) этот процесс может иметь и чисто спонтанный характер и быть инициированным неосознанными желаниями.

В осознаваемом мышлении процесс в сенсориуме приобретает определенную целеустремленность в силу функционального соотношения между логическим и образным мышлением, соответственно, превалирующими в Языковой системе и Сенсориуме. Сохранение в этом случае возможности распараллеленной обработки знаний не случайно. Как показали психологические исследования, в мыслительной деятельности человека его неосознаваемая компонента приблизительно в 10-15 раз превышает осознаваемую. Следовательно, подсознание, на уровне работы со знаниями, при этом, не выходя на уровень сознания, но являющееся, по сути, его основой, имеет весьма большее удельное значение в мыслительной деятельности Человека. Сказанное особо наглядно характеризуется явлением целенаправленного мышления, как процессом решения проблемы, заданной исходной и целевой ситуациями, модельно отображаемыми в Памяти Человека на уровне сознания. Суть мышления, согласно его классической трактовке, именно и заключается в решении Проблем, вовсе не обязательно конкретно обозначенных, как в рассматриваемом случае, а лишь просто означающих достижение чего-то желаемого, отправляясь от действительного. Рассматриваемый же случай является наиболее интересным как для познания человеческого мышления, так и для технического применения этих знаний, т.е. бионического подхода к развитию искусственного интеллекта. Именно поэтому гипотетическая концептуальная модель

решения Проблемы в памяти мозга впервые изложеная еще в 1979 году [4], а затем развитая в дальнейших публикациях З.Л. Рабиновичем, в том числе в соавторстве и с Г.С. Воронковым [5] и Ю.А. Беловым [6], вызвала большой интерес у нейрофизиологов, психологов и кибернетиков. Повторим кратко ее основные черты, необходимые для дальнейшего изложения материала доклада.

Исходная и целевая ситуации решаемой Проблемы (термин «модель» будем лишь подразумевать) представлены в Памяти возбужденными структурами, что, как известно из нейрофизиологии, благодаря так называемой эмоции интереса (и, вообще, удовлетворения потребности), вызывает процесс достижения целевой ситуации. Что же он из себя представляет? Для наглядности изложения ответа на этот вопрос еще в указанной публикации введено понятие «Генератора Проблемы» (ГП), как модели, объединяющей исходную и целевую ситуации, которые, соответственно, являются его полюсами. Таким образом, ГП выступает как поддерживающий наличие для решаемой Проблемы разницы потенциалов на его полюсах (символизирующей упомянутый интерес). Решение же Проблемы осуществляется образованием цепи, замыкающей исходную и целевую ситуации причинно-следственными связями или, иначе говоря, преобразованием исходной ситуации в целевую – в общем случае, через образование ряда промежуточных ситуаций. Как же возникает указанная цепь? Исходная и целевая ситуации, как указывалось, являются осознанными, хотя и отображенными в сенсориуме, но и заданными в языке. А это уже приводит к логическим рассуждениям, как последовательному образованию звеньев цепи замыкания, причем, возможно, осуществляемому и от обоих его полюсов. Каждый шаг такого построения содержит в себе и элемент догадки, как поиска нужного сведения в Памяти, либо даже образования в ней нового знания. Т.е. каждый такой шаг содержит в себе осознаваемый вопрос и получаемый в результате догадки ответ. А догадка – это уже проявление интуиции.

Если же в результате всего процесса решение Проблемы приобретается и фиксируется в Языке Новое Знание, то он уже, собственно, является осознаваемым и творческим.

В интуитивной догадке главную роль, по-видимому, выполняют процессы в сенсориуме, т.е. на уровне подсознания, но будучи возникшей, она уже проникает на уровень сознания, как восполняющая пробел в цепи ГП, что при большом значении этой догадки воспринимается как озарение. И это озарение будет внезапным (как замечательным психологическим феноменом), если в момент его появления ГП не возбужден, т.е. в языковой системе процесс решения Проблемы отсутствует, что и означает отсутствие текущего осознавания работы над знаниями. Но чтобы это осознание вновь возникло (т.е. возбудился ГП) – возбуждение структур в сенсориуме (т.е. на уровне подсознания) при озарении должно быть столь сильным, чтобы благодаря связи Сенсориума с языковой системой оно и возобновило возбуждение ГП, включая в него недостающее звено. Таким образом, в мышлении при решении Проблем осуществляется работа над знаниями, происходящая во взаимодействии сенсориума и языковой системы, т.е. не уровне неосознаваемом и осознаваемом. Эта работа осуществляется базисными операциями с вертикальными двухсторонними передачами возбуждений в пирамидальных структурах Памяти, охватывающей операциями связи между структурами Памяти, как вертикальными так и горизонтальными, в результате которых и формируется цепь структур, замыкающих полюса ГП, означающий решение Проблемы.

Эта цепь – внешняя по отношению к ГП, образует в данном процессе (совместно с возбуждением структур ГП для его «поддержания») циклические маршруты возбуждения в Памяти, которые, по-видимому, именно и наблюдались в нейрофизиологических исследованиях, как сопровождающие мышление. Кроме такого, фактически косвенного экспериментального подтверждения, приведенного гипотетического концептуально-модельного представления о структурном воплощении в Памяти процесса целенаправленного мышления, истинность (или, скажем так, осторожнее, правдоподобие) этого представления убедительно подкрепляется и различными нейропсихологическими феноменами. Например, чем более отличаются исходная и целевая ситуации Проблемы, тем больше требуется умственных усилий (осуществляемых указанными операциями по ее решению), еще например, явление получения предполагаемого решения Проблемы, как внезапного озарения, влияние эмоционального фактора на процесс решения Проблемы (о котором будет сказано далее) и т.п.

И, вообще, все мышление представимо подобными же моделями, но лишь в более широко очерченных формах, выражающих «желаемое и действительное». Согласно самому понятию «мышления», которое осуществляется как осознаваемое и неосознаваемое во взаимодействии этих процессов, первый из

которых воплощается в сознании и охватывает обработку информации во всей Памяти, а второй – в подсознании, являющийся прерогативой сенсориума и невыражаемый в Языковой Системе.

Таким образом, сознание не отделимо от подсознания, в котором представлены все чувственные образы, а подсознание может восприниматься, как отдельный субъект мышления, хотя и участвующий в осознаваемом мышлении, но и действующий независимо от него (под влиянием внешней среды или спонтанно).

Следовательно, внутренний мир Человека содержит как сознание (что само собой разумеется), так и подсознание, и его можно называть душой Человека – согласно обиходному и энциклопедическому пониманию этого термина, а также материалистической его трактовке (структурной реализации в мозге), положенной в основу излагаемой концепции.

Но внутренний мир Человека, т.е. его душу заполняют не только сознание и подсознание, как, собственно, смысловые категории, но и проявление эмоций, вызываемые эмоциогенными органами воздействующие с помощью химических медиаторов непосредственно на физиологические характеристики компонент Памяти (пороги возбуждений базисных элементов, «проводимость» связей между ними). Причем химические медиаторы выделяются, в основном, в нижних слоях Памяти (в ее, так называемом, «животном поле», свойственном, но уже в качестве верхних слоев и животным, обладающим центральной нервной системой). И вот такого рода состояние Памяти, причем еще локализованное, т.е. связанное с возбуждением определенных ее структур (уже электрическими медиаторами) и порождает и соответствующее настроение как ощущение. Например, интерес, вдохновение, удивление, тревога и т.д. Таким образом, Память как бы имеет два класса выходов – главный смысловой и настроенческий эмоциональный, в котором уже главную роль выполняет подсознание. На входах же Памяти рецепторные сигналы также имеют так называемую «эмоциональную» окраску» в зависимости от того, какими воздействиями они были сформированы и каким комплексом медиаторов выражены.

Таким образом, мышление на уровне сознания и подсознания не просто осуществляется в среде Памяти, но и подвергается воздействию изменчивости этой среды, в свою очередь, также оказывая влияние на эту изменчивость. Вот такая диалектика.

В данном смысле, весьма характерным примером является уже приводимый процесс целенаправленного мышления. На этот процесс оказывает сильное влияние эмоция интереса как движущая сила человеческого мышления вообще. И она, оказываясь в повышении «разности потенциалов» ГП и «проводимости» связей между возбуждаемыми ГП структурами и т.п. способствует этим появлению вдохновения, как высшего эмоционального благотворного воздействия на этот процесс.

Таким образом, душа Человека является сосредоточением трех взаимосвязанных компонент – сознания, подсознания и системы эмоций, каждая из которых весьма индивидуализирована в каждом человеке; что, однако, и обуславливает важность определения общеконцептуальных парадигм внутреннего мира Человека, как для их познания, так и для его прикладного использования в самых различных видах человеческой деятельности.

В соответствии с предметом доклада, погружая познание Души в проблематику КДС, главным образом остановимся на ее первой проблеме, относящейся к знаниям, над которыми работает и сознание, и подсознание Человека. И в этом плане, но, несколько отвлекаясь от буквального изложения предмета доклада, приведем одно сенсационно интересное сообщение, дополнительно подтверждающее правильность принятия за основу соотношений между понятиями знаний, языка, их представления и эмоций.

Среди всех природных организмов только Человеку свойственно такое богатство языка и знаний, на нем выражаемых. И, по определению, т.е. только по договоренности, это отнесено к понятию души Человека. А может это и не так? Может быть это плохая договоренность, а душа человека нечто другое, а, именно, независимое от знаний и языка понятие в какой-то степени, свойственное и другим организмам, не обладающим человеческими качествами, относящимися к Знаниям и Языку их представления. Так вот недавно проведенные во Львове эксперименты показали (газета «Факты», 17 марта 2006 года), что свечение вокруг головы человека (аура) продолжается трое суток после его ухода из жизни, что подтвердило результаты исследований профессора К. Короткова из Санкт-Петербургского государственного технического университета информационных технологий, точной механики и оптики. Но

мало того, что эти исследования подтвердили открытие К. Короткова (которое еще не получило признание официальной наукой), они еще и показали, что такое свечение у погибших животных совершенно отсутствует, т.е. исчезает сразу же после их смерти. О чем это говорит? А том, что свойственное только Человеку развитое осознаваемое мышление (как обработка знаний, представляемых в языке) дополняется еще вторым, свойственным только Человеку признаком, а имено, инерцией ауры. Естественным образом возникает вопрос: не является ли эта аура материальной полевой формой информационного представления души Человека (что, вообще говоря, укладывается в теорию ноосферы В.И. Вернадского, в явления телепатии, телекинеза и т.п.)? Но если это так, то проведенная интерпретация понятия души относимого только к человеку, получает еще дополнительное подтверждение (в чем, впрочем, для прикладных технических аспектов, охватываемых КДС, в этом, в нынешнее время, пожалуй, нужды нет, но в целях познания это интересно).

Теперь непосредственно перейдем к возможности бионического подхода к представлению и обработке знаний в искусственном интеллекте.

Как уже говорилось, знания в Памяти представимы комплексом сетей пирамидальных структур, состоящие из двух видов нейронных сетей – чувственных образов (Сенсориума) и словесных понятий (Языковая система), имеющих общие области на их высших уровнях. Эти сети образуются базовыми операциями связи – между возбуждаемыми элементами этих структур, по которым и распространяются возбуждения, причем эти связи являются вертикальными двухсторонними в пределах каждой пирамиды и горизонтальными, связывающие остальные пирамиды между собой. Т.е. нейронные сети начинаются от входов информации в Память, но не кончаются отдельными торчащими головками пирамид, а представляют сложную «паутину» их сплетений. И в этой «паутине» и высвечиваются динамически образы, которые либо опознаются, если они в ней имеются, либо осознаются, как новые, еще неизвестные. В Сенсориуме концентрируются в основном декларативные знания, а в Языковой системе - и декларативные, и управляющие.

Бионический подход к представлению знаний и работа над ними в искусственном интеллекте означает, прежде всего, концептуальное моделирование того, что есть в Памяти, вернее наших представлений об этом. И здесь, собственно, имеются два пути – структурная и программная реализации. Первая, в силу непреодолимых технических, а также информационно-технологических трудностей, практически возможна лишь для ограниченных специализированных применений. Вторая же, которая может оказаться и не столь эффективной в работе, как первая, является, собственно, универсальной, ограниченной только возможными количественными параметрами, что уже определяется мощностью применения вычислительных средств.

Таким образом, для решения различных задач, относимых к искусственному интеллекту (из которых, пожалуй, главнейшие – задачи решения проблем), практически и эффективно применимо именно вторая универсальная реализация. Но она осуществляется, как моделирование структур и процессов только в Языковой системе Памяти – поскольку вся обрабатываемая программно информация должна иметь словесное представление.

Конечно, здесь нужен математический аппарат для представления указанной сети пирамидальных структур и действий в этой сети. Основой такого аппарата, вполне и эффективно, могут являться уже упоминавшиеся ранее растущие пирамидальные сети (РПС). Действительно, в них образы, как сочетание признаков, предъявляются соответствующими пирамидами, связи между которыми осуществляются общими ассоциативными элементами и передачами возбуждения. Математический аппарат РПС уже успешно использовался для решения ряда практических задач – синтеза химических соединений, классификации, распознавания по аналогии и ряда других, что изложено в известных публикациях В.П. Гладуна и участников его научной школы. Это же аппарат был апробирован как для моделирования некоторых процессов в мозге (а, именно, в упоминавшейся модели процесса опознавания образа), так и (с соответствующим развитием) в создании методов проектирования высокопроизводительных ЭВМ на базе бионического подхода [7,8].

В последнее время не без влияния требований, исходящих от условий развития этого подхода, аппарат РПС был расширен введением в него процедур декомпозиции (они же дедуктивные и дивергентные), и композиционных процедур (они же индуктивные и конвергентные).

Также желательно создание распараллеленных вариантов решения задач искусственного интеллекта с использованием процедур на основе бионического подхода ориентированных на реализацию этих вариантов на базе создаваемых в последнее время сверхвысокопроизводительных мультимикропроцессорных ЭВМ (с кластерной архитектурой), в которых, как с уверенностью можно предположить [9], должна эффективно отображаться среда мышления. Такие машины, как первые модели украинского ряда кластерных суперЭВМ, уже созданы [10] и развитие этого ряда продолжается [8,9].

И пусть дальнейшее развитие в этом направлении будет воспринято как эпитафия по внезапно ушедшему из жизни главному конструктору этих машин, выдающемуся ученому, участнику прошлых конференций КДС профессору Валерию Николаевичу Ковалю.

Литература

[1] З. Л. Рабинович. Концептуальное представление об опознании и решении проблем в памяти человека и возможностях его использования в искусственном интелекте. XI-th International Conference KDS-2005 Proceedings V.1, FOI-Commerce, Sophia, 2005, pp. 1-8.

[2] Yuriy A. Belov, Sergiy V. Tkachuk, Roman V. Iamborak. Mathematical and Computer Modeling and Research of Cognitive Processes in Human Brain. Part II. Applying of Computer Toolbox to Modeling of Perception and Recognition of Mental Pattern by the Example of Odor Information Processing, XI-th International Conference KDS-2005 Proceedings V.1, FOI-Commerce, Sophia, 2005, pp. 32-36.

[3] Воронков Г. С., Рабинович З. Л. Сенсорная и языковая система – две формы представления знаний // Новости искусственного интелекта 1993, №2, с. 116-124.

[4] Рабинович З. Л. Некоторый бионический подход к структурному моделированию целенаправленого мышления // Кибернетика 1979, №2, с. 114-118.

[5] Рабинович З. Л., Воронков Г. С. Представление и обработка знаний во взаимодействии сенсорной и языковой нейросистем человека // Кибернетика и системный анализ 1998, №2, с. 3-11.

[6] Zinoviy L. Rabinovich, Yuriy A. Belov. Conceptual Idea of Natural Mechanism of Recognition, Purposeful Thinking and Potential of Its Technical Application. Mechanisms, Symbols and Models. First Intern Work-Conference on the Interplay, Spain, June, 2005, Proceedings, Part I.

[7] Рабинович З. Л., Ященко В. А. Подход к моделированию мыслительных процессов на основе нейроподобных растущих сетей // Кибернетика и системный анализ 1996, №5, с. 3-20.

[8] А. А. Морозов, Ященко В. А. Интелектуализация ЭВМ на базе новго класса нейроподобных растущих сетей // ИПММС НАН Украины 1997, 125.

[9] Рабинович З. Л. О концепции машинного интелекта и его развития // Кибернетика и системный анализ 1995, №2, с. 163-173.

[10] Valeriy Koval, Sergey Ryabchenko, Volodimyr Savjak, Ivan Sergienko, anatoliy Yakuba, XI-th International Conference KDS-2005 Proceedings V.1, FOI-Commerce, Sophia, 2005, pp. 98-103.

[11] Ю. А. Белов, В. П. Диденко, О. Е. Цитрицкий и др. Математическое обеспечение сложного эксперимента, Киев: «Наук. думка», 1982-1990, т.1-5.

Информация об авторах

Зиновий Львович Рабинович – *доктор технических наук, профессор, Институт Кибернетики НАН Украины имени В. М. Глушкова, Киев-187, проспект Акад. Глушкова, 40; e-mail: eco@public.icyb.kiev.ua*

Юрий Анатолиевич Белов – *доктор физико-математических наук, профессор, Киевский Национальный университет имени Тараса Шевченко, Украина, Киев-680, Академика Глушкова 2 корп. 6; e-mail: belov@ukrnet.net*

BASIC STRUCTURE OF THE GENERAL INFORMATION THEORY

Krassimir Markov, Krassimira Ivanova, Ilia Mitov

***Abstract**: The basic structure of the General Information Theory (GIT) is presented in the paper. The main divisions of the GIT are outlined. Some new results are pointed.*

***Keywords**: General Information Theory.*

***ACM Classification Keywords:** A.1 Introductory and Survey*

Introduction

There exist several common theoretical information paradigms in the Information Sciences. May be, the most popular is the approach based on the generalization of the Shannon's Information Theory [Shannon, 1949], [Lu, 1999]. Another approach is the attempt to be synthesized in a common structure the existing mathematical theories, which are applicable for explanation of the information phenomena [Cooman et al, 1995].

Besides of this, we need to point the diligence of the many researchers to give formal or not formal definitions of the concept "information". Unfortunately, although they are quite attractive in some cases, these definitions did not bring to any serious theoretical results [Abdeev, 1994], [Bangov, 1995], [Markov P., 2002], [Tomov, 1991], [Elstner, 1993].

At the end, there exist some works that claim for theoretical generality and aspire to be a new approach in the Information Science, but theirs authors should clear up what they really talk about [Burgin, 1997].

The theoretical base of the informatics needs the philosophical support and substantiation to become wide accepted scientific paradigm. This way, the closely scientific research in the domain of informatics would be able to leap across its boundaries and to become as elements of the scientific view of life.

Discovering the common philosophical base has exceptional importance.

The philosophical rationalizing and substantiating of the information phenomena become as leading goal of the scientific knowledge.

Starting point need to be the consideration that the General Information Theory (GIT) needs to be established as internal non-contradictory logical system of contentions [Markov et al, 1993]. This rule contrasts the understating of the informatics as a mosaic of formal theoretical works and applications.

Basic requirement is that the GIT needs to explain the already created particular theories and paradigms.

The mathematical structures ought to serve as a tool for achievement the precise clearness of the philosophical formulations and establishing the common information language for describing and interpreting the information phenomena and processes.

The second very important requirement is to build the GIT on the base of the inceptive philosophical definition of the concept "information" using as less as possible the primary undefined concepts with maximal degree of philosophical generalization. This requirement follows the consideration that ***the concept "information" is not mathematical concept.*** The behavior, peculiarity and so on could be described by the mathematical structures but this is another problem. In this case, the accent is stressed on the comprehension that the information has purely material determination and that it is a consequence of the interaction between the material objects as well as of the real processes and phenomena occurred in them and with them.

The presented in this paper General Information Theory (GIT) is based only on primary consideration of the world as variety of entities, which are formed by relationships between entities that form lower levels.

The development of GIT had started in the period 1977-1980. The first publication, which represents some elements of GIT, was published in 1984 [Markov, 1984]. The establishment of GIT was not rectilinear. Occasionally, the influences of other paradigms have disturbed this process and have turned it to the vain effort [Burgin, Markov, 1991].

The fundamental notion of the GIT is the concept "Information". All other concepts are defined based on this definition. In 1988, the not formal definition of the concept of Information was published in [Markov, 1988].

It became as a fundamental definition for the General Information Theory [Markov et al, 1993], [Markov et al, 2003a]. The translation of the philosophical theory into the formal one is a good approach for verification of the scientific ideas [Markov et al, 2003b], [Markov et al, 2004]. Because of this, the basic concepts of the General Information Theory ware presented philosophically and formally.

This paper is aimed to present the internal structure of GIT in its current state. For this purpose we will remember some main results as well as we will discuss the new achievements of GIT.

The GIT is build by three specialized theories:

- Theory of Information,
- Theory of Infos,
- Theory of Inforaction.

Theory of Information

The fundamental notion of the General Information Theory is the concept "Information". All other concepts are defined based on this definition. The first not formal definition of the concept of Information was published in [Markov, 1988]. The main philosophical explanations were published in [Markov et al, 1993]. Several attempts to develop a formal definition were introduced during the years [Markov et al, 2003b], [Markov et al, 2004].

Entity

In our examination, we consider *the real world* as a space of *entities.* The entities are built by other entities, connected with *relationships*. The entities and relationships between them form the internal *structure* of the entity they build. To create the entity of a certain structural level of the world, it is necessary to have:

- the entities of the lower structural level;
- established forming relationship.

The entity can dialectically be considered as a relationship between its entities of all internal structural levels.

The forming relationship has a representative significance for the entity. The destruction of this essential relationship causes its disintegration. The establishment of forming relationship between already existing entities has a determine significance for the emerging of the new entity.

The forming relationship is the reason for *the emergence* of individual properties, which distinguish the new entity from the forming ones. ***The relationships form and present the entity***.

Impact, Interaction, Reflection

Building the relationship between the entities is a result of the ***contact*** among them. During the contact, one entity ***impacts*** on the other entity and vice versa. In some cases the opposite impact may not exist, but, in general, the contact may be considered as two mutually opposite impacts which occur in the same time.

The set of contacts between entities forms their ***interaction***. The interaction is a specific ***interactive relationship*** between entities which take part in it.

The contacts of the given structural level are processes of interaction of the entities on the lower levels.

During the establishing of the contact, the impact of an entity changes temporally or permanently the internal structure of the impacted entity. In other words, the realization of the relationships between entities changes, temporary or permanently, their internal structure at one or at few levels.

The internal change in the entity, which is due to impact of the other entity we denote with the notion "***direct reflection***".

Every entity has its own level of sensibility. This means that the internal changes occur when the external influence is over the boundary of the sensibility of the entity.

The ***"reflection impulse"*** for given entity is the amount of the external influence needed for transition from one state to the reflection one.

The entities of the world interact continuously. It is possible, after one interaction may be realized 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. In general, the concept “transitive impact” (respectively “transitive reflection”) of the first entity on the third entity through the second one will denote every chain of impacts (reflections) which start from first entity and ends in the third entity, and include the second entity in any internal place of the chain.

One special case is the ***external transitive self-reflection*** where the entity reflects its own relationships as a secondary reflection during any external interaction.

Some entities have an opportunity of ***internal self-reflection***. The internal self-reflection is possible only for very high levels of organization of the entities, i.e. for entities with very large and complicated structure. The self-reflection (self-change) of the entity leads to the creating of new relationships and corresponding entities in it.

Of course, the internal self-reflection is a result of the interaction provided between entities in the lower levels of the structure of the entity. Such kind of entities has relatively free sub-entities with own behavior in the frame of self-preservation of the whole entity. As a result of the self-reflection, some relationships and corresponding sub-entities are created or changed in the entity.

The combination of the internal and external self-reflection is possible.

Finally let remark that the reflection could not be detected by the entity that contains it. This is dialectical behavior of the reflection - it is only an internal change caused by the interaction.

Information

The real world contains unlimited number of entities. When an entity contacts another, there exists a 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 process of realization of the interaction between them — the process of realization of the contact is a specific (temporal) forming relationship between entities and during the process of establishing the contact the entities form new (temporal) entity which in the same moment may be reflected by the third entity. So, the third entity may reflect any vestiges of this interaction from both first and second entities.

In the special case when the third entity contains reflections of the first entity received by both two different ways:

1. by transitive impact of the first entity on the third one through the second entity,
2. by impact of the first entity on the third one which is different from the transitive one, i.e. it can be direct impact or transitive impact through another entity (-ies)

then the third entity became as an external relationship between first entity and its reflection in the second entity – it became as ***“reflection evidence”*** of this relationship.

- The first entity is called ***reflection source;*** the second entity is called ***reflection recipient***; and the third entity is called ***reflection evidence***.

In this special case, when there exist the triple

"(source, recipient: evidence)",

the reflection of the first entity in the second is called ***information*** in the second for the first entity.

Let point one very important case of the real world - simultaneous contacts of the three entities. Every one of them may be source, recipient and evidence in the same time. There exist six cases which represent the simultaneous contacts of three entities. Therefore, the entities A, B and C may be in the next six reflection relations: $(A, B: C)$; $(B, C: A)$; $(C, A: B)$; $(A, C: B)$; $(C, B: A)$; $(B, A: C)$.

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 $(A, B: C)$.

For practical needs, it is more convenient to follow the next consideration.

The reflection in the recipient represents both the relationships and the sub-entities 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, iff there exist reflection evidence than the reflection of the forming relationship may be considered as "information" for reflected entity. Therefore, in the sense that the evidence exists to point what relationship (between what entities) is reflected and where it is done, we may say

"***The information is reflected relationship".***

So, the *reflection* of the first entity in the second one is “***information”*** for the first entity if there is corresponded *reflection evidence*. The generalization of this idea leads to assertion that ***every reflection can be considered as information, iff there exists corresponding reflection evidence.***

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 for low levels. Therefore, the main idea is:

The General Structure of Information is a Space Hierarchy.

Information Elements and Information Memory

The triple

***i* = (source, recipient : evidence)**

defines concrete ***(single) information element.*** The triple "**i**" is called ***"information relationship".***

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; the entity memory may be more *temporal* or more *permanent.*

Information Spaces and Information Environment

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 the corresponded sets of information elements may exist.

The set of information elements, which is defined by:

- single source and single recipient, is called ***single information space***;
- many sources and single recipient, is called ***common information space***;
- single source and many recipients, is called ***single information environment,*** which contains many information spaces;
- many sources and many recipients, is called ***common information environment***.

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 of 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 corresponded types of information.

The types of information memories as well as the structures of the information environments define corresponded types of information, too.

Further Investigation in the Theory of Information

The further investigation and development of the Theory of Information may be directed towards investigation the types and characteristic of information in correspondence with the specific of entities and relationships as well as characteristics of the environment.

Contacts and Interactions need to be investigated according different types of entities.

The philosophical support is very important so the research need to take in account the "Theory of reverberation" [Pavlov, 1987] as well as the development and extending of ideas about reflection and self-reflection given in this paper.

The main place needs to take the investigation of the types and possible interconnections of the basic information triple ***i* = (source, recipient: evidence)** as well as the types and the characteristics of the direct, transitive, and interactive information and self-information based of the hypothesis about the general structure of information.

Special attention needs to be paid on the basic types of information in closely correspondence of type of interaction and reflection as well as the different levels of the structure of the entities.

As we have seen, the types and the characteristics of the information memories, the information spaces as well as the information environments are another main theme of this theory.

Theory of Infos

The genesis of the concept of **Infos** started from the understanding that the concept **"Information Subject"** is perceived as human characteristic. It is clear that in the nature there exist many creatures which may be classified to this category. To exclude the misunderstandings we decide to introduce new word to denote all possessors of the characteristics of the Information Subject.

This word is **"*Infos*"**.

Activity and Information Expectation

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 originated.

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:

1) all possibilities of the entities for establishing the contact are satisfied by such possibilities of other entities;
2) 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 whole entity or any its part may generate any possibilities too.

In the second case, the entity has any ***"free valences"***, 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***.

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 generated by the information relationship is called ***"information activity".***

On given level of complexity of the entities a new quality becomes — the existence of self-reflection and internal activity based on the main possibilities for contact of the sub-entities as well as based 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 sub-entities 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 sub-entities if they have no possibilities for integration with the others, i.e. if they have no free valences 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 valences.

One special kind of activity is the information one. We assume that the secondary activity needs to be resolved by relevant to the information valences corresponded opposite (information) valences which need to be of the same genesis, i.e. generated by any information relationship. So, not every entity may be used for resolving the secondary activity.

This way, the entity expects a special kind of (information) contacts and (information) interaction for resolving the information activity. Because of this the information activity is called ***"information expectation".***

Information Witness

Let remember the special case from above when the third entity contains reflections of the first entity received by both two different ways:

1) by transitive impact of the first entity on the third one through the second entity,
2) by impact of the first entity on the third one which is different from the transitive one, i.e. it can be direct impact or transitive impact through another entity (-ies).

In this case the third entity became as an external relationship between first entity and its reflection in the second entity — it became as ***"reflection evidence"*** of this relationship.

In addition, if during establishing the information relationship ***i*** = (source, recipient: evidence) in the reflection evidence is generated information expectation (activity) it is called ***"information witness".***

As the information witness is more complex entity so the information relationship may be more complex. In addition, let remark that the complex reflection is time-depended process. In other hand, the memory and actual context determine the result of the complex reflection.

Information is a Model

As Marx Wartofsky remarks, the concept "**model**" has been used for denotation of the very large class of phenomena: mechanical, theoretical, linguistic, etc. constructions. He gave a good definition of the model relation and made clear the main characteristics of the model [Wartofsky, 1979]. This definition is as follow:

The model relation is triple M:

$$M: (S, x, y)$$

where "S" is subject for whom "x" represents "y". In other words only in this relation and only for the subject "S" the entity "x" is a model of the entity "y".

As we point above, the interaction between two entities is a specific theirs relationship. If there exist information witness (**W**) of the interaction between two entities as well as of the existence of the information about the first

entity in the second entity, **W** became as subject for whom the information in the second entity represents the first one. In other words, there exists relation

$$M: (\mathbf{W}_{BA}, I_{BA}, A),$$

where "A" and "B" are entities, and the $\mathbf{W}_{BA}$ is the information witness, which proofs that the assertion "$I_{BA} \subset B$ is information in B for A" is true.

In the relation ($\mathbf{W}_{BA}$, I_{BA}, A) the information I_{BA} is a model of A*.*

Information Model

The entities of the world interact continuously in the time. It is possible, after any interaction one another may be realized. 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.

Let A, B and C are entities; A and B interact and after that B interacts with C. If there exist the relations:

- M_{BA}: ($\mathbf{W}_{BA}$, I_{BA}, A), where $\mathbf{W}_{BA}$ is the information witness, which proofs that the assertion "$I_{BA} \subset B$ is information in B for A" is true,
- M_{CB}: ($\mathbf{W}_{CB}$, I_{CB}, B), where $\mathbf{W}_{CB}$ is the information witness, which proofs that the assertion "$I_{CB} \subset C$ is information in C for B" is true,

and if $M_{C(B)A}$: ($\mathbf{W}_{C(B)A}$, $I_{C(B)A}$, A), where $\mathbf{W}_{C(B)A}$ is the information witness, which proofs that the assertion "$I_{C(B)A} \subset C$ is information in C for information in B for A" is true.

In such case, from point of view of the $\mathbf{W}_{C(B)A}$ the information $I_{C(B)A}$ is a model of A. In other hand, because of transitive reflection, $I_{C(B)A}$ is created as reflection of the I_{BA} but not directly of A.

This means that **$I_{C(B)A}$ is a model of the information in B for A**.

In other words the $I_{C(B)A}$ is an ***information model*** in C for A*.*

The collecting of information models for given entity in one resulting entity may exist as a result of the process of interaction between entities. Such process is in the base of the ***Information modeling***.

If an information model **IM** contains information for (reflected from) the two source information models $\mathbf{IM}_1$ and $\mathbf{IM}_2$ than the source information models are "***similar***" in the sense of the model **IM**.

The similarity of the information models causes the establishing the relation of aggregation between them.

The relation of similarity aggregates the similar models in new ***internally determined information model*** in the memory of the information witness.

The aggregation may cause the generating the new information activity, which may be resolved not only in the environment around the information witness. The possibility of self-reflection may cause the generating the new information models in his memory without any external influence and so on.

This process of aggregation and generation of new models is not limited.

The (information) models internally generated via self-reflection are called "***mental (information) models***" of the information witness.

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 valences with any others.

The combining the valences of the information expectation with some others is called ***resolving the information expectation***.

Let "n" is the number of free valences in an information expectation. After the contact some of them are combined as well as the others are not. The new valences, 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 normalized by "*n*" number *D'* of the not combined valences is called ***degree of discrepancy*** (**D**) of the incoming reflection to the information expectation, i.e.

$$\mathbf{D} = \frac{D'}{n}$$

The normalized by "*n*" number *C*' of the combined valences is called ***degree of combining*** (**C**) of the incoming reflection to the information expectation, i.e.

$$\mathbf{C} = \frac{C'}{n}$$

There exists the equation: **C + D = 1.**

From point of view of given expectation for contact the number of free valences is fixed. After the contact, as a result of reflection, some of the free valences of the entity may be combined with any new (internal or external) valences. Of course, new free valences may occur. The number "*n*" varies in the process of interaction. Every contact may change it.

The more valences of the information expectation have been resolved, the more qualitative is the incoming information and vice versa.

The difference **A** between normalized number **C** of resolved valences and normalized number **D** of not resolved valences of the information expectation is called ***adequacy of the reflection to the information expectation***, i.e.

$$\mathbf{A = C - D}$$

It is easy to see that the values of adequacy A are in the interval [-1,1].

Infos

The resolving of the information activity is ***a goal*** of the information witness.

This goal may be achieved by the establishment and providing (information) contacts and interaction.

The entity, which has possibility for:

- ***(primary) activity*** for external interaction;
- ***information reflection and information memory***, i.e. possibility for collecting the information;
- ***information self-reflection***, i.e. possibility for generating "secondary information";
- ***information expectation*** i.e. the (secondary) information activity for internal or external contact
- ***information modeling and resolving the information expectation***

is called ***Infos***.

Further Investigation in the Theory of Infos

What gives us the concept “Infos”?

At the fist place, this is the common approach for investigating the natural and artificial information agents.

In other hand, this is the set of common characteristics which are basic for all entities, which we may classify to the category of the Infos.

And, at the end, this is a common philosophical basis for understanding the information subjects.

Our main goal is to provoke the scientists to continue the research in this important area and to make the next step.

The concept **"Infos"** is basic for the General Information Theory [Markov et al. 2003a]. 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 needs to be investigated and classified in the future research. At the first step, we may propose that may be at least two main types of Infoses exist:

- ***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?
- Etc.

It is impossible to answer to all questions in a single article. We may make only the next little step. This is the aim of the present paper.

The concept "Information Model" (IM) is fundamental for the Informatics. There exist many approaches to define this concept. As a rule, every definition is based on those concepts, which the concrete scientific paradigm had given. Every new theoretical approach needs to redefine the concepts it uses in the frame of the corresponded to it new paradigm. This way in different paradigms we may have different definitions of the given concepts [Popper, 1968].

There exists a long list of names of scientists who worked to define more precisely the concept "Model" (and respectively - the information model). It contains the names of N.Wiener and A.Rosenblueth [Rosenblueth, Wiener, 1945], V.M.Glushkov [Glushkov, 1986], M.W.Wartofsky [Wartofsky, 1979] and many others.

For long period, the concept "Information model" has been used to denote one of the main activities in using the computer technique. May be, now it is the most popular understanding of it and many scientists are satisfied of the meaning it contains.

Nevertheless, the definition of the concept of information model may and need to be extended to cover the new scientific paradigms, which come from the current Informatics. This is the goal of the Theory of the Infos.

Presented above so simple and clear definition of the concept "information model" has very great impact on GIT and key role for definition and explanation of all subjective information phenomena in the world. In addition, it may be used as a base concept in the area of Artificial Intelligence research.

The information models initiated inside the Infos form subjective information set. Inside his information set, the Infos can build "information spaces". The information space of the Infos is dynamic as a structure and content. When a new information model is generated the Infos compares it with the information models from context and with the information expectation. This is the starting point for the processes of reasoning.

However, the information modeling is only the first part of the complex process of decision making in usual practical situations. The decision making is at least two-level process:

- Collecting information models for given entity in one resulting entity;
- Analyzing and knowledge discovery on the base of given goal, which results aimed to be used for predicting of any characteristics of the modeled entity.

In everyday language, the concept "Information modeling" is assumed to denote the whole chain of phases of decision making, which we make to solve the problem [Gladun, 1994].

Theory of Inforaction

Information Objects

When the Infos interact with the entities around in the environment, there exist at least two cases of reverberation:

- the contacts and interaction are casual and all reflections in the Infos as well as in the entities have casual origin;
- the contacts and interactions are determined by the information activity of the Infos.

In the both cases, the contacted entity may reflect any information model from Infos. The possibility for reflection of the information model is greater in the second case.

An entity, in which one or more information models are reflected, is called ***"information object".***

The information objects can have different properties depending on:

- the kind of influence over the entities - by ordering in space and time, by partial or full modifying, etc.,
- the way of influence over the entities - by direct or by indirect influence of the Infos on the object,
- the way of development in time - static or dynamic,

etc.

It clear, the Infoses are information objects.

Information Operations

The information is kind of reflection. Therefore, the only way one to operate with information is to operate with the entity it contains. Every influence on the entity may cause any internal changes in it and this way may change the information already reflected. Another type of influence is to change the location of entity or to provoke any contact between given entity and any other.

The influence over the information object is called "***information operation***".

The information operations may be of two main types:

- the Infos internal operations with the sub-entities that contain information,
- external operations with the information objects that contain information.

Internal Operations

The internal operations with the subentities closely depend of the Infos possibilities for self-reflection and internal interaction of its subentities.

The self-reflection (self-change) of the Infos leads to the creating of new relationships (and corresponding entities) in it. These are subjectively defined relationships, or shortly - ***subjective relationships***. When they are reflected in the memory of the Infos they initiate information model too, but on a higher level. These high-level information models may have not real relationships and real entities that correspond to them.

The possibility for creating the relationships of similarity is a basis for realizing such very high level operations as "comparing elements or substructures of the information models", "searching given substructure or element pattern in the part or in the whole structure of the information model", etc.

It is clear, the Infos is built by entities some of which may be also Infoses, but on the lowest levels. So, the internal operations are determined by the concrete internal level but from the point of view of the higher levels, they are assumed as external operations. Because of this, we will concentrate out attention on the second type of operations.

External Operations

The external operations with information objects may be differed in two main subtypes — basic and service operations.

There are two "***basic information operations***" which are called I-operations:

- I-reflection (reflecting the information object by the Infos, i.e. the origination of a relevant information model in the memory of the Infos).
- I-realisation (creating the information object by the Infos);

In the process of its activity, the Infos reflects (perceives) information from the environment (entities O_i, i=1,2...) by proper subentities (sensitive to video, acoustic, tactile, etc. influences) called "***receptors***" R_i (i=1,2...). Consequently, the Infos may receive some information models. This subjective reflection is called "***I-reflection***".

When necessary, the Infos can materialize (realize) in its environment (entities O'_j, i=1,2...) some of the information models, which are in his memory, using some sub-entities called "***effectors***" M_j (j=1,2...). Consequently, new or modified already existing entities reflect information, relevant to these information models. This subjective realization is called ***"I-realization"***.

There are several operations, which can be realized with the information objects: transfer in space and time, destroying, copying, composition, decomposition, etc. Because of the activity of the Infoses, these operations are different from other events in reality. In this case, the Infos determined operations with information objects are called "***service information operations***".

For example, some of the very high-level service operations are based on the external influence on the information object to change any existing reflection: including and removing an element in and from the structure; copying or moving substructures from one place to an other; building new structure using as a basis one or several others; composing or decomposing of elements or substructures; etc.

Information Processes

Let "O" is a set of real information objects and "M" is a set of information models.

We will consider every set of real information objects as an information object, if the opposite is not stated.

Every set of information models we consider as information model.

The *information operations* are:

- the function **r**: $O \rightarrow M$. (*I-reflection)*

- the function **e**: $M \rightarrow O$. (*I-realization)*

- the function **s**: $O_d \rightarrow O_r$ between two sets of information objects, O_d and O_r may be coincidental (*service operation*).

Let t_1, t_2,..,t_n are information operations. The consequence of information operations P, created using the composition, i.e.

$$P = t_1 \circ t_2 \circ \ldots \circ t_n$$

is called "***information process***".

In particularly an information process can include only one operation.

It is clear, the composition of two or more I-reflections as well as the composition of two or more I-realizations are not allowed.

Information Contact

If an information model from the Infos is reflected in another entity, there exist possibility, during the "a posterior" interactions of the given entity with another Infos, to transfer this reflection in it. This way an information model may be transferred from the Infos to another.

If the second Infos has already established information expectation, the incoming reflection will be perceptible for the Infos. The information expectation will be resolved in some degree and the incoming information model and information in it will be received by the second Infos.

Let S_1 and S_2 are Infoses and O is an arbitrary entity.

The composition of two contacts

$$S_1 \xrightarrow{\Theta^{S_1 O}} O \xrightarrow{\Theta^{O S_2}} S_2$$

is called "***information contact***" between Infos S_1 and Infos S_2 iff during the contacts any information model from S_1 is reflected in the Infos S_2 true the entity O. The Infos S_1 is called "***information donor***", the Infos S_2 is called "***information recipient***", and the entity O is called "***information object***".

In this case, when the donor and the recipient are different Infoses the information contacts between them consist of a composition of at least two information operations - I-realization and I-reflection. For the realization of a direct information contact between two different Infoses is necessary the execution of the composition of these two "basic" operations. All the rest information operations are necessary for supporting the basic ones i.e. they are auxiliary (service) operations.

For the realization of one information contact at least one information object is necessary.

This way the elementary communicative action will be provided.

In general, every information process "k", having as a start domain the set S_d (of information models) and as a final domain the set S_r (again of information models), which may be coincidental, we call "information contact":

$$k: S_d \rightarrow S_r$$

S_d is called "Infos-donor" and S_r - "Infos-recipient".

Information Interaction

The set "R" of all information contacts between two Infoses S_a and S_b

$$R= \{k_i \mid i=1,2..; k_i: S_a \rightarrow S_b\}$$

is called "***information interaction***" or simply "***inforaction***".

When S_a and S_b are coincident, we call it Information interaction with itself (in space and time).

The set "B" of all information objects, used in the information interaction between given Infoses is called "***information base***".

Information Society

The **"Information Group"** (IG) is a set of Infoses, with common Information Base of the information interactions between them.

The **"Information Society"** (IS) is a set of Information Groups, with common Information Base of the information interactions between them.

In the small Information Group the service information operations may be provided by the every Infos when it is necessary.

In the Information Society this is impossible or not optimal. In such case, some Infoses or Information Group became as **"information mediators"** between the others. They start to provide the service information operations.

They realize **"Information Service"**.

Further Investigation in the Theory of Inforaction

For more than twenty years the Theory of Inforaction has traversed the way from the exotic and unusual concepts such as "information contact" and "information object", presented by the authors in 1983, to actual and world-wide investigated area of informatics.

Nevertheless, there exist many problems for future research.

Note that I-realization is not just reflections of information models in material entity. They include both a reflection of the information models of the Infos and a reflection of the state of the Infos in the moment of I-realization.

This means that here we consider the notion of "information objects" as a more general than the notion of "message".

It is possible that the entity of the information object is not able to keep (save) the whole influence of the I-realization. In other hand, the Infos consciously, by proper actions, restricts the I-realization to reflection of information model only by suppressing the reflection of his condition in the moment of I-realization.

In this case, we are near to the notion of "message" as we use it conventionally.

For example, from the point of view of the notion message, the speech of one politician on the meeting and the same speech printed in the paper are the two equal variants of the message. However, the influence and the result from the perceiving of the speech are different in both cases. In the first one (direct contact) the perceiving one can reflect the condition of the speaker (intonation, pauses, etc.) but in the second case (indirect contact) this is almost impossible. From this point of view, there exists a relation between two different information objects.

The Theory of Inforaction closely depends on the results of information operations. After the execution of some of the information operations, a new information object is possible to be created (for example, after the composition or decomposition). In other cases, the operation may not lead to appearance of new information object but to destroying of a certain existing information object.

The Infos is the only one who can determine whether after the execution of one operation (or a consequence of operations) an information object has appeared. Analogously, the Infos is the witness whether a new information object appears, when in the process of I-realization the Infos acts upon entities, which include some reflection of earlier I-realization. That is why, when there is not exact instruction from the Infos, we suppose that all information operations, with the exception of two - "destroying" and "I-reflection" will produce (one or more) information objects. The operation destroying initiates "empty" information object by destroying the starting one. We suppose that the operation "I-reflection" always initiates information model in the memory of the Infos.

Because of the growing of the communicative aspects of the information service now all over the world the everyday concept is the "Information society".

The growth of the global information society shows that the knowledge turns into important and necessary article of trade. The open environment and the market attitudes of the society lead to arising of the knowledge customers and knowledge sellers, which step-by-step form the "Knowledge Markets". As the other markets,

the Knowledge Market is the organized aggregate of participants, which operates in the environment of common rules and principles.

Examination of the market demand for various types of courses and training modules is an essential criterion for effectiveness and high efficiency. Market trends, industry requirements, and companies training needs have to be examined on a regular basis in accordance with the Theory of Information Interaction.

Conclusion

The development of the General Information Theory should not become by the single creative impulse. For a long period, the constructive activity of the many researchers is needed for establishing the new common paradigm.

We all need free scientific look at things, which will permit us to build the general theory without partiality, and aberrations taking in account all information paradigms already created and adopted.

During the years, the investigation in the area of the GIT has showed that the received theoretical results may be used for building the ontology of informatics. Our opinion is that the GIT may be used as main classification scheme. The first step is to describe the main divisions of informatics. The further investigation needs integration with other scientific areas and paradigms.

We have made a little walk toward the establishing the new paradigm. It is synthesized in the table below.

Basic Structure of the General Information Theory

Occurrence	Specificity	Subject	Theory
Reflection	Information Relationship	Evidence	Theory of Information
Activity	Information Expectation	Witness	Theory of Infos
Modeling	Information Modeling	Infos	
Interaction	Information Interaction	Society	Theory of Inforaction

Acknowledgements

This paper is the next step of the process of establishing the GIT as common paradigm. It 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 organized by the International Workgroup on Data Base Intellectualization (IWDBI). The creative discussion at the KDS 2003 International conference, based on the [Markov et al, 2003a] gives us a great impulse to continue working in this very important scientific area. Authors are very grateful to all participants in the fruitful discussions at KDS and ITA International Conferences and to all members of the International Workgroup on Data Base Intellectualization (IWDBI) and the Association of Developers and Users of Intellectualized Systems (ADUIS) for supporting the advance of the General Information Theory.

This work is partially financed by project **ITHEA-XXI** of FOI Institute of Information Theories and Applications.

Bibliography

[Abdeev,1994] R.F.Abdeev. The Philosophy of the Information Civilization. Moscow, VLADOS, 1994. (in Russian)

[Bangov, 1995] I.Bangov. A Graph-Topological Model of Information. Int. J. Information Theories and Applications, 1995, v.3, n.6, pp.3-9.

[Burgin, 1997] M.S.Burgin. General Information Theory. http://www.math.ucla.edu/~mburgin/res/compsc/Site3GTI.htm

[Burgin, Markov, 1991] M.Burgin, Kr.Markov. Formal Definition of the Concept Materialization. Mathematics and Mathematical Education, BAS, Sofia, 1991, pp.175-179.

[Cooman et al, 1995] G.de Cooman, D.Ruan, E.Kerre, Eds. Foundations and Applications of Possibility Theory. World Scientific, Singapore, 1995.

[Elstner, 1993] D.Elstner. About Duality of the Information and Organization. Int. J. Information Theories and Applications, 1993, v.1, n.1, pp. 3-5. (in Russian)

[Gladun, 1994] V.P. Gladun. Processes of New Knowledge Formation. Pedagog 6, Sofia, 1994 (in Russian).

[Glushkov, 1986] V.M.Glushkov. Epistemological Nature of the Information Modeling. In: V.M.Glushkov. Cybernetics, Questions and Answers, (in Russian), Moscow, Science, 1986, pp. 33-41.

[Lu, 1999] C.-G.Lu. A Generalisation of Shannon's Information Theory. Int. J. of General Systems, 28:(6), 1999, pp.453-490.

[Markov, 1984] Kr. Markov. A Multi-domain Access Method. Proc. of Int. Conf. "Computer Based Scientific Research". Plovdiv, 1984. pp. 558-563.

[Markov, 1988] Kr.Markov. From the Past to the Future of 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

[Markov et al, 2003a] Kr.Markov, Kr.Ivanova, I.Mitov. General Information Theory. Basic Formulations. FOI-Commerce, Sofia, 2003.

[Markov et al, 2003b] K. Markov, K. Ivanova, I. Mitov, E. Velikova-Bandova. *The Information*. IJ Information Theories and Applications, FOI ITHEA, Sofia, 2003, Vol.10, No.1, pp.5-9.

[Markov et al, 2004] K. Markov, K. Ivanova, I. Mitov, E. Velikova-Bandova. *Formal Definition of the Concept "INFOS"*. Proceedings of the Second International Conference "Information Research, Applications and Education" i.TECH 2004, Varna, Bulgaria. Sofia, FOI-Commerce, 2004, pp. 71-74. Int. Journal "Information Theories and Applications", 2004, Vol.11, No.1, pp.16-19

[Markov P., 2002] P.Markov. Think with Your Mind. Markov College. Sofia, 2002. (in Bulgarian)

[Pavlov, 1987] T.Pavlov. Collection of Selected Works, Vol. II. Theory of Reverberation. Science and Art, Sofia, 1987. (in Bulgarian).

[Rosenblueth, Wiener, 1945] A.Rosenblueth, N.Wiener. Role of Models in Science. Philosophy of Science, Vol.12, No.4, 1945.

[Shannon, 1949] C.E.Shannon. The Mathematical Theory of Communication. In: The Mathematical Theory of Communication. Ed. C.E.Shannon and W.Weaver. University of Illinois Press, Urbana, 1949.

[Tomov, 1991] K.Tomov. The Resomal-Isomorphic Principle. Arges, Sofia, 1991. (in Bulgarian)

[Wartofsky, 1979] M.W.Wartofsky. Models. Representation and the Scientific Understanding. D.Reidel Publishing Company, Dordrecht: Holland /Boston: USA, London: England/, 1979.

[Popper, 1968] K.R.Popper. Conjectures and Refutations: The Growth of Scientific Knowledge. Harper & Row, Publishers, New York and Evanston. 1968.

Authors' Information

Krassimir Markov *- Institute of Mathematics and Informatics, BAS, Acad.G.Bonthev St., bl.8, Sofia-1113, Bulgaria; Institute of Information Theories and Applications FOI ITHEA, 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*

ПРОБЛЕМЫ АТРИБУТИВНОГО АНАЛИЗА ДИНАМИЧЕСКИХ ОБЪЕКТОВ, ПРЕДСТАВЛЕННЫХ ВРЕМЕННЫМИ РЯДАМИ

Александр Андреев, Виталий Величко, Виктор Гладун, Юрий Иваськив, Сергей Чеботарь

***Abstract**: Problems of the analysis of the dynamic objects represented by time series on the basis of growing pyramidal networks are investigated. The way of representation of dynamic objects in pyramidal networks is proposed. The problem of recognition of operation modes of the objects, represented by time series is considered.*

***Keywords**: Attributive analysis, dynamic objects, time series, pyramidal networks.*

Введение

Атрибутивный анализ предусматривает представление объектов и ситуаций признаковыми (атрибутивными) описаниями. Анализ, основанный на использовании признаковых описаний, применяется при решении задач в средах, образованных дискретными объектами. В отличие от непрерывных сред, решение задач в дискретных средах связано с обработкой не количественной, а качественной, смысловой информации и основывается на применении логико-лингвистических моделей [Поспелов,1981], [Закревский, 1988]. При этом процессы решения задач оказываются процессами обработки атрибутивных описаний.

Необходимость в проведении атрибутивного анализа возникает при решении различных классов задач, относящихся к сфере техники, экономики, медицины и связанных с обнаружением закономерностей, классификацией, диагностикой, прогнозированием состояний и поведения различных объектов и процессов, а также с созданием интеллектуальных систем поддержки принятия решений.

Известные методы атрибутивного анализа основываются преимущественно на использовании признаковых описаний, характеризующих состояние статических объектов независимо от времени. Для статических объектов предложены методы и средства атрибутивного анализа, ориентированные на различные свойства и характеристики данных, описывающих анализируемые объекты. Разработана и нашла широкое применение система *CONFOR* [Святогор, 2005], основанная на применении сетевых структур специального типа – растущих пирамидальных сетей [Gladun, 2003], [Гладун, 2002], [Гладун, 2000]. Растущей пирамидальной сетью (РПС) называют ациклический ориентированный граф, в котором нет вершин, имеющих только одну заходящую дугу (пример приведен на рис.1). Вершины, не имеющие заходящих дуг, названы рецепторами, все другие вершины – концепторами. Подграф пирамидальной сети, включающий некоторую вершину ***a*** и все вершины, от которых имеются пути к этой вершине, называются пирамидой вершины ***a***. Наличие элементов, образующих подграфы типа «пирамиды», сочетается с иерархической организацией структуры рассматриваемых сетей, при которой вершины - рецепторы образуют нулевой уровень иерархии.

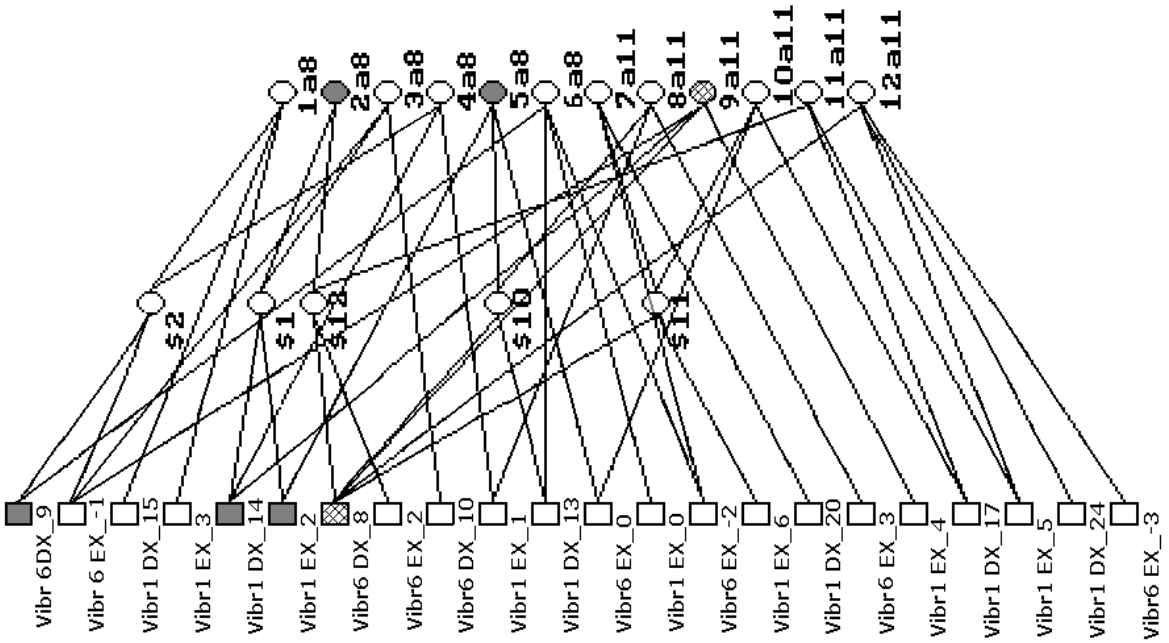


Рис. 1

В пирамидальной сети входной информацией служат наборы значений признаков, описывающих объекты некоторого множества (обучающей выборки). В качестве таких значений в различных задачах могут рассматриваться имена свойств, отношений, состояний, действий, объектов и т.п. Значения признаков поступают на рецепторы сети. Концепторы разных уровней иерархии представляют конъюнкции значений признаков, представляющих описания отдельных объектов либо пересечения таких описаний.

Пирамидальные сети обладают преимуществами при реализации различных операций ассоциативного поиска. Например, для выбора всех объектов, включающих заданное сочетание значений признаков, необходимо проследить пути вверх от соответствующей вершины до вершин верхнего уровня, соответствующих объектам. Выбор всех объектов, описание которых пересекается с описанием заданного объекта, осуществляется просмотром путей, исходящих из вершин, образующих его пирамиду.

Алгоритм построения РПС [Гладун, 1987], [Gladun, 2003] обеспечивает автоматическое установление ассоциативной близости между объектами по общим элементам их описаний. Для РПС определен алгоритм формирования обобщенных логических определений классов объектов – понятий [Гладун, 1987], [Гладун, 2000], [Гладун, 2004]. В сети понятия представлены ансамблями вершин, в наибольшей степени характеризующих классы объектов, соответствующих понятиям. В результате обеспечивается возможность использования понятий в задачах классификации, диагностики, прогнозирования.

Преимущество РПС по сравнению с другими известными системами атрибутивного анализа состоит в сравнительно высокой точности решения задач обнаружения закономерностей, классификации, диагностики, прогнозирования. Однако отсутствие в РПС возможностей непосредственного учета временных параметров, не позволяет использовать их для анализа динамических объектов. В настоящей работе решение проблемы атрибутивного анализа объектов такого типа основывается на использовании системы *CONFOR* и представлении данных временными рядами, сформированными в результате наблюдений объекта исследований в различные моменты времени.

Представление динамических объектов в РПС

Применительно к динамическим объектам предполагается, что изменение их состояния во времени является вероятностным (стохастическим) процессом. В реальных объектах могут действовать неизвестные или малоизученные факторы, имеющие случайный характер. В результате точное вычисление будущего состояния объекта в различные моменты времени не представляется возможным. Однако может быть вычислена вероятность того, что его будущее состояние определяется некоторым интервалом значений известных переменных. В этом случае мгновенное состояние объекта рассматривается как точка некоторого пространства состояний ***R***, а стохастический процесс представляется функцией времени ***t*** со значениями из ***R***. Точки выбранного пространства задаются одним или несколькими числовыми параметрами, случайно принимающими различные значения. В настоящей работе временной ряд рассматривается как случайный процесс с дискретным временем, когда ***t*** принимает только целочисленные значения.

Известные способы анализа временных рядов основываются преимущественно на использовании численных методов [Бокс, 1972]. Техника обработки данных на основе растущих пирамидальных сетей связана с применением логических моделей, в которых используются различные логические функции: И, ИЛИ, НЕ.

Рассмотрим принципы преобразования исходных данных, представленных точками временного ряда к виду, пригодному для обработки с помощью РПС.

Пусть имеется некоторый динамический объект, информация о состоянии которого считывается некоторыми датчиками. Выход каждого из датчиков представляет замеры амплитуд сигналов, характеризующих состояния соответствующего участка исследуемого объекта и являющихся значениями временного ряда.

Каждый временной ряд, содержащий ***S*** замеров, разделим на ***Z*** равных отрезков, каждый из которых содержит S/Z замеров. Эти информационные отрезки рассматриваются как некоторые объекты, из описаний которых предполагается сформировать обучающую и экзаменационную выборки для РПС. Выделим признаки, с помощью которых будем описывать полученные объекты.

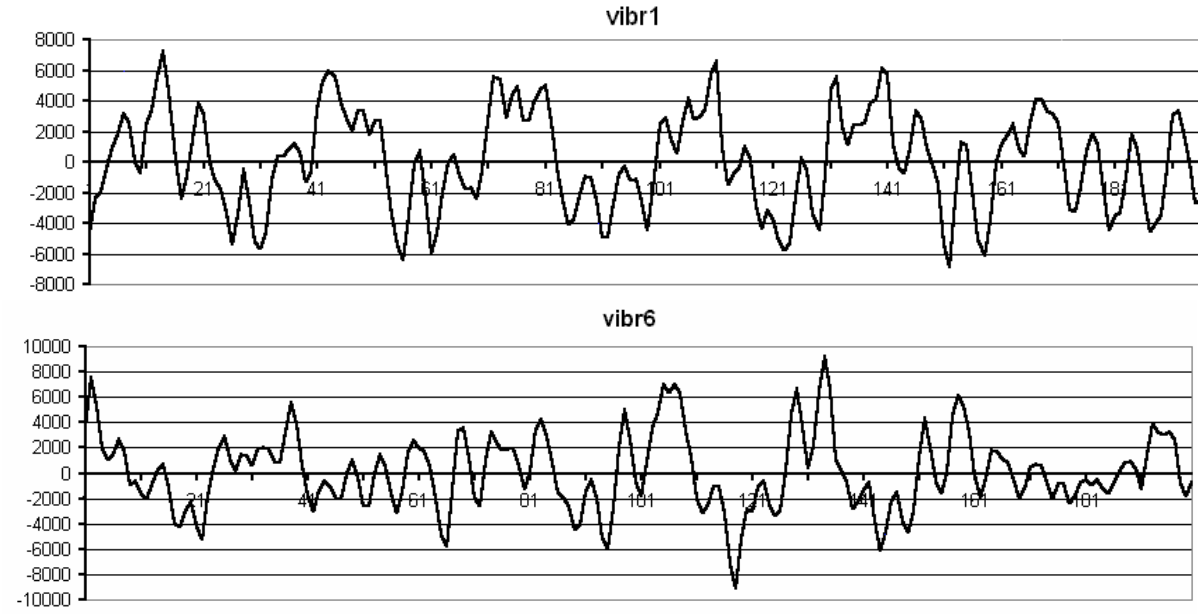
В качестве характеристик таких объектов могут быть использованы математическое ожидание (среднее значение) ***EX*** случайной величины ***X*** и дисперсия – мера ***DX*** отклонения случайной величины ***X*** от ее математического ожидания, определяемая равенством $DX = E(X - EX)^2$. Среднее значение и дисперсию, которые могут быть вычислены с помощью табличного процессора, например, MS Excel, предлагается использовать в качестве признаков для описания динамических объектов, представленных временными рядами.

Разделим ***Z*** объектов на ***V*** – обучающую выборку и ***W*** – экзаменационную выборку, $V + W = Z$. Таблица данных, содержащая сформированную обучающую выборку, служит основой для построения РПС и последующего ее обучения, а так же для формирования понятий, на основе которых будет выполнена классификация объектов экзаменационной выборки.

Задача распознавания режимов работы динамических объектов, представленных временными рядами

Рассмотрим построение таблицы данных и соответствующей РПС применительно к задаче распознавания режимов работы некоторого динамического объекта. В качестве примера такого объекта, в частности, может быть выбран вал двигателя летательного аппарата. Рассматривается два режима работы объекта – с нагрузками, условно обозначенными **a8** и **a11**. Каждый режим представлен замерами амплитуд сигналов на выходе двух вибродатчиков (**vibr1**,**vibr6**), установленных в различных местах конструкции летательного аппарата. Для каждого временного ряда ***S*** = 48000, ***Z*** = 24, ***V*** = ***W*** = 12. Фрагменты графиков огибающих кривых замеров амплитуд для нагрузок **a8** и **a11** показаны на Рис. 2 а,б соответственно.

Нагрузка **a8**



Нагрузка **a11**

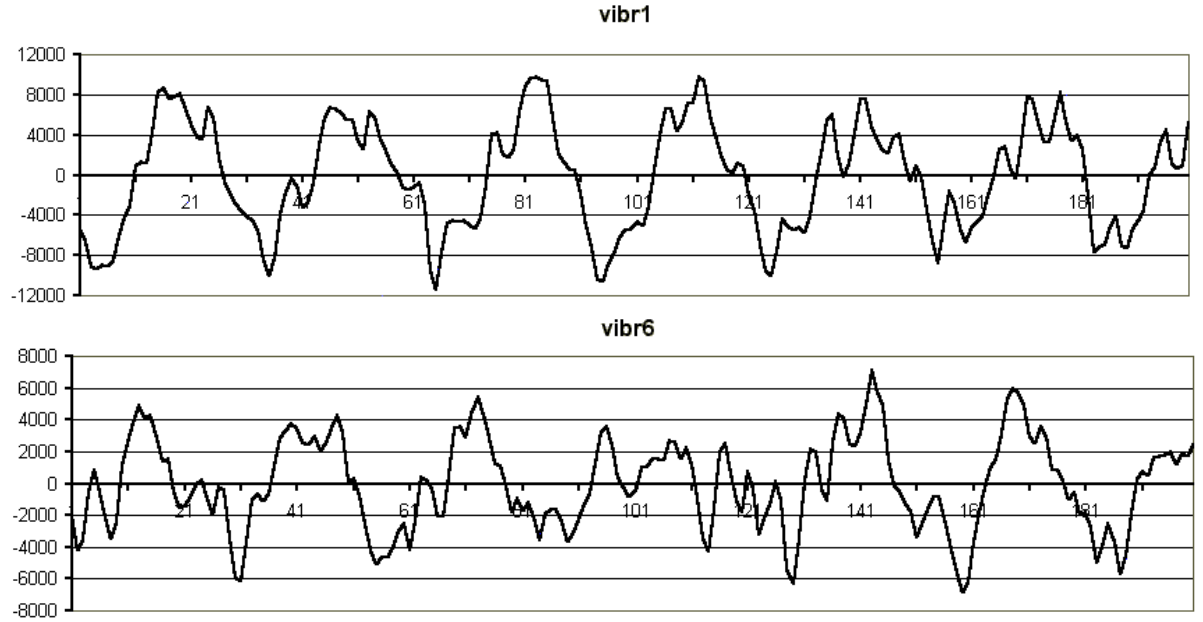


Рис. 2 а,б.

Расчетные данные, полученные применительно к каждому из режимов, для временных рядов представлены в таблице 1, в которой объекты 1-12 образуют обучающую выборку, а объекты 13-24 – экзаменационную. Подсчитанные значения дисперсии и среднего значения округлены до 2-х старших разрядов чисел.

Эксперименты проводились на основе программной системы CONFOR [Гладун, 2002], реализующей процессы построения и обучения РПС. Сеть, построенная по данным таблицы 1 в соответствии с правилами, описанными в работе [Гладун, 1987], [Gladun, 2001], [Gladun, 2003], показана на рис. 1. Обучение сети выполнено по обучающей выборке, представленной в таблице 1, в соответствии с правилами формирования в РПС понятий [Гладун, 1987], [Gladun, 2001], [Гладун, 2000]. В результате получены логические выражения (понятия), определяющие исследуемые классы объектов в режимах, соответствующих нагрузкам **a8** и **a11**.

Таблица 1

Тип выборки	Объект	Класс (нагрузка)	Vibr 1 *DX*	Vibr 1 *EX*	Vibr 6 *DX*	Vibr 6 *EX*
Обучающая	1	a8	15	3	9	-1
	2	a8	14	2	8	2
	3	a8	14	2	10	-1
	4	a8	14	1	9	-1
	5	a8	13	2	8	0
	6	a8	13	0	9	-2
	7	a11	20	6	8	-2
	8	a11	13	1	8	3
	9	a11	14	4	8	-1
	10	a11	17	6	8	0
	11	a11	17	5	8	2
	12	a11	24	5	8	-3
Экзаменационная	13	a8	15	3	9	1
	14	a8	15	4	9	-2
	15	a8	14	1	9	1
	16	a8	13	3	9	-1
	17	a8	12	1	9	2
	18	a8	13	5	8	0
	19	a11	16	2	9	1
	20	a11	16	6	8	-1
	21	a11	17	3	8	1
	22	a11	19	3	8	0
	23	a11	20	4	8	0
	24	a11	16	3	9	-2

После формирования таблицы признаков (табл.1) последовательно выполняются этапы связанные с построением и обучением РПС, а именно:

1. Построение РПС по описаниям объектов, представленных в таблице 1. В результате возникает сеть, приведенная на рис. 1.
2. Обучение РПС. В результате в построенной сети выделяются контрольные вершины понятий, определяющие в совокупности понятия, соответствующие нагрузкам **a8** и **a11**. На рис. 1 вершины, соответствующие контрольным вершинам построенных понятий, выделены темным цветом (нагрузка **a8**) или штриховкой (нагрузка **a11**).
3. Описание выделенных понятий логическими выражениями. В результате строятся логические выражения, описывающие выделенные классы.

Понятие, соответствующее нагрузке **a8**

Vibr 1 EX_2 ∨ Vibr 1 DX_14 & ~{Vibr 1 EX_4 & Vibr 6 DX_8 & Vibr 6 EX_-1} ∨ Vibr 6 DX_9 ∨
∨ Vibr 1 DX_13 & Vibr 1 EX_2 & Vibr 6 DX_8 & Vibr 6 EX_0 ∨
∨ Vibr 6 DX_8 & Vibr 6 EX_2 & Vibr 1 DX_14 & Vibr 1 EX_2

Понятие, соответствующее нагрузке **a11**

Vibr 6 DX_8 & ~{Vibr 6 EX_2 & Vibr 1 DX_14 & Vibr 1 EX_2} & ~{Vibr 1 DX_13 & Vibr 1 EX_2 & Vibr 6 EX_0}

V Vibr 1 DX_14 & Vibr 1 EX_4 & Vibr 6 DX_8 & Vibr 6 EX_-1

На основе сформированных понятий выполнена классификация объектов экзаменационной выборки (табл. 1). Классификация объектов выполняется путем вычисления значений логических выражений, определяющих понятие соответствующего класса, после подстановки 1 или 0 из описаний классифицируемых объектов на места значений признаков в логических выражениях.

Результаты классификации приведены в табл. 2 и показывают, что распознавание объектов экзаменационной выборки производится с достаточно высокой точностью.

Таблица 2

Объект	Класс (нагрузка)	Результат классификации
13	**a8**	**a8**
14	**a8**	**a8**
15	**a8**	**a8**
16	**a8**	**a8**
17	**a8**	**a8**
18	**a8**	**a11**
19	**a11**	**a8**
20	**a11**	**a11**
21	**a11**	**a11**
22	**a11**	**a11**
23	**a11**	**a11**
24	**a11**	**a8**

Оценка результатов экзамена

всего объектов - 12

правильно опознанных – 9 [75 %]

неправильно опознанных – 3 [25 %]

неопределенных – 0 [0,00 %]

Логические выражения, определяющие различные классы объектов и найденные с помощью РПС, объединяются в *кластерные базы данных* (КБД). КБД содержит информацию о группах объектов (кластерах), специфичных для исследуемой предметной области. На основе КБД решаются задачи классификации, диагностики и прогнозирования. После того, как понятие для некоторого класса объектов сформировано, проблемы прогнозирования и диагностики сводятся к проблеме классификации.

Заключение

В целом результаты проведенного эксперимента дают основание полагать, что РПС могут использоваться в качестве эффективного инструмента классификации временных рядов. При этом удовлетворительные результаты получаются при использовании для описания временных рядов их статистических характеристик. Важную роль играет присущее РПС достоинство, состоящее в генерации и использовании различных сочетаний значений признаков, представленных концепторами РПС. Путем сведения к классификации можно успешно решать другие аналитические задачи, связанные с временными рядами, например, такие, как диагностика свойств и прогнозирование событий во временных рядах.

Литература

[Поспелов,1981] Поспелов Д.А. Логико-лингвистические модели в системах управления. Москва: Энергоиздат.- 1981.

[Закревский, 1988] Закревский А.Д, Логика распознавания. - Минск, 1988. - 117с.

[Гладун, 2000] Гладун В.П. Партнерство с компьютером. - Киев: "Port-Royal", 2000. - 128с.

[Гладун, 1987] Гладун В.П. Планирование решений. Наукова думка, Киев, 1987. - 186с.

[Гладун, 2004] Гладун В.П. Растущие пирамидальные сети // Новости искусственного интеллекта.- 2004. - N. с. 30-40.

[Gladun, 2003] Gladun V., Intelligent systems memory structuring. // Proceedings of the X-th International Conference "Knowledge-Dialogue-Solution"(KDS'2003).- Varna, Bulgaria.-2003.- pp.16-20.

[Gladun, 2001] Gladun V., Vashchenko N. Analitical Processes in Pyramidal Network. // Information Theories and Application, Sofia: FOI-COMMERCE, 2001.

[Гладун, 2002] Гладун В.П., Ващенко Н.Д., Величко В.Ю. Прогнозирование на основе растущих пирамидальных сетей // Программные продукты и системы.-2002.-№2.- с.22-26.

[Святогор, 2005] Святогор Л.А. К вопросу о развитии интерфейса «разработчик-заказчик». // Proceedings of the XI-th International Conference "Knowledge-Dialogue-Solution"(KDS'2005).- Varna, Bulgaria.-2005.- pp.371-374 vol.2.

[Бокс, 1972] Бокс Дж., Дженкинс Л. Анализ временных рядов (в 2-х томах). – Москва: Мир, 1972. – 456с.

Информация об авторах

Александр Андреев *– Национальный авиационный университет. Институт компьютерных технологий. Киев-58, 03058, просп. Космонавта Комарова, 1*

Виталий Юрьевич Величко *– Ин-т кибернетики им. В.М. Глушкова НАН Украины, Киев-187 ГСП, 03680, просп. акад. Глушкова, 40, e-mail: glad@aduis.kiev.ua*

Виктор Поликарпович Гладун *– Ин-т кибернетики им. В.М. Глушкова НАН Украины, Киев-187 ГСП, 03680, просп. акад. Глушкова, 40,e-mail: glad@aduis.kiev.ua*

Юрий Лукич Иваськив *– Национальный авиационный университет. Институт компьютерных технологий. Киев-58, 03058, просп. Космонавта Комарова, 1*

Сергей Сергеевич Чеботарь *– Национальный авиационный университет. Институт компьютерных технологий. Киев-58, 03058, просп. Космонавта Комарова, 1*

СТРУКТУРНЫЙ АНАЛИЗ КОНТУРОВ КАК ПОСЛЕДОВАТЕЛЬНОСТЕЙ ОТРЕЗКОВ ЦИФРОВЫХ ПРЯМЫХ И ДУГ ЦИФРОВЫХ КРИВЫХ

Владимир Калмыков

Аннотация: *Рассматриваются вопросы сегментации контуров изображений на отрезки цифровых прямых и дуги цифровых кривых. Предлагаются определения отрезков цифровых прямых и дуг цифровых кривых, методы и алгоритмы их выделения в последовательности элементов контура. Алгоритмы сформулированы в терминах растущих пирамидальных сетей с учетом концептуальной модели памяти и опознавания (Рабинович [4]).*

Ключевые слова*: контур, отрезки цифровых прямых, дуги цифровых кривых*

Введение

Структурный анализ контуров изображений как последовательностей отрезков прямых и дуг кривых является одной из задач обработки изображений с целью их интерпретации в системах искусственного интеллекта.

В большинстве случаев изображение можно рассматривать как часть плоскости, разделенную на области с постоянными или меняющимися по некоторому закону параметрами, например, оптической плотностью, цветом, текстурой. Граница, то есть контур, является неотъемлемым свойством каждой из этих областей и представляет собой односвязную последовательность, состоящую из отрезков прямых и дуг кривых. Изображение, как правило, рассматривается в дискретном виде. Соответственно, отрезки прямых и дуги кривых, образующие контуры изображений, являются *отрезками цифровых прямых* и *дугами цифровых кривых*.

Сегментация произвольного контура на отрезки цифровых прямых и/или дуги цифровых кривых является целью настоящей работы.

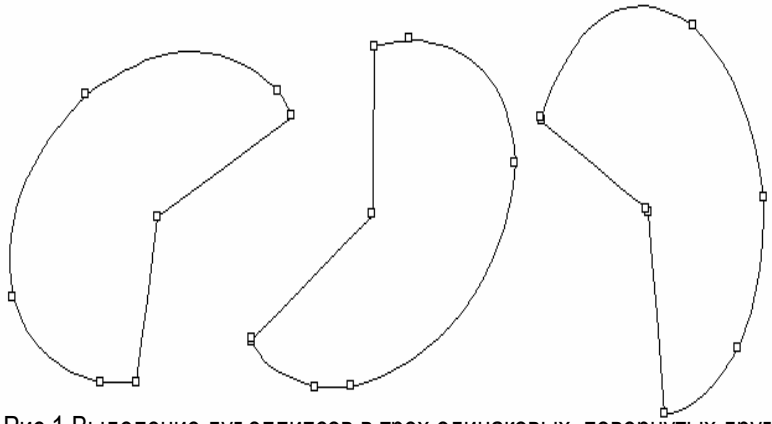


Рис.1 Выделение дуг эллипсов в трех одинаковых, повернутых друг относительно друга объектах средствами Corel Draw

В работах [1,2] приведены алгоритмы, позволяющие выделить отрезок цифровой прямой в контуре. Криволинейные элементы изображений, представленные в виде сплайнов, кривых Безье и т.п., используются в большом количестве приложений. Однако дуги произвольных кривых, как элементы описания контуров не часто используются при распознавании контуров изображений в основном по причине отсутствия общего определения дуги произвольной цифровой кривой. Использование дуг кривых как структурных элементов описания контуров изображений приблизило бы его описание к интуитивному, естественному представлению изображений человеком, существенно сократило бы затраты памяти для хранения изображения и времени его обработки. В качестве примера приведем описание контуров, полученных для бинарных изображений в распространенном графическом редакторе Corel Draw. На рис.1 представлены контуры трех одинаковых объектов без помех, содержащих дугу эллипса, отличающихся положением в пространстве и углом поворота относительно центра каждого из объектов. Граничные точки, которые разделяют контуры на дуги кривых и отрезки прямых, обозначены квадратиками. Одинаковые дуги представлены последовательностями, содержащими неодинаковое количество различных дуг кривых, то есть каждый из одинаковых объектов представлен различными элементами. Такое описание объектов не может быть непосредственно использовано в системах искусственного интеллекта для интерпретации изображений, поскольку предполагает еще достаточно сложную обработку. Приведенный пример показывает существование проблемы даже при обработке изображений, не искаженных помехами и актуальность ее решения.

Контур как последовательность L-элементов

Дуги цифровых кривых, как и отрезки цифровых прямых образуются при дискретизации изображений, содержащих контуры, состоящие из отрезков прямых и дуг кривых. Пикселы изображения имеют форму квадрата, образованного соседними парами вертикальных и горизонтальных линий решетки, использованной при дискретизации изображений. Будем рассматривать дискретное изображение, как двумерный клеточный комплекс [1]. В данном случае двумерные элементы – это пикселы. Помимо пикселов имеются *креки (crac)* и точки. Креки – это стороны пикселов, являющиеся одномерными элементами. Точки являются конечными точками креков и угловыми точками пикселов. Контур объекта в

таком случае – это связная замкнутая последовательность *контурных* креков, граничных между пикселами объекта и фоном. Характерные признаки отрезков прямых и дуг кривых в результате дискретизации утрачиваются.

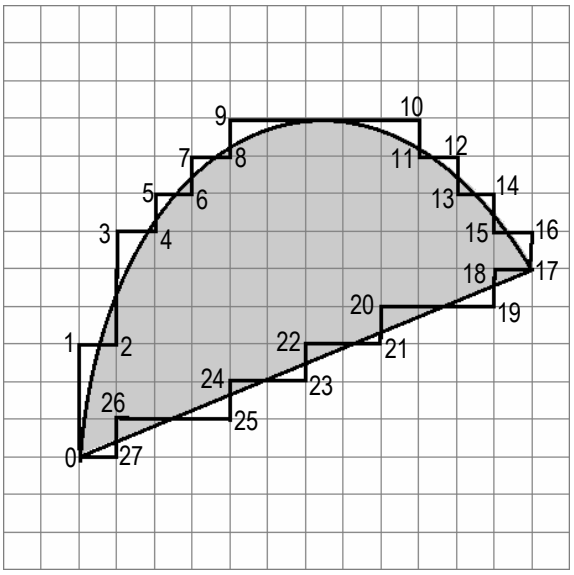


Рис.2 Пример дискретизации контура объекта

На рис. 2 приведен пример исходного контура объекта, образованного дугой кривой и отрезком прямой, а также его цифровой эквивалент как последовательность креков. Связные части последовательности креков можно объединить в ***L-элементы***.

Под L-элементом будем понимать связную последовательность креков вертикальной (горизонтальной) ориентации, которая содержит не более одного крека горизонтальной (вертикальной) ориентации.

Как и работе [2], будем использовать такие L-элементы, в которых отличный по ориентации крек (если он имеется) содержится в конце последовательности. Каждый L-элемент характеризуется такими параметрами: направлением относительно начальной его точки *g* (принято *g*= 0 – для направления вверх, 1 – вправо, 2 – вниз, 3 – влево); *l* – количеством креков направления *g* (*l* = 1,2,…); направлением последнего крека *q* относительно направления *g* предыдущих креков (*q* = –1 – последний крек направлен влево относительно направления *g*, +1 – вправо, 0 – совпадает с направлением *g*). Для L-элемента (0-2) *g*=0, *l*=3, *q*=+1. Для L-элемента (27-0) *g*=3, *l*=1, *q*=0.

Смысл использования L-элементов при обработке визуальной информации, в частности, в процессе структурного анализа контуров заключается в том, что L-элементы являются минимально возможными структурными объектами, еще сохраняющими после дискретизации пространственную ориентацию исходного объекта – отрезка прямой или сегмента дуги кривой. Анализ пространственных свойств L-элементов (направлений, длин) в последовательности позволяет судить об их принадлежности к одному и тому же отрезку прямой.

Очевидно, что последовательность L-элементов, соответствующая отрезку прямой состоит из L-элементов с одинаковыми значениями *g*, *q*, которые определяют направление отрезка цифровой прямой в пределах их значений.

Определение отрезков цифровых прямых в последовательности L-элементов контура

Определить отрезок цифровой прямой в последовательности L-элементов контура – значит указать координаты его начальной и конечной точек. Пусть (x_1,y_1), (x_2,y_2) – целочисленные координаты начальной и конечной точек отрезка прямой. Угол наклона отрезка определяется отношением разностей его координат $n= \Delta x=|x_1-x_2|$ и $m=\Delta y=|y_1-y_2|$, которое в общем случае не является целым числом. Положим для определенности, что $n > m$. Очевидно, что отображение отрезка прямой произвольного угла наклона посредством L-элементов, имеющих целочисленные значения длины l, осуществляется путем чередования L-элементов, длины которых отличаются на 1, например, равны соответственно $l, l+1$, причем $l \le n/m \le l+1$. Порядок их чередования определяет структуру отрезка цифровой прямой и определяется значениями членов цепной дроби $[l; k_1, k_2, \ldots, k_t]$ или

$$\frac{n}{m} = l + \frac{r}{m} = l + \frac{1}{k_1 + \frac{r_1}{r}} = \ldots = l + \frac{1}{k_1 + \frac{1}{k_2 + \frac{r_2}{r_1}}} = \ldots = l + \frac{1}{k_1 + \frac{1}{k_2 + \frac{1}{k_3 + \ldots + \frac{1}{k_t}}}}, \quad (1)$$

получаемой при делении целых чисел *n* и *m*. Для определенности и без нарушения общности будем полагать, что $r<m-r$, $r_1<r-r_1$, $r_2<r_1-r_2$, … и т.д. Как следует из формулы (1) l – целая часть от деления *n* на *m* – соответствует в отрезке цифровой прямой количеству из l подряд идущих креков одного направления. Вместе с примыкающим перпендикулярным креком они образуют L-элемент длины l.

k_1 подряд идущих L-элементов длины l и один L-элемент длины l+1, если он имеется в последовательности, образуют K_1-элемент, состоящий из k_1 L-элементов, иначе говоря, K_1-элемент "длины" k_1. Аналогично, k_2 подряд идущих K_1-элементов длины k_1 и один K_1-элемент длины k_1+1, если он имеется в последовательности, образуют K_2-элемент длины k_2 и так далее до исчерпания членов цепной дроби. Числитель r определяет количество L-элементов длины l+1 в данном отрезке цифровой прямой, а также и количество K_1-элементов, каждый из которых содержит k_1 или k_1+1 L-элементов, один из которых имеет длину l+1. В свою очередь числитель r_1 определяет количество K_2-элементов длины k_1+1 в данном отрезке цифровой прямой, а также и количество K_1-элементов, каждый из которых содержит k_1 или k_1+1 L-элементов, один из которых имеет длину l+1, а остальные – длину l. Вообще числитель r_{t-1} определяет количество K_{t-1}-элементов длины k_{t-1}+1 в данном отрезке цифровой прямой, а также и количество K_t-элементов, каждый из которых содержит k_{t-1} или k_{t-1}+1 K_{t-1}-элементов, один из которых имеет длину k_{t-1}+1, а остальные – длину k_{t-1}.

K_t-элемент, содержащий в своем составе среди K_{t-1}-элементов длины k_{t-1} (L-элементов длины l) один K_{t-1}-элемент длины k_{t-1}+1(один L-элемент длины l+1), называется завершенным.

Под отрезком цифровой прямой с координатами начальной и конечной точек (x_1,y_1), (x_2,y_2) будем понимать последовательность L-элементов, имеющих одинаковые направления g,q, целочисленные длины которых равны соответственно l, l+1, а $l \le n/m \le l+1$, где $n = |x_1-x_2|$ и $m = |y_1-y_2|$, причем порядок чередования L-элементов разных длин определяется цепной дробью n/m.

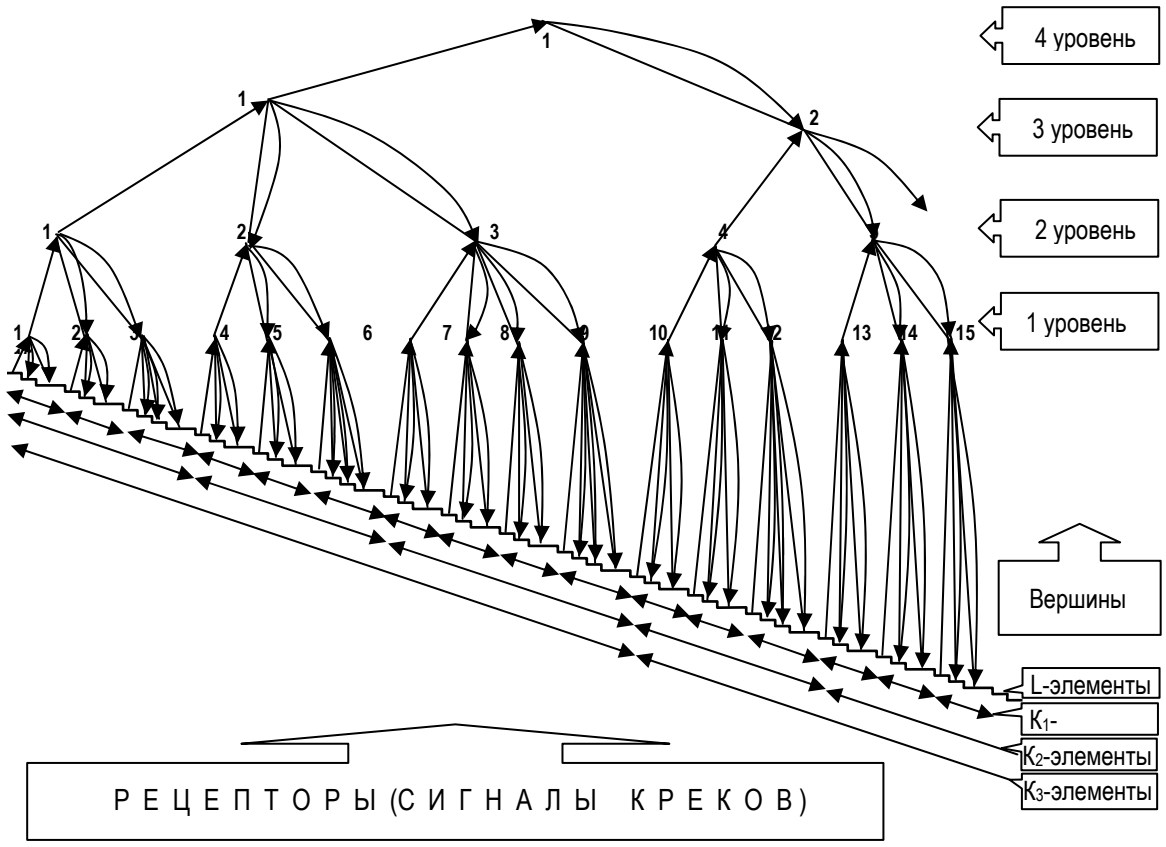


Рис.3 Формирование образа отрезка цифровой прямой в терминах растущей пирамидальной сети

Алгоритм выделения отрезков прямых в последовательности был сформулирован Т.М.Власовой при участии автора [3]. Процесс формирования образа отрезка цифровой прямой упомянутым алгоритмом можно представить в терминах растущих пирамидальных сетей [5] (рис.3) с учетом работ З.Л.Рабиновича [4]. Рассматривается частный случай растущей пирамидальной сети, при котором пирамиды вершин не имеют пересечений. В качестве рецепторов используются сигналы креков, которые формируют

L-элементы. Множество вершин сети связаны концепторами и обратными связями. Концепторы представлены восходящими прямыми стрелками и линиями без стрелок. Обратные связи представлены нисходящими стрелками. Рассматривается последовательная процедура процесса формирования образа – слева направо. Алгоритм формирования образа произвольного отрезка прямой заключается в следующем.

Начальные условия. Счетчик L-элементов s_0=1; общее количество L-элементов в последовательности N; счетчики вершин t уровня s_1= 1, s_2= 1,…, s_t= 1; t = 1; рабочая константа p = 0.

1. Определяются параметры L-элемента с номером s_0 и передаются концептором в вершину номер s_1. Выбор следующего L-элемента s_0:= s_0 +1; Переход к операции 2. Если s_0>N, запоминание значения s_1 и установка s_1 в исходное состояние p = s_1; s_1=1; Переход к операции 3.

2. Посредством дуги обратной связи параметры L-элементов из вершины номер s_1 сравнивают с параметрами следующего по порядку L-элемента номер s_0. При положительном результате сравнения (рассматриваемые L-элементы образуют K_1-элемент) Переход к операции 1. Иначе Конец построения K_1-элемента. Если построенный K_1-элемент является завершенным, то выбор новой вершины первого уровня s_1:= s_1 +1; Переход к операции 1. Иначе конец отрезка цифровой прямой, установка счетчиков вершин в исходное состояние, запоминание координат концов отрезка; Переход к операции 1.

3. Передача параметров K_t-элемента из вершины t-го уровня номер s_t в вершину t+1-го уровня номер s_{t+1} посредством концептора, переход к новой вершине t-го уровня s_t:= s_t +1; Переход к операции 4. Если s_t > p, запоминание значения s_t и установка s_t в исходное состояние p = s_t; s_t=1; выбор следующего уровня t:=t+1. Если количество вершин следующего уровня s_t>1, то Переход к операции 3, иначе Конец работы алгоритма.

4. Посредством дуги обратной связи параметры K_t -элементов из вершины номер s_{t+1} сравнивают с параметрами K_t -элемента из вершины номер s_t. При положительном результате сравнения (рассматриваемые K_t-элементы образуют K_{t+1}-элемент) Переход к операции 3. Иначе конец построения K_{t+1}-элемента. Если построенный K_{t+1}-элемент является завершенным, то переход к новой вершине t-го уровня s_{t+1}:= s_{t+1} +1; Переход к операции 3. Иначе конец отрезка цифровой прямой, установка счетчиков вершин в исходное состояние, запоминание координат концов отрезка; Переход к операции 3.

Определение дуги цифровой кривой

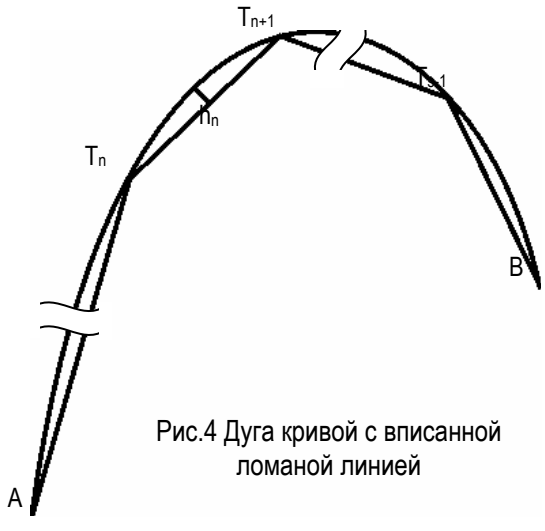


Рис.4 Дуга кривой с вписанной ломаной линией

В данном разделе будет сформулировано такое определение дуги цифровой кривой, которое позволяет установить или отвергнуть факт: часть последовательности отрезков цифровых прямых контура образована в результате дискретизации дуги некоторой произвольной кривой. Будем исходит из того, что дуги кривых, используемых в графических изображениях, отображают сегменты непрерывных функций с непрерывными производными. Под непрерывными кривыми линиями [6], заданными уравнениями $x = \varphi(t)$, $y = \phi(t)$, будем понимать кривые Жордана без кратных точек или простые дуги, то есть такие, что для любых двух разных значений t' и t'' соответствующие им точки на плоскости M′ [$\varphi(t')$, $\phi(t')$] и M'' [$\varphi(t'')$, $\phi(t'')$] – разные. Пусть $x = \varphi(t)$, $y = \phi(t)$, где $\varphi(t)$, $\phi(t)$ – непрерывные функции параметра t, определенные на отрезке [a, b].

При возрастании t от a к b точка с координатами x, y описывает дугу AB (рис.4). Рассмотрим разбиение отрезка [a, b] точками деления

$$a = t_0 < t_1 < \ldots < t_{s-1} < t_s = b, \qquad (2)$$

и пусть этим точкам деления соотвествуют точки кривой A, $T1$, …, T_{s-1}, B. Соединив последовательно отрезками прямых точку A с точкой T_1, точку T_1 с точкой T_2, …, точку T_{s-1}, с точкой B, построим ломанную

линию, и назовем ее ломаной, вписанной в дугу *AB*. Фигуру, ограниченную отрезком ломаной линии T_n, T_{n+1} и соответствующим звеном дуги $\cap Tn, Tn_{+1}$ будем называть сегментом дуги T_n, T_{n+1}, а максимальную длину линии между отрезком T_n, T_{n+1} и $\cap T_n, T_{n+1}$, перпендикулярной к отрезку T_n, T_{n+1} – высотой сегмента дуги h_n. Пусть

$$\beta = \max_{n=0,1,\ldots,s-1} l(T_n, T_{n+1}) \qquad (3)$$

Если β будет стремиться к нулю при соответствующем увеличении *s*, то к нулю будет стремиться длина любого из звеньев вписанной ломаной, также как и высота каждого сегмента дуги, благодаря непрерывност функций $\varphi(t)$, $\phi(t)$.

При отображении дуги и вписанной ломаной линии в дискретном пространстве дискретности *d*, отрезки вписанной ломаной будут отображаться отрезками цифровых прямых. Поскольку в дискретном пространстве значения координат принимают целочисленные значения, кратные *d*, то, начиная с момента, когда $h_n < d$, объекты, меньшие величины дискретности, в частности, высоты сегментов не будут отображены в этом пространстве, – длины их станут равными нулю. Итак, при $h_n < d$ дискретные отображения частей дуги совпадут с соответствующими звеньями вписанной ломаной – отрезками цифровых прямых. Таким образом, контур, который состоит из отрезков прямых и дуг произвольных кривых, после дискретизации определен как последовательность отрезков цифровых прямых. Последовательности отрезков цифровых прямых, которые соотвествуют дугам кривых, могут рассматриваться как ломанные линии, вписанные в эти дуги кривых. Такие вписанные ломаные линии будем называть дугами цифровых кривых. Контур может включать как отдельные отрезки прямых, так и последовательности таких отрезков - ломаные линии, которые не являются дугами цифровых кривых.

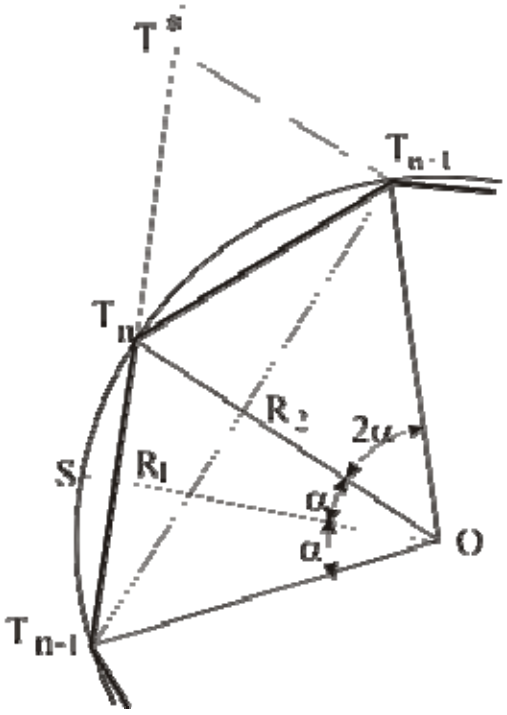


Рис. 5 Оценка величины отклонения направления соседних отрезков дуги цифровой кривой

Будем рассматривать пары соседних отрезков цифровых прямых в последовательности, которая соотвествует дуге кривой. Пара соседних отрезков определяет конечную разность второго порядка. Две пары, которые имеют общий отрезок, будем называть соседними парами. Соседние пары определяют конечную разность третьего порядка.

Если конечные разности второго порядка не равны нулю (для целочисленних значений координат точек конечные разности должны быть больше 1), то не исключено, что пары отрезков цифровых прямых являются частью дуги цифровой кривой. Вообще, через три точки, определенные парой отрезков, можно провести много кривых. Тем не менее, как уже отмечалось, длины высот сегментов дуг кривых, которые соотвествуют отрезкам вписанной ломаной линии, не должны превышать значения дискретности пространства *d*. Таким образом, для того чтобы считать пары отрезков цифровых прямых $T_{n-1}T_n$, T_nT_{n+1} частью дуги цифровой кривой, необходимо установить существование кривой, которая проходит через точки T_{n-1}, T_n, T_{n+1}, такой, для которой выполняется условие: $(h_{n-1}<d)\&(h_n<d)$.

Кривизну плоской кривой обычно отождествляют с кривизной соприкасающейся окружности [7]. Соприкасающейся окружностью плоской кривой в точке T_1 называют предельное положение окружности, проходящей через две соседние точки T_2 и T_3 при стремлении T_2 и T_3 к T_1. Оставляя в стороне решение этой задачи в общем случае, далее будем рассматривать частный случай этой задачи, когда кривой второго порядка, проходящей через три точки является окружность.

На основании приведенных соображений можно сформулировать следующее ***определение***:

Под дугой цифровой кривой в двумерном дискретном пространстве дискретности d будем понимать такую последовательность отрезков цифровых прямых, что через конечные три

точки каждой пары соседних отрезков можно провести такую окружность, что высота сегментов этой окружности для соседних отрезков не превышает d.

Это определение справедливо в той мере, насколько правомерно отождествлять сегмент дуги произвольной кривой, которая отвечает паре соседних отрезков с дугой касательной окружности.

Построив окружность в соответствии с определением дуги цифровой кривой по точкам T_{n-1}, T_n, T_{n+1}, сделаем оценку величины отклонения конца каждого такого отрезка от направления линии предшествующего отрезка (рис. 5). Величиной отклонения может служить длина отрезка $T_{n+1}T'$ при условии, что $l(Tn'') \cong l(T_n, T_{n+1})$. Определим сначала длину отрезка Tn_2 - высоты треугольника $T_{n-1}T_n\,T_{n+1}$. Как уже было отмечено, максимальное расстояние между точками линий дуги кривой и соответствующего отрезка цифровой прямой $SR_1 = d$. В то же время $SR_1 = OT_{n-1} - OT_{n-1} \times \cos\alpha = r - r\cos\alpha = r(1 - \cos\alpha)$. Длина высоты $\Delta(T_{n-1}T_n\,T_{n+1})$ $Tn_2 = OT_n - OT_n \times \cos 2\alpha = r - r\cos\alpha = r(1 - \cos 2\alpha) = 2r(1 - \cos 2\alpha)$. $Tn_2 / SR_1 = 2(1 + \cos\alpha)$; или $Tn_2 = 2(1 + \cos\alpha) \times SR_1$. Если $SR_1 \approx d$ и $\alpha \le 30°$, то высота треугольника $(T_{n-1}T_n\,T_{n+1})$ $Tn_2 \le 3{,}85_d$. Нетрудно видеть, что максимальное отклонение

$$T_{n+1}T' = 2 \times T_nR_2 \approx 7.7\mathrm{d}. \qquad (4)$$

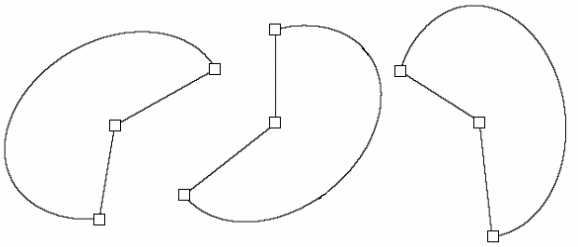


Рис.6 Сегментация объектов с использованием предлагаемых методов

Это означает, что для того, чтобы рассмотренная пара отрезков могла быть отнесена к дуге цифровой кривой необходимо, чтобы величина максимального отклонения Tn' не превышала бы 7.7d. Минимальная величина отклонения $T_{n+1}T' > d$, поскольку при меньшем значении отклонения направления отрезков $T_{n-1}T_n$ и T_nT_{n+1} неотличимы, и пара отрезков разных направлений превращается в один отрезок прямой. Величина отклонения $d < g_n = T_{n+1}T' < 7.7d$ также является значением второй конечной разности на сегменте контура $T_{n-1}T_n\,T_{n+1}$, в точке T_n при условии, что данный сегмент есть частью дуги кривой.

Пример работы программы распознавания контура как последовательности отрезков цифровых прямых и дуг цифровых кривых приведен на рис. 6. Показаны результаты обработки того же изображения, что и на рис. 1, но программой, реализующей предлагаемый алгоритм. В отличие от контуров рис. 1, полученных посредством программы Corel Trace, дуги контуров представлены без разбивки промежуточными точками на несколько дуг независимо от положения в пространстве и угла поворота.

Заключение

Рассмотрено построение образа отрезка цифровой прямой, как частного случая растущей пирамидальной сети [5] с учетом концептуальной модели опознавания и памяти [4].

Предложено определение дуги цифровой кривой как последовательности отрезков цифровых прямых [3]. Разработанные на основе достигнутых результатов алгоритмы и программы позволяют, в отличие от известных, выполнять сегментацию контуров естественным образом на отрезки цифровых прямых и дуги цифровых кривых независимо от положения в пространстве, угла поворота, масштаба рассматриваемого объекта.

В данной работе приведены результаты обработки контуров бинарных изображений без потери информации. Вопросы обработки полутоновых изображений при наличии помех являются предметом следующих работ.

Литература

[1] Kovalevsky V.A. Applications of Digital Straight Segments to Economical Image Encoding, In Proceedings of the 7th International Workshop, DGCI'97, Montpellier, France, December 3-5, 1997, Springer 1997, P. 51-62.

[2] Вишневский В.В., Калмыков В.Г. Структурный анализ цифровых контуров изображений как последовательностей отрезков прямых и дуг кривых. // Искусственный интеллект 4'2004, Институт проблем искусственного интеллекта НАНУ – Донецк: "Наука и образование" 2004 С.450-457

[3] Власова Т.М., Калмыков В.Г. Алгоритм и программа распознавания контуров изображений как последовательности отрезков цифровых прямых // Математичні машини і системи. – 2005. -№4 С.84-95

[4] Рабинович З.Л. О естественных механизмах мышления и интеллектуальных ЭВМ // Кибернетика и системный анализ. – 2003. -№5 С.82-88

[5] Гладун В.П. Планирование решений. – Киев: Наукова думка, 1987. – 167с.

[6] Макаров И.П. Дополнительные главы математического анализа. – Москва: Просвещение, 1968.

[7] Корн Г. и Корн Т. Справочник по математике для научных работников и инженеров. – Москва: Наука, 1974

Информация об авторе

Владимир Калмыков - *старший научный сотрудник, кандидат технических наук, Институт проблем математических машин и систем, просп. акад. Глушкова 42, 03680, Киев 187, Украина; e-mail: kvg@immsp.kiev.ua*

ТЕХНОЛОГИИ КОМПРЕССИИ ГРАФИЧЕСКОЙ ИНФОРМАЦИИ

Николай Б. Фесенко

Abstract: *The classification of types of information redundancy in symbolic and graphical forms representation of information is done. The general classification of compression technologies for graphical information is presented as well. The principles of design, tasks and variants for realizations of semantic compression technology of graphical information are suggested.*

Keyword*: semantics, abundance, compression, graphical information, decompression, classification, ontology.*

Введение

Представление графических файлов в компактном сжатом виде необходимо для удобства хранения и обеспечения приема и передачи информации по каналам связи. При этом избыточность в представлении информации позволяет эту информацию сжать, т.е. сократить ресурсы, затрачиваемые на ее представление. Когда для уменьшения размеров памяти, занимаемых файлами, или при подготовке файлов к пересылке в компактной форме применяются специализированные программы, то принято говорить, что файлы подвергаются сжатию, или компрессии[1]. Технологии сжатия файлов используют, как правило, программы, уменьшающие размеры файлов графики за счет изменения способа организации данных, например, заменяя повторяющиеся элементы данными, более эффективными для хранения. При этом графическая информация сжимается только по форме представления, но не по содержанию. Представляется актуальной разработка технологии семантического сжатия графической информации. С этой целью исследуем технологии компрессии графической информации на предмет применения в них подходов и методов искусственного интеллекта.

Разновидности информационной избыточности

Каждая форма, или модель, разного представления фактически одной и той же мысли отличается информационной избыточностью - от минимальной до значительной. Это может быть либо набросок, зарисовка, либо синтезировано качественное цветное компьютерное изображение. Причем, чем адекватнее выбранная модель, тем лучше достижимая степень сжатия. Рассмотрим *виды информационной избыточности*, характерные для различных организационных форм представления информации, и выполним их классификацию с целью определения возможного скрытого резерва для проведения дополнительной, более глубокой компрессии (рис.1).

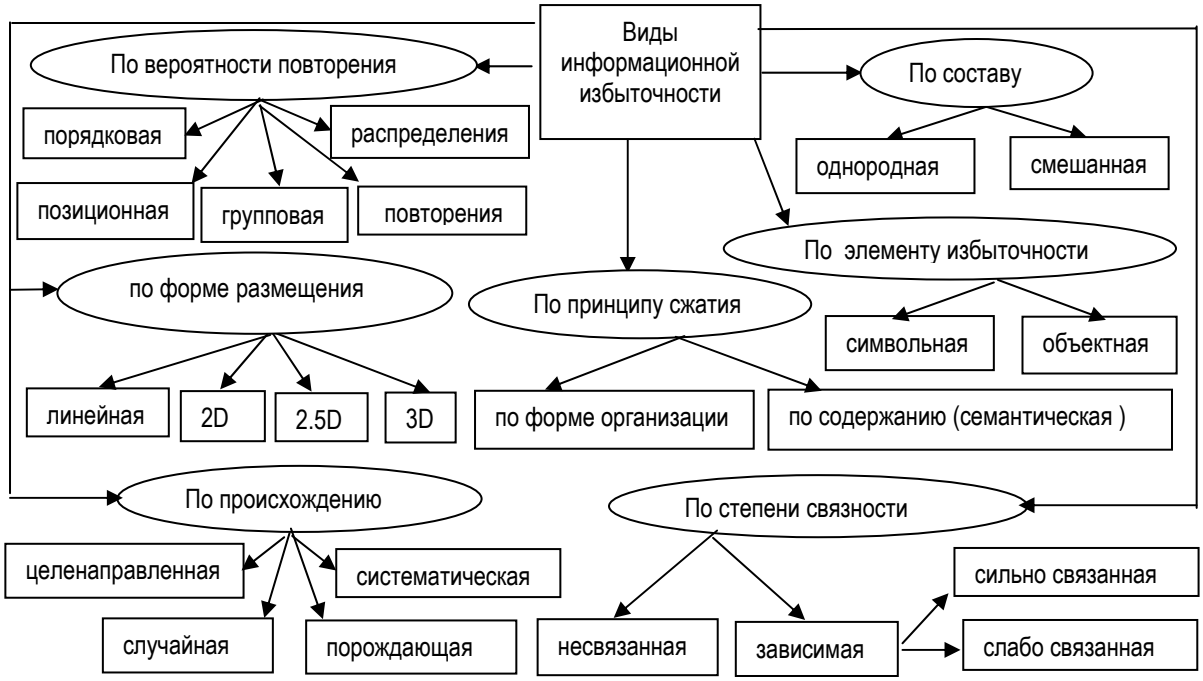


Рис.1 Классификация информационной избыточности.

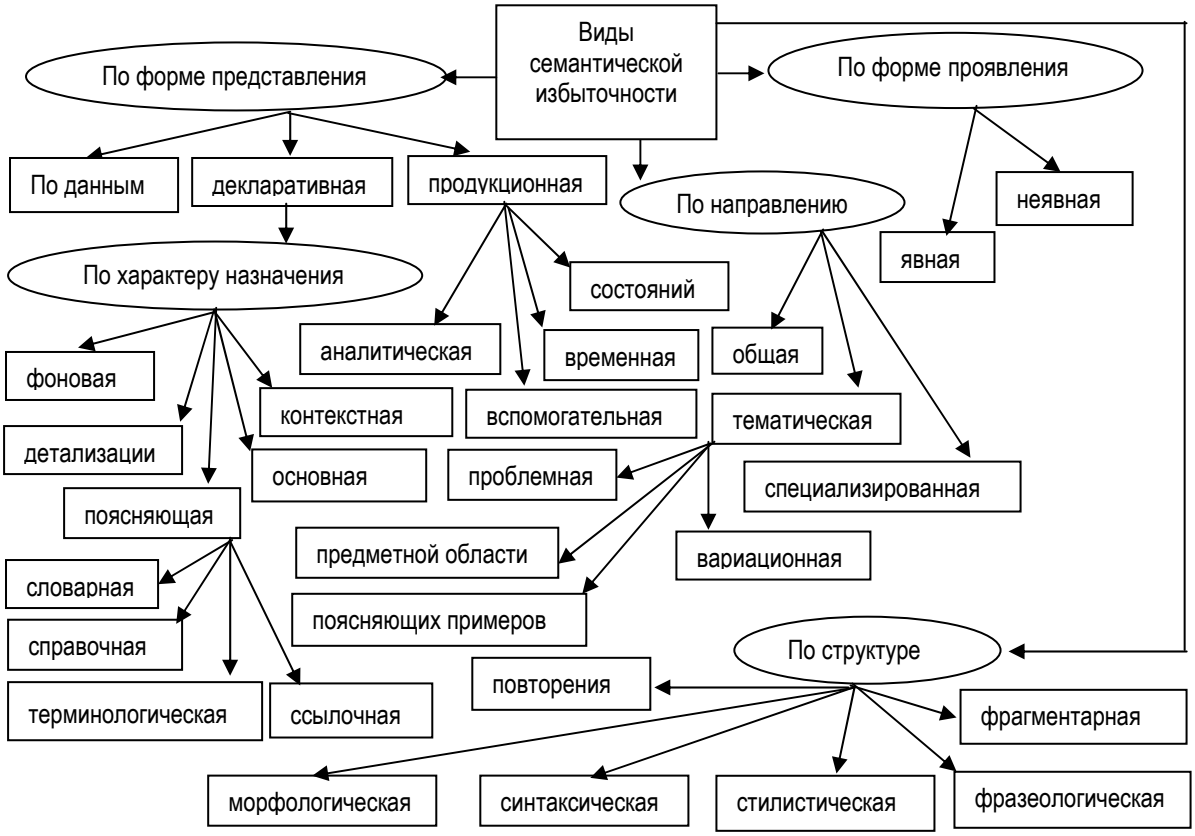


Рис.2 Классификация семантической избыточности.

Основные виды избыточности[2] близки для символьной и графической форм представления информации. При этом для каждого вида избыточности существуют свои модели сжатия. Моделирование в той или иной степени отражает возможность предсказания вероятности наступления возможных событий. ***По вероятности повторения***, когда разные события имеют разные вероятности своего появления, то имеет место избыточность *распределения*. Если несколько одинаковых событий могут следовать друг за другом, то имеет место избыточность *повторения*. Избыточность *порядка* появляется тогда, когда наступление определенных событий можно предсказать в зависимости от некоторого порядка следования. Повторение группы событий приводит к *групповой* (цепочной) избыточности. Возможность учета вероятности появления определенных событий в некоторых позициях потока событий определяет *позиционную* избыточность. Вероятность близости значений соседних пикселей графического изображения, как и их пространственных производных, характеризует пространственную избыточность ***по форме размещения***: *линейную, 2D, 2.5D, 3D*. ***По составу*** реже встречается *однородная* избыточность, чаще - *смешанная*. ***По элементу избыточности*** проявляется *символьная* и *объектная*. ***По принципу сжатия*** будем различать избыточность *по форме организации* размещения, или расположения символа или образа, и *семантическую*, или смысловую избыточность *по содержанию*. ***По происхождению*** выделим *целенаправленную*, или преднамеренную избыточность, *случайную*, *порождающую* и *систематическую*. ***По степени связности*** отметим *несвязанную*, или независимую избыточность, и *зависимую*, которая может быть как *слабо связанной*, так и *сильно*. Из проведенного анализа и классификации избыточности следует, что менее других исследован семантический аспект избыточности.

Разновидности семантической избыточности

Проведем анализ видов семантической избыточности и выполним их классификацию для определения задач по созданию технологии семантической компрессии графической информации (рис.2). Семантическая избыточность различается ***по форме представления***, ***по форме проявления***, ***по направлению***, ***по структуре*** и ***по характеру назначения***. ***По форме представления*** выделим избыточность *по данным*, *декларативную* и *продукционную* избыточности. ***По характеру назначения*** *декларативную* избыточность будем разделять на *основную*, *фоновую*, *контекстную*, *поясняющую* и избыточность *детализации*. Вероятность появления символа в контексте будем именовать *контекстной* избыточностью. В свою очередь, *поясняющая* избыточность проявляется как *словарная*, *справочная*, *терминологическая* и *ссылочная*. *Продукционная* избыточность включает *аналитическую*, *временную*, избыточность *состояний* и *вспомогательную*, к которой отнесем также вспомогательные контейнеры модификаторов. ***По форме проявления*** выделим *явную* избыточность и *неявную*. ***По направлению*** проявляется *общий* характер избыточности, *тематический* и *специализированный*. *Тематическая* избыточность подразделяется на *проблемную*, *предметной области*, *вариационную* и избыточность *поясняющих примеров*. ***По структуре*** отметим *морфологическую*, *синтаксическую*, *стилистическую*, *фрагментарную*, *фразеологическую* и избыточность *повторения*. Предложенная классификация позволяет сосредоточить внимание на семантической избыточности в контексте сжатия графической информации, поскольку на сегодняшний день практически отсутствуют ее программные реализации.

Анализ и классификация технологий сжатия графической информации

С целью систематизации проведем анализ и общую классификацию технологий сжатия графической информации (рис.3). ***По форме представления графической информации*** будем различать *растровую*, *векторную*, *программную*, *текстовую*, *мультимедийную* и *комбинированную* формы представления. *Растровая* форма характеризуется значительной пространственной избыточностью. Тип избыточности *векторной* формы зависит от большого числа факторов и не обладает достаточной универсальностью для создания общих формальных подходов к ее устранению [2]. Среди форматов векторной графики выделим два: CDR (CorelDRAW) и DXF (AutoCAD), которые являются стандартами для соответствующих графических пакетов. Помимо растровых, векторных, текстовых форматов, разработаны также смешанные форматы, или *метафайловые*. Например, форматы ESP (Encapsulated PostScript) и CGM (Computer Graphics Metafile) позволяют хранить растровые, векторные и текстовые данные. Универсальные файлы данных UDF (Universal Data File) охватывают различные структуры данных и могут входить в состав документов, реализованных в других форматах.

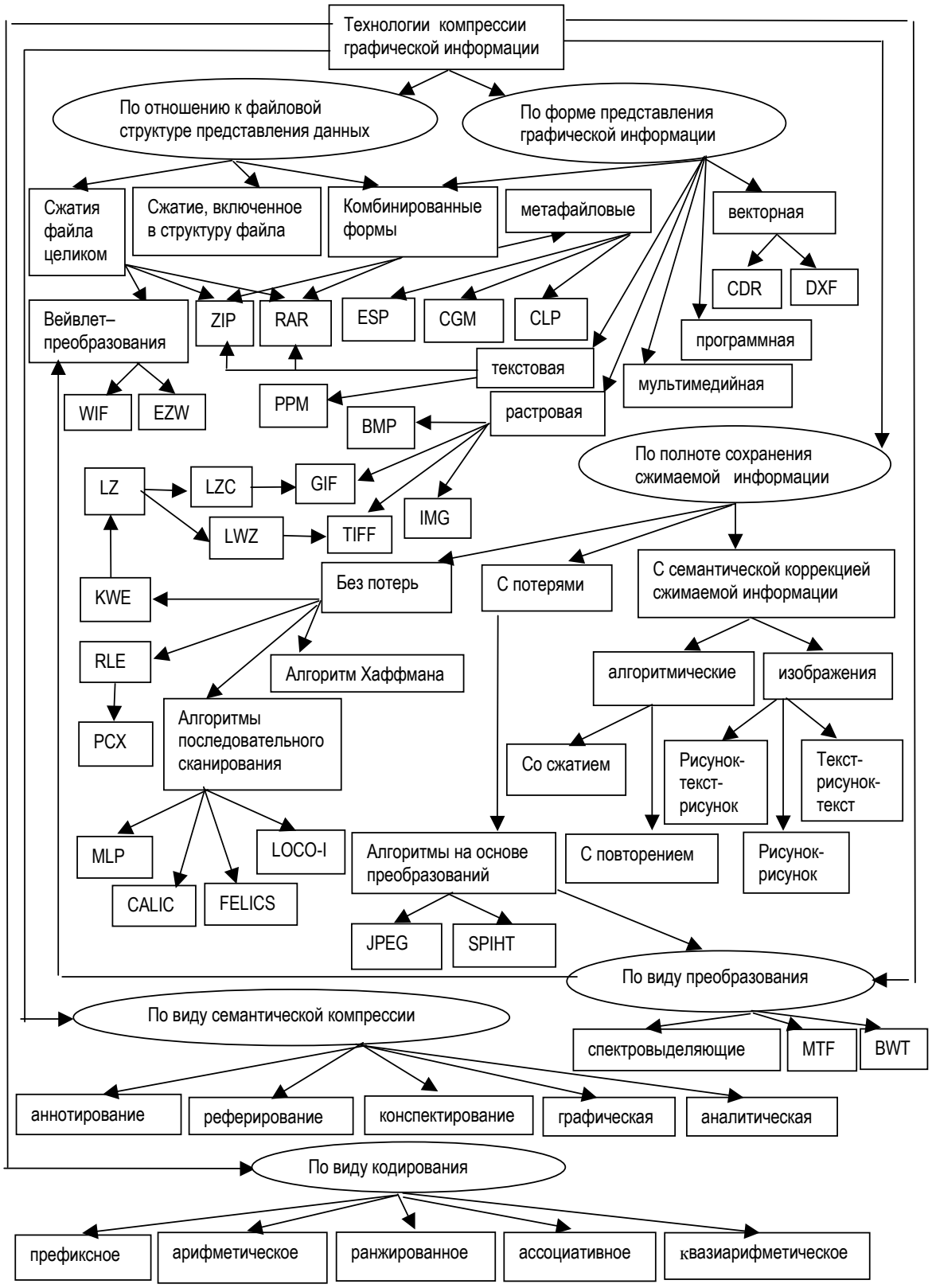


Рис.3 Общая классификация технологий сжатия графической информации

Представление сцены изображения в виде программных файлов как на языках программирования, так и языках инструкций графических пакетов, позволяет в значительной степени сократить объем информации об изображении, поскольку такой подход предполагает применение иной формы описания изображения и допускает использование в графических файлах программных процедур. Однако при этом обязателен компилятор, или наличие соответствующего графического пакета. Для более сложных изображений требуется полный цикл повторного создания изображения.

По отношению к файловой структуре представления данных анализируемые технологии выполняют сжатие сразу всего файла целиком или сжатие, включенное в структуру файла; а также комбинированные формы. Форматы, в которых сжатие является частью структуры файла, менее зависимы при их дальнейшем использовании. Полностью сжатый файл обычно нельзя использовать до тех пор, пока он не будет восстановлен до исходного состояния. Поэтому, в дальнейшем, сжатый формат растровых изображений декомпрессируется соответствующей совместимой программой. Наиболее распространены форматы сжатия ZIP и RAR, которые используются для сжатия как символьной, так и графической информации. Вейвлет-компрессия графических данных, которую отнесем к технологиям **по виду преобразований**, заключается в сжатии всего изображения целиком до так называемых WIF-файлов (Wavelet Image File) с целью сохранения временных и частотных характеристик сигналов. Основная идея лифтинг-преобразований, называемых также вейвлетами второго поколения, выражается в том, что при каждой операции должна изменяться только одна составляющая, поэтому, например, для построения обратного преобразования достаточно выполнить зеркальное отображение прямого преобразования и добавить инвертирование сигнала там, где это нужно. Алгоритм сжатия *EZW* (Embedded Zerotree Wavelet) предполагает простое отбрасывание коэффициентов со значениями, меньшими некоторой величины. Не намного более сложный в вычислительном отношении алгоритм *SPIHT* (Set Partitioning in Hierarchical Trees) дает гораздо лучшие результаты компрессии.

По полноте сохранения сжимаемой информации разделим компрессионные алгоритмы на выполняющие сжатие *без потерь*, *с потерями* информации и *с семантической коррекцией сжимаемой информации*. Известные технологии сжатия *без потерь* практически не выполняют семантического анализа сжимаемых данных, и при декомпрессии стремятся восстановить как файл, так и рисунок в полном исходном объеме. Распространенные технологии компрессии графической информации, выполняющие сжатие *с потерями* информации, не учитывают смыслового содержания этой информации. Следовательно, *потеряна* может быть именно та часть графического изображения, ради которого оно, собственно, все сохранялось и передавалось. Даже полутоновые искажения при сжатии могли бы носить избирательный характер, например, не распространяться на изображение лица портретного изображения, а затрагивать только фон и одежду.

В работе *алгоритмов* на основе *последовательного сканирования*, ориентированных на сжатие изображений *без потерь*, можно выделить следующие фазы: предсказание, моделирование ошибки и кодирование, причем предсказание значения следующего пикселя производится на основе значений уже закодированных соседних пикселей. В основу технологий сжатия *по ключевым словам KWE* (Key Word Encoding) положен принцип кодирования лексических единиц, например обычных слов, группами байт фиксированной длины. Результат такого кодирования помещается в специальном словаре. Технология Лемпеля-Зива-Велча (*LZW*), которая является модификацией технологии Лемпеля-Зива (*LZ*), основана на поиске и сохранении шаблонов внутри заданной структуры. Алгоритм *LZW* построен вокруг таблицы фраз (словаря), которая заменяет строки символов сжимаемого сообщения на коды фиксированной длины. Сжатие по этой технологии выполняет растровый формат *TIFF*, который поддерживает большое количество алгоритмов сжатия для различных типов изображений. В растровом формате *GIF* изображение рассматривается как одномерная последовательность пикселей, которые кодируются по строкам с использованием алгоритма *LZC* (модификация алгоритма *LZW*). Формат рассчитан на изображения с индексными цветами и обладает более высокой степенью сжатия, чем формат *PCX*. Существуют модификации этого формата, которые отличаются количеством поддерживаемых типов блоков расширений, в том числе анимации. В целом алгоритмы *KWE* выполняют сжатие *без потерь* информации, и наиболее эффективны для текстовых данных больших объемов, но из-за необходимости создания и сохранения словаря малоэффективны для файлов небольших размеров. В основе *алгоритма Хаффмана*, также производящего сжатие *без потерь* информации, лежит идея кодирования битовыми группами. После проведения частотного анализа входной последовательности данных, то есть

определения частоты вхождения каждого символа, который в ней встречается, символы сортируются по уменьшению этой частоты. Основная идея состоит в следующем: чем чаще встречается символ, тем меньшим количеством бит он кодируется. Результат кодирования заносится в словарь, необходимый для декодирования. Алгоритм *RLE* (Run-Length Encoding) основан на выявлении повторяющихся последовательностей данных и замены их более простой структурой, в которой указывается код данных и коэффициент повторения, поэтому он дает лучший эффект сжатия при большей длине повторяющейся последовательности данных. Следовательно, алгоритм *RLE* более эффективен при сжатии графических данных (в особенности для однотонных изображений). Кодирование по алгоритму *RLE* поддерживает графический формат *PCX*.

Алгоритмы ***по виду преобразований*** предназначены для сжатия изображений *с потерями* качества. Типовая схема алгоритма такова – преобразование, квантизация, моделирование преобразованных коэффициентов, кодирование. Существуют *два подхода* к выполнению преобразований. *Первый* выражается в многократном выполнении преобразования, отделяющего самую высокочастотную составляющую изображения. *Второй подход* заключается в выполнении. преобразования, результатом которого является спектр изображения. Высокая вычислительная сложность таких преобразований не позволяет применять их для фрагментов изображения большого размера. Тогда при сжатии *с потерями*, например в формате *JPEG*, происходит отбрасывание части данных исходного изображения. При этом используются особенности чувствительности нашего зрения по восприятию изменений яркости, цвета или тона. Сначала изображение разделяется на яркостную и две цветовые составляющие. Яркостная составляющая более важна при восприятии изображения человеческим глазом, поэтому ее следует сжимать с лучшим качеством. Затем эти составляющие разбиваются на квадраты размером 8x8 пикселей, и для каждого из которых выполняется дискретное косинус-преобразование Фурье. После этого производится квантизация результатов и кодирование. Для кодирования коэффициентов используется *кодирование Хаффмана* с фиксированными таблицами, как разновидность *префиксного* кодирования, либо *арифметическое* кодирование. ***По виду кодирования*** выделим: *префиксное*, *арифметическое*, *ранжированное*, *ассоциативное* и *квазиарифметическое* кодирование.

Разделение ***по виду семантической компрессии*** позволяет выделить *аннотирование*, *реферирование*, *конспектирование*, или изложение, как разные по степени сжатия и обобщения универсальные формы пересказа содержания, а также *графическую*, или образную, и *аналитическую* (формализация) формы представления содержания. Причем две последние формы выполняют функцию сжатия графической информации только в системе некоторых ограничений.

Семантическая компрессия графической информации

Рисунок, как наиболее простая из возможных и распространенных форм представления информации, может быть предназначен для пояснения текстового описания, или наоборот. Поэтому проанализируем *три схемы*: *Текст-Рисунок-Текст, Рисунок–Текст-Рисунок* и *Рисунок-Рисунок*. *Первая схема* целесообразна для предметных областей с развитой системой сжатия графической информации. превосходящей алгоритмы сжатия текстовых естественно-языковых или программных структурных описаний. *Вторая схема* применима для случая предметной области, в которой сжатие естественно-языковых текстов и программных формализмов выполняется лучше, чем компрессия графических представлений информационного материала. *Третья схема* предполагает семантическое сжатие непосредственно графических изображений без промежуточных вспомогательных текстовых форм. К этой же схеме примыкает задача понимания графических языков: рисунков, чертежей, иллюстраций, фотоизображений и др. Такие *схемы* допускают несколько вариантов реализации. В ВЦ РАН [3] построена система ТЕКРИС - генерации рисунков на основе текстов, использующая схему: анализ текста - синтез графического образа.

Мы предлагаем производить процесс сжатия графической информации на основе семантического анализа. Удачным решением представляется введение в технологию сжатия графической информации онтологических подходов. Одним из вариантов практической реализации такого подхода является текстовое описание рисунков и применение к полученному описанию методов семантического сокращения, к которым отнесем конспектирование естественно-языковых текстов, в частности, на основе растущих пирамидальных сетей.

В качестве примера реализации конспектирования естественно-языковых текстов приведем программу КОНСПЕКТ[4], которая передает смысл, но не восстанавливает исходный текст после проведенного сжатия. При работе этой программы фразы исходного текста подвергаются синтактико-семантическому анализу с целью отбора полносоставных предложений. После этого на основе онтологии ассоциаций осуществляется тематический анализ текста. Онтологией ассоциаций является словарь терминов, в котором термины индексируются наборами ассоциативных признаков, обозначающих ассоциации понятий. В результате работы программы КОНСПЕКТ формируется обобщенный текст, который по объему и связности близок к сложившимся представлениям о свойствах конспектов. Другим вариантом является использование словарного комплекса РУСЛАН [5] как инструмента сжатия содержания текста. Функцию, аналогичную по производимому эффекту декомпрессии, выполняют и стандартные информационно-поисковые системы: используя заданные извне ключевые слова и терминологические словосочетания, которые являются предельно сжатой формой представления темы, они находят полное содержание текста.

Графический объект можно считать определенным только тогда, когда зафиксирована его форма для визуализации, проставлены размеры и выполнена пространственная ориентация и расположение[6]. Одной из первых программ, которая в интерактивном режиме могла воспринимать взаимное расположение блоков разных форм и цветов и производить их описание, является программа Винограда SHRDLU[7], однако ее разработчикам не удалось решить задачу абстрагирования. Семантическое сжатие непосредственно графического изображения также близко семантическому масштабированию. Отличие лишь в том, что при семантическом сжатии сокращенная информация опускается или видоизменяется, а при семантическом масштабировании она временно исключается при увеличении масштаба, но затем, при уменьшении масштаба, вновь возвращается. При декомпрессии также возможен семантический аспект, т.е. декомпрессия изображения с раскрытием изображаемой темы и ее дополнении информацией из базы знаний или базы данных, например, в самом простом варианте, списком литературы. Ряд алгоритмов компьютерной графики [8,9] используют генерирование семейств отрезков прямых, окружностей, эллипсов, парабол и гипербол, и допускают применение сжатия и повторения. Выигрыш от применения метода повторений тем больше, чем длиннее генерируемый отрезок и чем больше коэффициент повторения. Для каждого квадранта с учетом асимптотических ветвей и симметрии определяются последовательности, по которым вычисляются точки, наиболее близко расположенные к проектируемой кривой линии и соответствующим образом распределяется яркость между выбранной алгоритмически основной точкой и вспомогательной, выбор которой зависит от расстояния относительно кривой. Такие алгоритмы применимы при разработке программ семантической декомпрессии.

Основные принципы построения технологии семантической компрессии графической информации

В общем контексте развития существующих компрессионных технологий сформулируем основные принципы построения и задачи по созданию технологии семантической компрессии графической информации. *Главным принципом*, который определяет предлагаемый подход к проблеме ***сжатия графической информации по ее смыслу***, является ***извлечение знаний из информации, представленной в графической форме***. Общую стратегию достижения этой цели необходимо строить на принципах, которые изложены ниже.

Первый принцип - ***представление сцены изображения графическими примитивами***. Задачи анализа и синтеза сцен графических изображений основаны на том, что такие сцены предварительно описываются при помощи словаря графических примитивов: точка, отрезок, линия, треугольник, прямоугольник, квадрат, ромб, многоугольник, дуга, окружность, круг, эллипс, гипербола, парабола. Из них можно составить многие другие графические объекты, поскольку разработаны квазиобобщенные методы представления графических объектов с помощью прямых и окружностей с использованием некоторых приемов сопряжения. Основными пространственными графическими объектами обычно считаются параллелепипеды, пирамиды, цилиндры, конусы вращения, сферы и торы. В случае несложного графического объекта задача его представления состоит в объединении этих тел с помощью сложения или вычитания и в выполнении операций описательной геометрии, переносов, поворотов, сечений, наложений. Однако, в большинстве случаев с помощью тел, называемых аналитическими (т.е. по сути

графические примитивы) нельзя описать реальные механические объекты. Для этого можно использовать существующие методы определения формы объектов, основанные на последовательной подгонке.
Второй принцип - ***формализация представления сцены изображения*** на основе разделения сцены изображения на отдельные графические объекты, скомпонованные в композицию.
Третий принцип ***- распознавание графических объектов.*** Графический объект можно как распознать сразу, так и решить эту задачу за несколько шагов, а также скопировать и подобрать похожий объект с известным аналитическим описанием. Для этого можно применить сшивающие поверхности. В самом простом случае такими поверхностями являются огибающие сферы постоянного радиуса, которые обеспечивают плавное изменение касательной при переходе от одной сшиваемой поверхности к другой. Решение задач узнавания или различения объектов по их изображениям относится функционально к машинному зрению. Следует заметить, что для поверхностей сложной формы, таких как: поверхности двойной кривизны, поверхности переменной кривизны, искривленные поверхности, решение задачи начинается с выполнения последовательности шагов аппроксимации. Задача состоит в математически точном воспроизведении формы графического объекта, исходя из координат точек, расположенных на его поверхности. Измерение наклонов касательных тоже может быть использовано для этой цели, однако оно не обеспечивает достижение точности, сравнимой с точностью методов, основанных на измерении линейных координат. При этом одной из промежуточных задач выступает классификация поверхностных форм графических объектов.
Четвертый принцип - ***составление онтологии графических ассоциаций***. Механизм смыслового сжатия должен быть заложен в систему, осуществляющую автоматическое понимание текстового описания изображения. Такие функции должны быть также учтены при проектировании используемого словаря графем. Если словарь графических терминов и примитивов объединить со словарями более общих понятий, характеризующих тематические предметные области, то можно составить онтологию графических ассоциаций для некоторого их подмножества.
Пятый принцип - ***семантическое сжатие можно проводить в интерактивном режиме с участием эксперта.***
Шестой принцип - ***перевод сцены графического изображения в промежуточную текстовую форму представления.***
Седьмой - ***семантическое сжатие текстовой формы представления графического изображения***. Проведение предварительной трансляции графического изображения в текстовое описание и применение к такой форме модификации известных технологий семантического сжатия символьной информации не является единственно возможным решением, так как к текстовому описанию можно дополнительно применить технологии символьного сжатия формы представления, например, *PPM* или *комбинированные* варианты.
Восьмой принцип - ***восстановление изображения из промежуточной текстовой формы представления в графическую.*** После проведения семантического сокращения текстового представления рисунка, производим восстановление его изображения в семантически сжатом варианте.
Девятый принцип - ***смысловая декомпрессия графической информации.***

Выводы

Семантическая информация, которая содержится в графическом представлении, может быть извлечена с помощью онтологического описания проблемной области, построенного на базовых графических примитивах и объектах, и использована затем для компрессии и декомпрессии графических изображений. Процедурной основой для компрессии содержания рисунка (изображения) может также служить перевод графики в текстовое описание с последующим смысловым сжатием текста при помощи программ типа КОНСПЕКТ. Будущие интеллектуальные технологии семантической обработки графической информации должны развиваться с учетом достижений в области текстовых сообщений и использовать онтологические конструкции, специализированные для графических приложений.

Таким образом, в настоящей работе по результатам анализа видов информационной избыточности и систематизации технологий компрессии графической информации в виде их общей классификации указаны *задачи*, *основные принципы построения* технологии семантического сжатия графической информации, основанной на применении семантической компрессии к ее промежуточным посредническим формам представления, формализации и текстовому описанию, и предлагаются варианты ее реализации.

Литература

1. Корриган Дж. Компьютерная графика: Секреты и решения. -М.: Энтроп, 1995.-352с.
2. Балашов К.Ю. Сжатие информации: анализ методов и подходов. // Препринт / Ин-т техн. Кибернетики НАН Беларуси; № 6– Минск, 2000. – 42 с.
3. Валькман Ю.Р., Быков В.С. О моделировании образного мышления в компьютерных технологиях: общие закономерности мышления. // Сборник научных трудов KDS-2005 20-30 июня 2005 г., т.1., София, 2005, с.37-45.
4. Гладун В.П., Величко В.Ю. Конспектирование естественно-языковых текстов. // Сборник научных трудов KDS-2005 20-30 июня 2005 г., т.2., София, 2005, с.344-347.
5. Леонтьева Н.Н., Семенова С.Ю. Семантический словарь РУСЛАН как инструмент компьютерного понимания**. //** Понимание в коммуникации. Материалы научно-практической конф. 5-6 марта 2003 г. М., МГГИИ, 2003. С.41-46.
6. Жермен-Ликур П., Жорж П., Пистр Ф., Безье П. Математика и САПР: в 2-х кн. Кн.2-М.: Мир, 1988.- 264с.
7. Люгер Дж.Ф. Искусственный интеллект: Стратегии и методы решения сложных проблем. - М.: Изд. дом “Вильямс”. 2003.- 864с.
8. Эгрон Ж. Синтез изображений: Базовые алгоритмы. -М.: Радио и связь, 1993.-216с.
9. Роджерс Д. Алгоритмические основы машинной графики. - М.: Мир, 1989. -512с.

Информация об авторах

Николай Борисович Фесенко *– Ин-т кибернетики им. В.М. Глушкова НАН Украины, Киев-187 ГСП, 03680, просп. акад. Глушкова, 40,e-mail: glad@aduis.kiev.ua*

NEURAL KNOWLEDGE DISCOVERY IN DISTRIBUTED DATABASES BY INTERNET

Adil Timofeev, Pavel Azaletsky

Abstract: *This paper proposes a distributed KDD system allowing remote usage of user data. KDD system based on polynomial neural networks is described. The system is an universal KDD tool as it can build decision-making models in any subject field. Its implementation as a web-service will allow third-party software developers to create specialized applications, which focus on neural knowledge base usage.*

Keywords*: knowledge discovery in databases (KDD); polynomial neural networks; distributed databases*

ACM Classification Keywords*: I.2. Artificial Intelligence*

Introduction

The main task of KDD systems is analysis of data contained in databases with the purpose of discovering hidden, unobvious and unknown patterns and rules. The heterogeneous nature of modern distributed databases significantly complicates the task. Besides, each type of database requires different method of access to its data. Consequently, for each intellectual analysis system it is necessary to develop specific algorithms that take into account its features and characteristics. Modern databases are not only heterogeneous, but redundant as well. For a KDD synthesis, it is advisable to simplify the schema and the knowledge base structure as much as possible.

KDD Technology in Information Processing

Knowledge Discovery in Databases (KDD) is a process of search of meaningful knowledge in raw data [6]. The knowledge could be represented as a set of rules describing relations between data properties (decision trees), frequently encountered models (association rules), and classification results (neural networks) or data clusters (Kohonen maps), to name a few.

Regardless of knowledge representation model used, a KDD process consists of the following stages:

- Data preparation;
- Data preprocessing;
- Data transformation and normalization;
- Data mining;
- Data postprocessing.

Knowledge Discovery in Databases does not determine what data analysis or processing algorithms should be used; instead, it defines the sequence of activities that should be performed in order to extract meaningful knowledge from source data. This approach is universal and does not depend on the application domain.

Neural Approach in KDD

The size and the structure of a network should correspond to the essence of the investigated problem (e.g., in terms of computational complexity). In the works [3-5] authors suggest three-tier polynomial and Diophantine neural networks, multi-tier "many-valued decision tree" neural network, as well as method for transformation of treelike neural networks into polynomial neural networks and for minimization of their complexity.

During the learning process a neural network uses input data to adjust its synaptic weights and to fine tune connections between neural elements. The resulting neural network expresses patterns present in the data and is able to make decisions on new data. The network serves as a functional equivalent for some data dependency model, i.e. a function of input variables, much like a one created using traditional modeling.

Learning Databases Preparation

At this stage qualified users or experts who possess knowledge in a certain application domain and want to automate the decision making process prepare a training set. This set $T = \langle Y, X \rangle$ is composed from known feature values $x_1, x_2, ..., x_n$ and decision function y values defined on an object set Ω:

$$\Omega = \{\omega_k : k = \overline{1,N}\}$$
$$Y = \{y(\omega_k) : k = \overline{1,N}\}$$
$$X = \{(x_1(\omega_k), ..., x_n(\omega_k)) : k = \overline{1,N}\}$$

The set T can be easily represented as a table:

Table 1. Learning database representation

ω	Y	x_1	…	x_n
ω_1	$y(\omega_1)$	$x_1(\omega_1)$	…	$x_n(\omega_1)$
…	…	…	…	…
ω_N	$y(\omega_m)$	$x_1(\omega_m)$	…	$x_n(\omega_m)$

Here $y(\omega_k)$ serves as an expected result of the problem solution on the input set $\{(x_1(\omega_k), x_2(\omega_k)..., x_n(\omega_k)\}$. Training set is a database table with the expected values of decision function in the first column and input data feature values in the rest. The following constraints apply to this database:

Validity: only qualified specialists take part in the creation.

Completeness: all possible solutions are presented in the table.

Consistency: there should not be two different values of decision function for the same input feature value.

These constraints allow to avoid noise influence, inconsistency and incompleteness in data representation.

Data Transformation and Normalization

If a KDD system uses neural networks, an expert (a user with strong knowledge in the application domain) has to fill in the training set. The system will then transform this input into numbers. Some systems use binary encoding of the set elements, i.e. {-1, +1}. If necessary, the training set could be sorted then.

Neural Network Construction and Learning

At this stage, a neural network is constructed and trained using various learning algorithms. As a result, a "neural knowledge base" is created.

Knowledge Base Quality Control

After the neural network is trained, an expert tests its quality using training and test sets (these two sets should not intersect). The test set has the same structure as the training set. Based on the test results the expert makes decision on whether the constructed knowledge base is suitable for the given task or not. If not, the knowledge base construction process should be restarted from the step 1.

Knowledge Base Usage

To solve a task using the constructed knowledge base, a user prepares a table of feature values and sends it to the neural network input. The answer given represents the value of the decision function or the synthesized ANN.

Architecture and Organization of a Distributed Neural KDD System

The problem of development of a distributed system with remote knowledge base creation and usage capabilities is of high importance today. Such system often implements client-server architecture and uses Internet for information transmission.

A distributed neural KDD system operates in two main stages:

- Neural knowledge base creation and control, implemented by an expert (the expert stage);
- Knowledge base usage for the purpose of solving specific user tasks (the user stage).

A knowledge base system based on threshold polynomial neural networks has been developed [3-5]. It uses multiagent technology for processing and transmitting databases (DB) and knowledge bases (KB) through e-mail and HTTP protocols.

In the given model, experts producing a knowledge base and users working with it act as agents. The neural network type in this case can be described with arithmetic (Diophantine) polynomials. It is used for recognition of complicated (linearly non-separable) pattern types defined in a space of either binary or multi-valued feature-predicate. The scheme is illustrated on the Fig. 1.

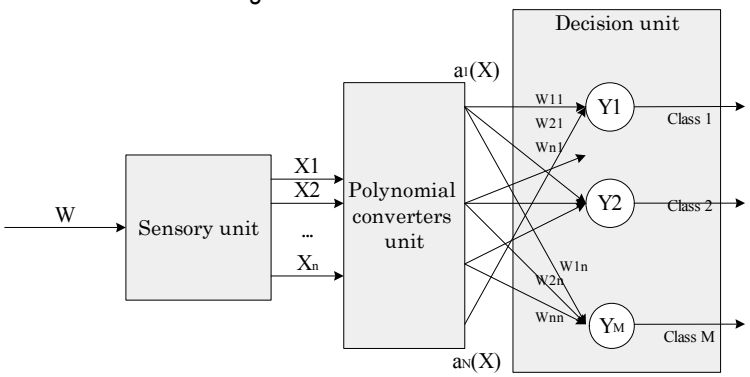


Figure 1. Architecture of a threshold polynomial neural network

At the input of a neural network there is a sensory unit consisting of threshold neuron-like elements, which encodes object features as binary codes $X = |x_1, x_2, ..., x_n|$.

The feature vector (X) is transmitted to a unit of polynomial converters (A-elements), which creates an m-dimensional vector of secondary (polynomial) features $Z = (a_1(X), a_2(X), ...a_N(X))$. These secondary features define a polynomial feature space $a_j(X), j = 1,2,...,n$, a so-called rectifying space. Explicit function form $a_j(X)$ is chosen in accordance with the given task and the training set, i.e. during the process of neural network construction and self-organization [3-5].

The output tier contains solution threshold neuron-like elements:

$$Y_i = sign(\sum_{j=1}^{N} w_j a_j(x)),\ i \in [1...,M].$$

The recurrent learning algorithm used in threshold polynomial neural network is a supervised learning algorithm, which offers a number of advantages over the frequently used back-propagation of error (BPE) [1], including the following:

It is not necessary to determine network structure in advance, since the algorithm adjusts itself during learning process.

This is a single-pass algorithm, i.e. the third layer (decision layer) neurons' weights are adjusted in the first pass through the training set.

The algorithm guarantees error-free classification of elements in the training set.

The algorithm constructs a neural network with a high degree of extrapolation to data beyond the training set.

The synthesized neural network allows creation of a neural knowledge base based on the source database.

The structural scheme of the distributed KDD system is shown on the Figure 2:

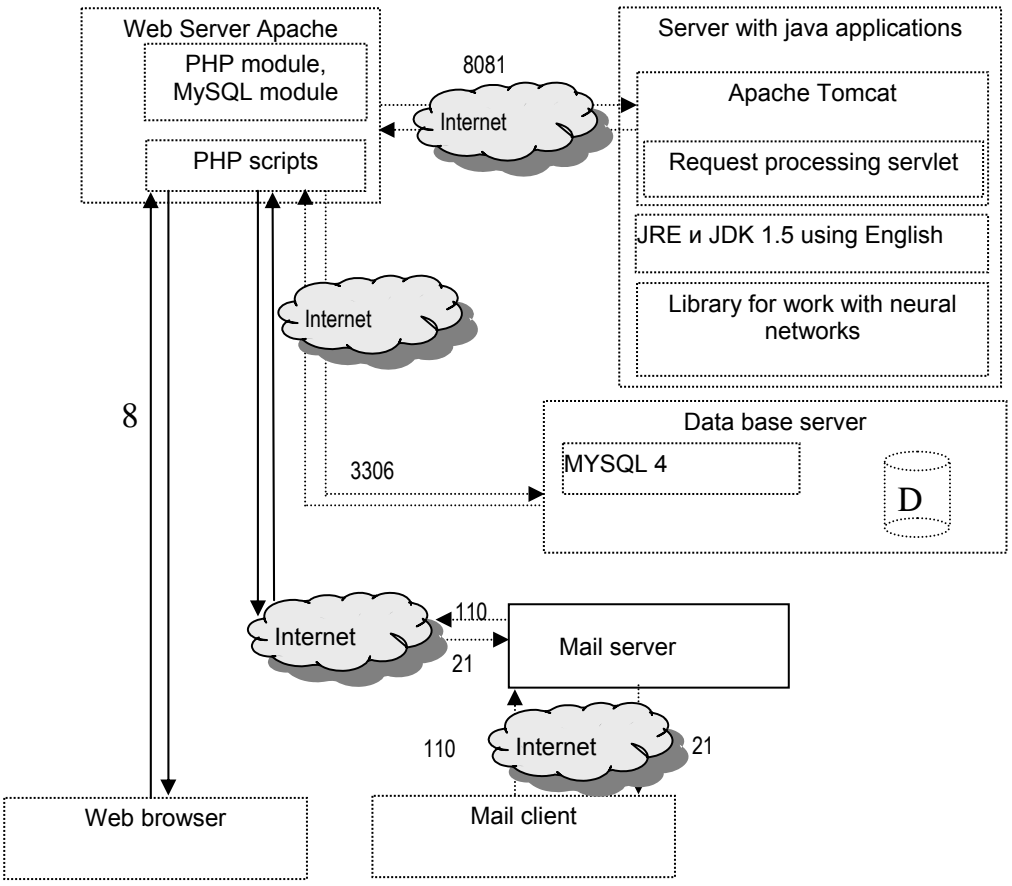


Figure 2. Structural scheme of the distributed neural KDD system.

At the first stage, an expert creates training and test sets in the form of database table. For this task, an expert query is constructed. The second stage concerns work with the knowledge base, at which a user query is constructed. As a reply, the system fills in the "Result" value corresponding to the certain feature cortege, which serves as an output generated by the system in response to the user query.

Conclusion

The distributed KDD system proposed in the article allows remote usage of experience of experts in a certain field, and its implementation as a neural knowledge base. The system is a universal KDD tool, since it makes it possible to build decision-making models in any subject field. Its improvement to a web-service will allow third-party software developers to create specialized applications oriented on neural knowledge base usage. The system can be applied in research as well: as a tool for in-depth study of different effects in ecology, economics and other fields. It serves as means to integrate problem solving experience of geographically distributed users.

Acknowledgements

The work has been done at partial support of RFBR-grant № 05-01-08044-ofi and Program "GRID" of RAS Presidium.

Bibliography

[1]. Lugger, G. F. Artificial Intelligence: Structures and Strategies for Complex Problem Solving, 2003, p. 864.

[2]. Bagresan, A. A., Kuprianov, M. S., Stepanenko, V. V., Holod, I. I. Methods and Models of Data Analyzing: OLAP and Data Mining, 2004, p. 336.

[3]. Simon Haykin, Neural Networks a Comprehensive Foundation, 2006, p. 1104.

[4]. Timofeev A.V. Methods of Creation of Diophantine Neural Networks with Minimal Complexity – RAS Report, 1995, Vol. 345 No.1, pp. 32-35.

[5]. Timofeev A.V. Parallelism and Self-Organization in Polynomial Neural Networks for Image Recognition – Proceedings of the 7th International Conference on Pattern Recognition and Image Analysis: New Information Technologies (18–23 October, 2004, St. Petersburg), pp. 97-100, 2004.

Authors' Information

Timofeev Adil Vasilievich *– Dr. Sc.,Professor, Honoured Scientist of Russian Federation, Saint-Petersburg Institute for Informatics and Automation of Russian Academy of Sciences, 199178, Russia, Saint-Petersburg, 14-th Line, 39, phone: +7-812-328-0421; fax: +7-812-328-4450, e-mail: tav@iias.spb.su*

Azaletsky Pavel *– Post Graduate Student, Saint-Petersburg State University of Aerospace Instrumentation, 190000, Russia, Saint-Petersburg, Bolshaya Morskaya, 67, phone: +7-812-328-0421; fax: +7-812-328-4450, e-mail: eaglenk2@mail.ru*

NEURAL NETWORK BASED OPTIMAL CONTROL WITH CONSTRAINTS

Daniela Toshkova, Georgi Toshkov, Todorka Kovacheva

Abstract*: In the present paper the problems of the optimal control of systems when constraints are imposed on the control is considered. The optimality conditions are given in the form of Pontryagin's maximum principle. The obtained piecewise linear function is approximated by using feedforward neural network. A numericak example is given.*

Keywords*: optimal control, constraints, neural networks*

Introduction

The optimal control problem with constraints is usually solved by applying Pontryagin's maximum principle. As is known the optimal control solution can be obtained computationally. Even in the cases when it is possible an analytical expression for optimal control function to be found, the form of this function is quite complex. Because of that reason the possibilities of using neural networks for solving the optimal control problem are studied in the present paper.

The ability of neural networks to approximate nonlinear function is central to their use in control. Therefore it can be effectively utilized to represent the regulator nonlinearity. Other advantages are their robustness, parallel architecture.

Lately, different approaches are proposed in the literature treating the problem of constrained optimal control for using neural networks. In [Ahmed 1998] a multilayered feedforward neural network is employed as a controller. The training of the neural network is realized on the basis of the so called concept of Block Partial Derivatives. In [Lewis 2002] a closed form solution of the optimal control problem with constraints is obtained solving the associate Hamilton-Jacobi-Bellman (HJB) equation. The solution of the value function of HJB equation is approximated by using neural networks.

In the present paper the problem of finding the optimal control with constraints is considered. A numerical example is given.

Problem Statement

The control system, described by following differential equations is considered:

$$\frac{dx_i}{dt} = \sum_{j=1}^{n} a_{ij} x_i + b_i u \qquad (i = 1, 2, \ldots, n) \tag{1}$$

where x_j are phase coordinates of the system, function u describes the control action and a_{ij} are constant coefficients. The admissible control u belonging to the set U of piecewise linear functions is constrained by the condition

$$|u(t)| \le 1 \tag{2}$$

Following problem for finding the optimal control is formulated. To find such a control function $u(x_1,\ldots, x_n)$ for the system (1) among all the admissible controls that the corresponding trajectory $(x_1(t),\ldots, x_2(t))$ of the system (1) starting from any initial state $(x_1(0),\ldots, x_2(0))$ to tend to zero at $t \to \infty$ and the performance index

$$J = \int_0^{\infty} \left(\sum_{i=1}^{n} q_i x_i^2 + ru^2 \right) dt \tag{3}$$

to be converging and to take its smallest possible value. The coefficient qi and r are positive weight constants.

Optimality Conditions

The notation is introduced [Pontryagin 1983]:

$$f_0(x_1, \ldots, x_n, u) = \sum_{j=1}^{n} q_j x_j^2 + ru^2 \tag{4}$$

$$f_i(x_1, \ldots, x_n, u) = \sum_{j=1}^{n} a_{ij} x_j^2 + b_i u \qquad (i = 1,\ldots,n) \tag{5}$$

One more variable θ_0 is added to the state variables $(x_1,\ldots, x_n)$ of the system (1) [Chjan 1961]. It is a solution of the following equation

$$\frac{dx_0}{dt} = f_0(x_1, \ldots, x_n, u) \tag{6}$$

and initial condition $x_0(0) = 0$. Then the quantity J according to (9) becomes equal to the boundary of x(t) when $t \to \infty$. The system of differential equation, which are adjoint to the system (7) is composed with new variables $\Psi=\{\Psi_0, \Psi_1, ..., \Psi_n\}$:

$$\frac{d\Psi_0}{dt} = -\sum_{\alpha=0}^{n} \frac{\partial f_\alpha}{\partial x_0} \Psi_\alpha = 0 \tag{7}$$

$$\frac{d\Psi_i}{dt} = -\sum_{\alpha=0}^{n} \frac{\partial f_\alpha}{\partial x_0} \Psi_\alpha = -2q_i\theta_i - \sum_{j=1}^{n} a_{ij}\Psi_j \qquad (i = 1,...,n) \tag{8}$$

After that the Hamilton function is composed:

$$H(\theta,\Psi,u) = \sum_{\alpha=0}^{n} \Psi_\alpha \frac{dx_\alpha}{dt} = \sum_{\alpha=0}^{n} \Psi_\alpha f_\alpha(x_1,...x_n,u) = \Psi_0\left(\sum_{i=1}^{n} q_i x_i^2 + ru^2\right) + \sum_{i=1}^{n} \Psi_i\left(\sum_{j=1}^{n} a_{ij}x_j + b_i u\right) \tag{9}$$

In the right-hand side of Eq. (9) the quantity *u* is contained in the expression

$$H_1 = r\Psi_0(t)u^2(t) + u(t)\sum_{i=1}^{n} b_i\Psi_i(t) \tag{10}$$

Because of that the condition for maximum of H coincide with the condition

$$\begin{aligned} \max_{|u|\le 1} H_1 &= \max_{|u|\le 1}\left[r\Psi_0(t)u^2(t) + u(t)\sum_{i=1}^{n} b_i\Psi_i(t)\right] = \\ &= \max_{|u|\le 1}\left\{ r\Psi_0(t)\left[u(t) + \frac{1}{2r\Psi_0}\sum_{i=1}^{n} b_i\Psi_i(t)\right]^2 - \frac{1}{4r\Psi_0}\left[\sum_{i=1}^{n} b_i\Psi_i(t)\right]^2\right\} \end{aligned} \tag{11}$$

Having in mind condition (7) the quantity Ψ_0 is a constant. As its value can be any negative number it is set to $\Psi_0 = -1$.

After placing this value in Eq. (11) the maximum of the expression in the square brackets will be reached when the first negative addend becomes zero if it is possible or takes its minimal absolute value. The expression

$$\left[u(t) - \frac{1}{2r}\sum_{i=1}^{n} b_i\Psi_i(t)\right]^2 \tag{12}$$

will take its minimal absolute value if on condition $|u| \le 1$ a value of the following kind is chosen for u

$$u(t) = \begin{cases} \frac{1}{2r}\sum_{i=1}^{n} b_i\Psi_i & \text{at} \quad \left|\frac{1}{2r}\sum_{i=1}^{n} b_i\Psi_i\right| \le 1 \\ 1 & \text{at} \quad \left|\frac{1}{2r}\sum_{i=1}^{n} b_i\Psi_i\right| \ge 1 \\ -1 & \text{at} \quad \left|\frac{1}{2r}\sum_{i=1}^{n} b_i\Psi_i\right| \le -1 \end{cases} \tag{13}$$

The values of $\psi_c(t)$ can be determined if the adjoint equations (7), (8) are solved. This leads to the requirement the initial values of $\psi_0(0)$ to be found beforehand.

First u(t) is assumed not to reach its boundary values. Then after placing the upper expression from (13) instead of u(t) in Eqs. (1), (7) и (8) one obtains

$$\begin{aligned} \frac{dx_i}{dt} &= \sum_{j=1}^{n} a_{ij}x_j + \frac{b_i}{2r}\sum_{j=1}^{n} b_j\Psi_j \quad (i = 1,...,n) \\ \frac{d\Psi_i}{dt} &= 2q_i x_i - \sum_{j=1}^{n} a_{ji}\Psi_j \end{aligned} \tag{14}$$

This system of equations has to be solved with the initial conditions $x_1(0),\ldots, x_n(0)$ as well as with the final (boundary) conditions

$$\lim_{t\to\infty} x_1(t) = \lim_{t\to\infty} x_2(t) = \ldots = \lim_{t\to\infty} x_n(t) = 0 \tag{15}$$

It is necessary the appropriate initial conditions $\psi_1(0), \ldots, \psi_n(0)$ to be selected in such a way that the initial and the final conditions for $x_1(t),\ldots, x_n(t)$ to be satisfied.

The relationship between $x_i(0)$ and $\psi_i(0)$ has the following form [4]:

$$\Psi_i(0) = \sum_{j=1}^{n} \chi_{ij} x_i(0) \qquad (i = 1, \ldots, n) \tag{16}$$

These relationships have to be kept in any time, for which one can always assume to be the initial one. Therefore the optimal control *u* within the boundaries is determined and it has the following form:

$$u = \frac{1}{2r}\sum_{i=1}^{n} k_i x_i \tag{17}$$

where $k_i = \sum_{j=1}^{n} b_j \chi_{ji}$

The expression (17) holds only in the cases when the absolute value of the sum $\frac{1}{2r}(k_1x_1 + \ldots + k_nx_n)$ is not greater than one. When $\left|\frac{1}{2r}(k_1x_1 + \ldots + k_nx_n)\right| > 1$ the optimal control passes on the boundary i.e. $|u| = 1$, if the right hand boundary conditions are satisfied i.e. the solution of the system (1), which became nonlinear in connection to the nonlinear relationship between *u* and $x_1,\ldots, x_n$, tends to zero. In other words the solution of the system has to be asymptotically stable. Thus the optimal control is defined by the expression

$$u(t) = \begin{cases} \frac{1}{2r}\sum_{i=1}^{n} k_i x_i & \text{at} \quad \left|\frac{1}{2r}\sum_{i=1}^{n} k_i x_i\right| \le 1 \\ 1 & \text{at} \quad \frac{1}{2r}\sum_{i=1}^{n} k_i x_i \ge 1 \\ -1 & \text{at} \quad \frac{1}{2r}\sum_{i=1}^{n} k_i x_i \le -1 \end{cases} \tag{18}$$

Structure and Training of the Neural Network

For the control function realization a feed forward neural network with one hidden layer is used. Thus the necessity of solving a large number of equations for determining the coefficients k_i drops off.

The neural network consists of three layers – an input, output and hidden one. The input and hidden layers have five neurons and the output layer – one. The activation function of the output neuron is piecewise linear. The neural network output is

$$y = \begin{cases} +1 & \varphi(\nu) \ge 1 \\ \varphi(\nu) & |\varphi(\nu)| \le 1 \\ -1 & \varphi(\nu) \le 1 \end{cases} \tag{19}$$

where $\nu = w^T z$. The neural network input is denoted z and w is the neural network weight. The neural network output represents the control u, x – the state vector and weights are the coefficient k.

The neural network is trained according to the back-propagation algorithm. Let the training sample $\{z(n), d(n)\}_{n=1}^{N}$ be given where z(n) are the system states and d(n) is the corresponding control, which are known preliminarily. The neural network is trained according to the back-propagation algorithm [Haykin 1999].

Simulation Results

In order to verify the suggested approach for solving the optimal control problem following system is considered:

$$\frac{dx_1}{dt} = x_2$$

$$\frac{dx_2}{dt} = -x_1 - 2x_2 + u$$

and the control is constrained by

$$|u| \le 1$$

The performance index to be minimized is of the form:

$$\int_0^\infty [x_1^2(t) + x_2^2(t) + u^2(t)]dt$$

The problem is solved by using Pontryagins principle and neural networks. The results, which are obtained by both approaches, are compared. In Fig. 1 the optimal control, obtained by using neural networks is shown. Fig. 2 depicts the corresponding states trajectory. In Fig. 3 and Fig. 4 the optimal control, obtained by applying the maximum principle and the corresponding trajectory are given respectively. By 1 and 2 are denoted x_1 and x_2 respectively.

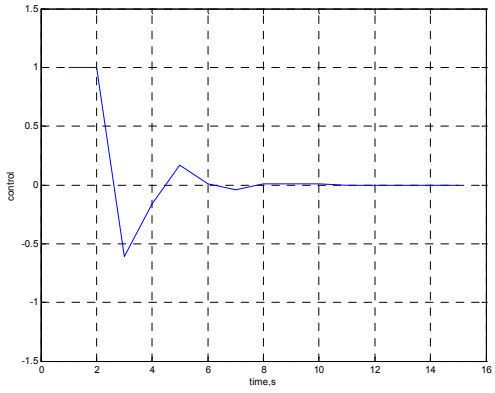


Fig. 1. Optimal control, obtained by using the suggested neural network based approach

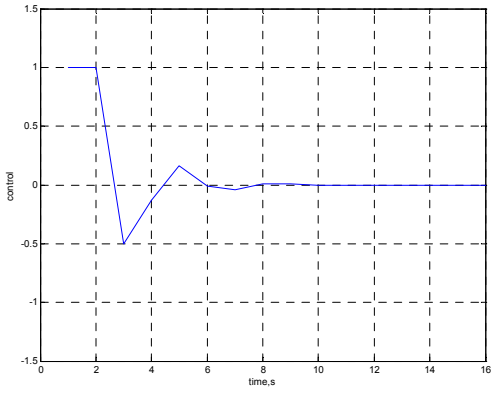


Fig. 2. Optimal control, obtained by applying the maximum principle

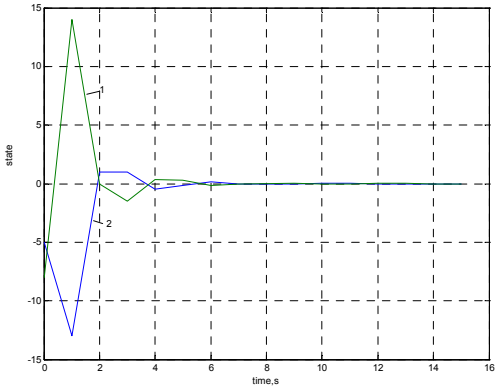


Fig.3 Optimal trajectory
(neural network based approach)

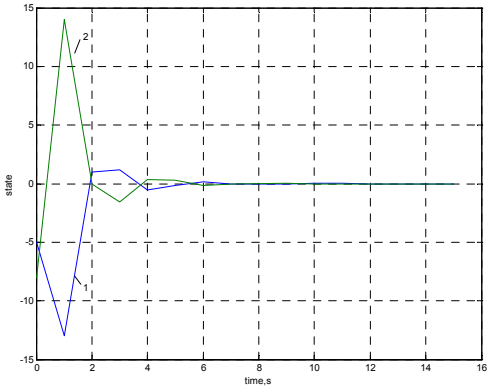


Fig.4 Optimal trajectory
(Pontryagin's maximum principle)

Conclusion

In the present paper an approach for optimal constrained control based on using of neural networks is suggested. On the basis of the simulation experiments one can say that the proposed approach for optimal control is accurate enough for the engineering practice. The suggested approach can be applied for optimal control in real time, where the control is constrained.

Bibliography

[Ahmed 1998] M.S. Ahmed and M.A. Al-Djani. Neural regulator design. Neural Networks, Vol. 11, pp. 1695-1709, 1998

[Chjan 1961] Zh.-V.Chjan. A problem of optimal system synthesis through the maximum principle. Automatization and telemechanics, Vol. XXII, No.1, 1961, pp. 8-12

[Haykin 1999] S. Haykin. Neural Networks: A Comprehensive Foundation., 1999, 2nd ed, Macmillan College Publishing Company, New York

[Lewis 2002] F.L. Lewis and M. Abu-Khalaf. A Hamilton-Jacobi setup for constrained neural network control. Proceedings of the 2003 IEEE International Symposium on Intelligent Control Houston, Texas *October 5-8. 2003

[Pontryagin 1983] L.S. Pontryagin, V.G. Boltyanskiy, R.V. Gamkrelidze and E.F.Mishtenko. Methematical theory of the optimal processes. Nauka, Moskow, 1983, in Russian

Authors' Information

Georgi Toshkov – *Technical University of Varna, 1, Studentska Str, Varna, 9010, Bulgaria,*
e-mail: g_toshkov2006@abv.bg

Daniela Toshkova *– Technical University of Varna, 1, Studentska Str, Varna, 9010, Bulgaria,*
e-mail: daniela_toshkova@abv.bg

Todorka Kovacheva *– Economical University of Varna, Kniaz Boris Str, Bulgaria,*
e-mail: todorka_kovacheva@yahoo.com

GENERALIZING OF NEURAL NETS: FUNCTIONAL NETS OF SPECIAL TYPE

Volodymyr Donchenko, Mykola Kirichenko, Yuriy Krivonos

Abstract*: Special generalizing for the artificial neural nets: so called RFT – FN – is under discussion in the report. Such refinement touch upon the constituent elements for the conception of artificial neural network, namely, the choice of main primary functional elements in the net, the way to connect them(topology) and the structure of the net as a whole. As to the last, the structure of the functional net proposed is determined dynamically just in the constructing the net by itself by the special recurrent procedure. The number of newly joining primary functional elements, the topology of its connecting and tuning of the primary elements is the content of the each recurrent step. The procedure is terminated under fulfilling "natural" criteria relating residuals for example. The functional proposed can be used in solving the approximation problem for the functions, represented by its observations, for classifying and clustering, pattern recognition, etc. Recurrent procedure provide for the versatile optimizing possibilities: as on the each step of the procedure and wholly: by the choice of the newly joining elements, topology, by the affine transformations if input and intermediate coordinate as well as by its nonlinear coordinate wise transformations. All considerations are essentially based, constructively and evidently represented by the means of the Generalized Inverse.*

Keywords*: Artificial neural network, approximating problem, beam dynamics with delay, optimization.*

Introduction

Artificial neurons nets are the technological elements, used in various applications ([Amit, 2002], [Veelenturf, 1995] for example) especially under model uncertainty. It may be approximation problem for the functions represented by its observations, the task of the control, classification problem and so on.

The power of the artificial neural nets (ArtNN) substantially specified by the using virtually the composition of the functions [Donchenko, Kirichenko, Serbaev, 2004]. The might of this instrument was certified by [Kolmogorov, 1966] and [Arnold, 1967].

But some faults of the ArtNN are obvious, namely, the constraints on the primary functional elements represented by neurons, constraints on topology, constraints on the optimizations, properly in the appearance of the Back Propagation.

In the report the attempt is undertaking to extend the ArtNN up to Functional nets taking as the primary the special functional element – so called ERRT: elementary recursive regressive transformation, spreading the topology and introducing "natural" optimizing parameter.

The fulcrum for the implementing this attempt arises from the theory of Generalized Inverse Matrixes (for ex. [Алберт, 1976]) developed by one of the authors and represented in [Кириченко, Лепеха, 2002].

The approach proposed is realized in the conception of so called Recursive nonlinear Functional Transformation - Functional Net (RFT-FN. The idea of such type transformations in the variant of inverse recursion was proposed in [Кириченко, Крак, Полищук, 2004]], focused on the optimizing procedure for ERRT. We represent here another variant of RFT - FN – the variant of so called inverse recursion, introduced and discussed in the paper [Donchenko, Kirichenko, 2005] by the authors. As it has been already noticed earlier, RFT – FN embodies the main ideas of the classical ArtNN: using the composition of standard primary functional elements (artificial neurons – a.n.) connected according some topology for constructing a complex object. The primary elements of the composition: a.n. – represents mathematically comparatively simple function: a composition of linear function – linear functional for input multi- dimensional variable, – and simple standard scalar function. Conception of the RFT-FN leaves the main idea of ArtNN unchangeable: that the idea of constructing a complex objects through a composition of standard simple ones. But it is proposed and implemented some substantial refinements. The main of them are actually in the next:

- Expanding the domain of the feasible functions for the basis standard elements and introducing the special procedure for the adjusting for the newly connecting ERRT;

Introducing the special recursive dynamic procedure for the constructing the RFT-FN. The term "dynamic" means the nor a priory structure is fixed for the RFT-FN, but it determines exclusively by the quality of the constructions, characterized by some "natural" functional of the quality, the residuals in the approximation for example. So the termination of the recursive procedure of the constructing the RFT-FN determines dynamically in the in the course of the procedure by itself.

- Expanding the variants of the feasible connections for the newly connecting elements and its number.

There some more additional enhancements for the ArtNN, proposed and implemented in the conception of the RFT – FN. These are: coordinate wise nonlinear transformations of the input and intermediate inner coordinate along with the linear transformations of them.

All the refinements make it possible to optimize the constructing of the RFT-FN on the each recursive step and wholly by the next ways:

- By the choice of the feasible functions for the ERRT, by the choice of linear transformations of its coordinates and coordinate wise nonlinear transformations of the coordinate for the input variables for the newly connected ERRT: single or a number of them;
- By the choice of a topology for newly connected elements. There are three mains types of the connections for the newly connecting elements: parallel by input (parinput), parallel by output (paroutput) or sequential (seq).

The implementation of the RFT–FN conception will be demonstrated on the approximation problem for the functions, represented by its observations. Draw the attention of the reader that this problem is represented widely in the publications as in deterministic as well in statistical enunciations. As regarding the last, we refer to [Линник, 1962], [Вапник, 1979], [Ивахненко, 1969].

1. General Conception of the Functional Net: RFT -FN

The RFT-FN constructing procedure conclude in joining the recurrently EERT to RFT-FN already has been constructed during previous steps in compliance with one of the three certain types of connecting,

These types will be denoted as parinput, paroutput and seq. The connecting types correspond to natural transformations of input signal: parallel by input or output, and also sequential.

1.1. Description of the Primary Functional Elements: ERRT

The basic constructive element for RFT-FN is the ERRP-element [9], which is defined as a map from R^{n-1} in R^m designed according to the next form:

$$y= A_+ \Psi_u \left(C \begin{pmatrix} x \\ 1 \end{pmatrix} \right), \quad (1)$$

It approximates the dependence, represented by the learning sample

$(x_1^{(0)}, y_1^0), \ldots, (x_M^{(0)}, y_M^{(0)})$, $x_i^{(0)} \in R^{n-1}$, $y_i^{(0)} \in R^m$, $i = \overline{1, M}$, where:

C is (n×n))-matrix, defining an affine map between R^{n-1} and R^m, fixed when synthesizing the ERRT;

Ψ_u – coordinate-wise nonlinear map from R^n in R^n; each of nonlinear real functions u_i, $i = \overline{1, n}$, transforming the coordinates, belongs to finite set $\Im$ of the functions. This set is fixed, but open for extension. The set $\Im$ include also identity function. We will consider each of such functions to be smoothly enough; one chose the nonlinear map Ψ to minimize the discrepancy between input and output on the learning sample;

$A+$– trace-norm minimal solution of the next matrix equation

$$AX_{\Psi_u C} = \mathrm{Y}, \quad (2)$$

in which $X_{\Psi_u C}$ -matrix assembled from vectors-columns $\Psi_u(C \begin{pmatrix} x_i^{(0)} \\ 1 \end{pmatrix}) = \Psi_u(z_i^{(0)})$

and matrix Y – from $y_i^{(0)}$, $i = \overline{1, M}$

1.2. Recursive Procedure and Topology of Connections in RFT – FN Constructing

Composition-recursion in function-building procedure in the proposed below variant of direct recursion will be considered in generalized version. In such version the number $\mathrm{k_m}$ of newly joined ERRT may be more, than one.

Total number of ERRP used will be denoted by T: $T = \sum_{m=1}^{N} k_m$, N being the number of recursive calls of the procedure.

Direct recursion represented by the next figures and corresponding equations depending on type of the joining.

- Parinput (see fig.1)

The input-output equations represented such type of topology in the recursive procedure are tot the next form:

$$\begin{aligned} & x(i+j) = A_{+i+j-1} \Psi_{u_{i+j-1}} (C_{i+j-1} x(i)), \\ & \hat{y}(i+j) = \hat{y}(i+j-1) + A_{+i+j-1} \Psi_{u_{i+j-1}} (C_{i+j-1} \cdot x(i)), \\ & i = \sum_{l=1}^{m} k_l, \; j = \overline{1, k_{m+1}} \end{aligned} \quad (3)$$

- Paroutput (see fig. 2)

Paroutput topology in the recurrent step is represented by the chart, presented in fig.2 and by the next input-output equations:

$$x(i+j) = A_{+i+j-1}\Psi_{u_{i+j-1}}(C_{i+j-1}x(i+j-1)),$$
$$\hat{y}(i+j) = \hat{y}(i+j-1) + A_{+i+j-1}\Psi_{u_{i+j-1}}(C_{i+j-1}\cdot x(i+j-1)), \quad (4)$$
$$i = \sum_{l=1}^{m} k_l,\ j = \overline{1, k_{m+1}}.$$

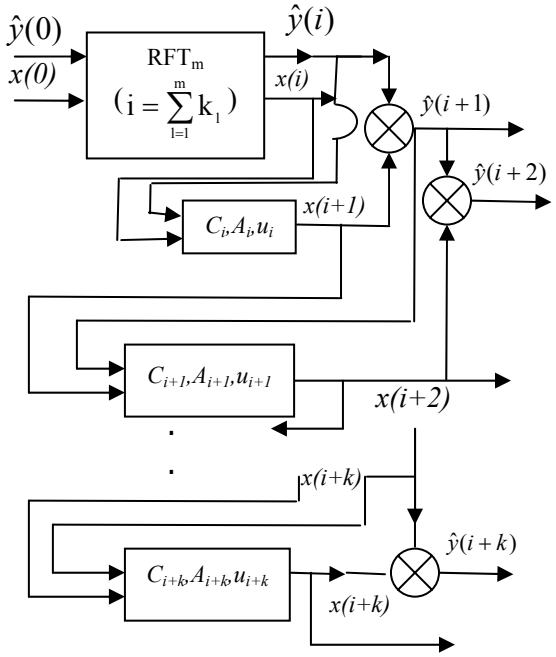


Fig. 1. Parinput

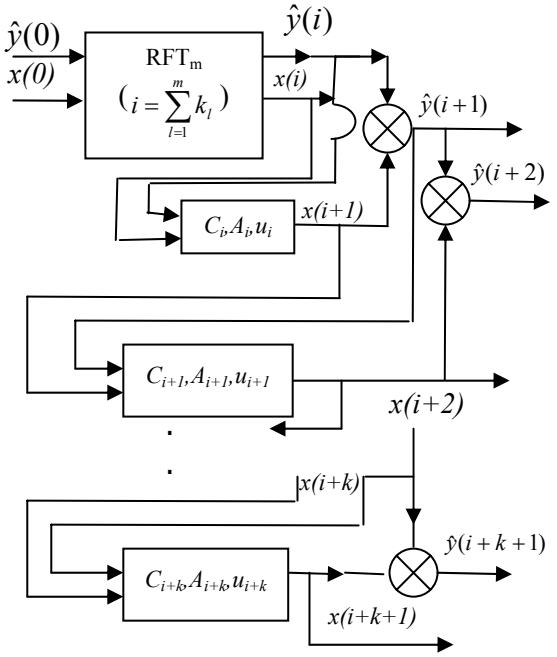
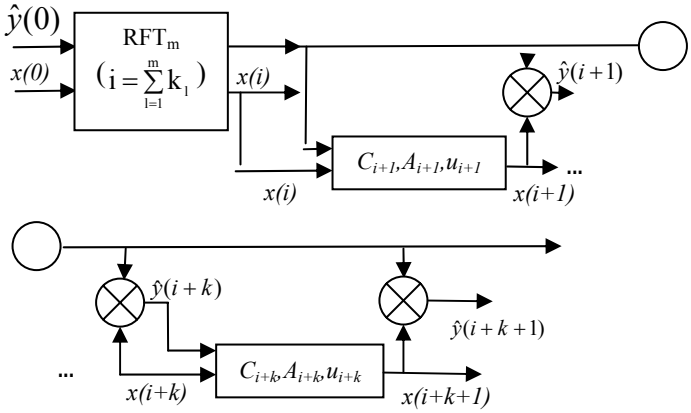


Fig. 2. Paroutput

And at last for sequential type we have correspondingly:

- Seq



$$x(i+j) = A_{+i+j-1}\Psi_{u_{i+j-1}}(C_{i+j-1}x(i+j-1)),$$
$$\hat{y}(i+j) = \hat{y}(i) + A_{+i+j-1}\Psi_{u_{i+j-1}}(C_{i+j-1}\cdot x(i+j-1)), \quad (5)$$
$$i = \sum_{l=1}^{m} k_l,\ j = \overline{1, k_{m+1}}$$

2. Special Class of Beam Dynamics with Delay

The optimization for RFT – FN as it is follows from (3)-(5)is reduced virtually to solving the optimization problem for the beam dynamics of special type, determined below. Namely, we will introduce and consider two classes of special discrete dynamic systems with delay named below simple and combined. Classical results about conjugate systems and Hamilton functions will be extended on the systems introduced as well as the results about functional differentiating respectively controls.

2.1. Special Class of Beam Dynamics with Delay: Basic Definitions

These two types of beam dynamics: with simple delay and combined – are defined in the next way:

- simple delay:

$$x(j+1) = f(x(j-s(j)),u(j),j)\ , \tag{6}$$

combined:

$$x(j+1) = f(x(j),x(j-s(j)),u(j),j)\ ;\quad j=\overline{0,N-1}\,; \tag{7}$$

set of the initial states $\mathrm{Ini}=\left\{x_1^{(0)},...,x_M^{(0)}\right\}$: $x(0)=x^{(0)}\in \mathrm{Ini}$;

- functional on the set of the trajectories

$$I(U)=\sum_{\mathrm{x}^{(0)}\in Ini}\Phi(x(\mathrm{N}))\,, \tag{8}$$

delay function $s(j)\in\{2,...,j\}$, $s(0)=0, s(j)\in\{2,...,j\}$, $j=\overline{0,N-1}$.

Simple systems are defined by the collection of the functions $f(z,\ u,\ j)$, $j=\overline{0,N-1}$ and combined ones – by the $f(z,\ v,\ u,j)$, $j=\overline{0,N-1}$.

2.2. Conjugate Systems and Hamilton Function

Given the system dynamics with delay: simple or combined – define the conjugate systems and the Hamilton functions depending on the type of the delay beam dynamics.

Simple delay:

- Conjugate system $p(k),\ k=\overline{N,0}$

$$p(N)=-grad_{x(N)}\Phi(x(N))\,,$$

$$p(k)=\sum_{j\in J(k)} grad_{x(k)}\{p^T(j+1)f(x(j),u(j),j)\}\,,$$

$$\mathrm{J(k)}=\left\{j:\ j-s(j)=k,\ \mathrm{j}\ge k\right\},\quad k=\overline{N-1,0}\,;$$

- Hamilton function:

$$H(p(k+1),x(k-s(k)),u(k),k)=\ p^T(k+1)f(x(k-s(k)),u(k),k)\,,\quad k=\overline{N-1,0}\,.$$

Combined delay:

- Conjugate system $p(k),\ k=\overline{N,0}$

$$p(N)=-grad_{x(N)}\Phi(x(N))\,,$$

$$p(k)=grad_z\{p^T(j+1)f(x(k),x(k-s(k)),u(j),j)+\sum_{j\in J(k)} grad_{\ v}\{p^T(j+1)f(x(k),x(j),u(j),j)\}\,,$$

J(k) is the same as for simple systems, $k=\overline{N-1,0}\,;$

- Hamilton function:

$$H(p(k+1),x(k),x(k-s(k)),u(k),k)=\ p^T(k+1)f(x(k),x(k-s(k)),u(k),k)\,.$$

2.3. Gradient in Beam Dynamics with Delay

The classical results take place for the beam dynamics with delay within the classical assumptions as to $f(z,u,j)$ or $f(z,v,u,j)$, $j=\overline{0,N-1}$ and Φ. These results are captured in the next two theorems.

Theorem 1. For the simple delay beam dynamics gradients respectively controls are represented by the next relations

$$grad_{u(k)} I(U) = -\, grad_{u(k)} \sum_{i=1}^{M} H^{(i)}(p^{(i)}(k+1)x^{(i)}(k),u(k),k)\,, \quad k=\overline{0,N-1}\,.$$

Theorem 2. For the combined delay beam dynamics gradients respectively controls are represented by the next relations

$$grad_{u(k)} I(U) = -\sum_{i=1}^{M} \mathrm{grad}_{\,u(k)} \mathrm{H}^{(i)}(\mathrm{p}^{(i)}(\mathrm{k}+1), \mathrm{x}^{(i)}(\mathrm{k}), \mathrm{x}^{(i)}(\mathrm{k}-\mathrm{s}(\mathrm{k}), \mathrm{u}(\mathrm{k}), \mathrm{k}))\,, \quad k=\overline{0,N-1}\,.$$

Index i: $i=\overline{1,M}$ corresponds to trajectories with initial states $x_i^{(0)} \in Ini$.

3. Functional Nets and Beam Dynamics with Delay

Combined delay beam dynamics are very important regarding their role in representation of RFT-FN constructions.

Theorem 3. RFT-FN – predictor with the direct N recursions in using $\mathrm{k_m}, \mathrm{m}=\overline{1,\mathrm{N}}$ ERRT respectively can be represented by the combined delay beam dynamics on the time interval $\overline{0,\mathrm{T}}$, $\mathrm{T}=\sum_{\mathrm{l}=1}^{\mathrm{N}} \mathrm{k_l}$. The elements of such representations are constructive, depending on the type of the joining. The quality functional of the system is of the next form

$$I(C) = \sum_{k=1}^{M} \| y_k^{(0)} - z_2(T) \|^2\,,$$

where $z_2(T)$ is one of two output components for the beam dynamics.

4. Functional Nets Optimal Design

Theorem 3 enables to choose optimally $C_0, C_1, ..., C_{T-1}$ for RFT – FN.

Theorem 4. Under assumption that any element from $\Im$ has the continuous second-order derivation, an RFT-FN is feasible to be constructively optimized by gradient methods respectively matrixes C.

Conclusion

Special kind of the "functional nets": so called RFT – FN, generalizing classical functional nets, namely, artificial neural nets, has been proposed and investigated in the report. The RFT – FN permit multilevel optimization. First level optimization is the optimization in the primary functional element and the pseudo inverse is principally in this stage. Another principal level of the optimization is the optimization due to optimization in the beam dynamics.

Bibliography

[Amit, 2002] Y. Amit., 2D Object Detection and Recognution: models, algorithms and networks.–The MIT Press, Cambridge, Massachusetts.– 2002.–306 p.

[Veelenturf,1995] L.P.J.Veelenturf, Analysis and applications of artificial Neural Networks.–Prentice Hall (UK).– 1995.–259 p.

[Donchenko, Kirichenko, Serbaev,2004] V.S. Donchenko, N.F.Kirichenko,D.P.Serbaev. Recursive regression transformations and dynamical systems.// Proceedings of the Seventh International Conference "Computer Data Analysis and modelling: robustness and computer intensive methods".– V.1.–September 6-10, 2004. – Minsk (Belarus).– p.147-151.

[Колмогоров,1956] А.Н.Колмогоров О представлении непрерывных функций нескольких переменных суперпозициями непрерывных функций меньшего числа переменных.// Доклады АН СССР. – 1956. – Т. 108. – № 2. – С. 179- 182.

[Арнольд,1957.] В.И. Арнольд О функциях трех переменных.// Докл. АН СССР. – 1957. – Т. 114. – № 4. – С. 953- 956.

[Алберт,1977] А.Алберт Регрессия, псевдоинверсия, рекуррентное оценивание.– Пер.с англ.– М.: Наука.–1977.–305с.

[Кириченко,Лепеха,2002] Н.Ф. Кириченко, Н.П.Лепеха Применение псевдообратных и проекционных матриц к исследованию задач управления, наблюдения и идентификации.// Кибернетика и системный анализ. – 2002. – № 4. – с. 107-124.

[Кириченко, Крак, Полищук,2004] Н.Ф.Кириченко, Ю.В. Крак.А.А. Полищук Псевдообратные и проекционные матрицы в задачах синтеза функциональных преобразователей.// Кибернетика и системный анализ –2004.–№3.

[Donchenko, Kirichenko,2005.] V.S Donchenko, N.F. Kirichenko Generalized Inverse in Control with the constraints.// Cybernetics and System Analysis.– v.39 (6) November-December– 2003.–p.854-861(USA).

[Линник,1962] Ю.В.Линник Метод наименьших квадратов и основы математико-статистической теории обработки наблюдений. Изд. 2-е. – М.:Физматгиз.– 1962.– 349 с.

[Вапник,1979] В.Н. Вапник. Восстановление зависимостей по эмпирическим данным. – М.: Наука. – 1979. – 447 с.

[Ивахненко,1969] А.Г. Ивахненко Самоорганизующиеся системы распознавания и автоматического управления. – К.: Техника. – 1969. – 395 с.

Authors' Information

Volodymyr Donchenko – *Kyiv National Taras Shevchenko University (Ukraine), Professor, e-mail: voldon@unicyb.kiev.ua*

Mykola Kirichenko – *Institute of the Cybernetics, National Academy of Sciences (Ukraine), Professor*

Yuriy Krivonos – *Institute of the Cybernetics, National Academy of Sciences (Ukraine), Member Correspondent of the National Academy of Sciences.*

СРАВНИТЕЛЬНЫЙ АНАЛИЗ НЕЧЕТКИХ НЕЙРОННЫХ СЕТЕЙ С РАЗЛИЧНЫМИ АЛГОРИТМАМИ ВЫВОДА В ЗАДАЧАХ ПРОГНОЗИРОВАНИЯ КУРСОВ АКЦИЙ

Юрий Зайченко, Юрий Келестин, Севаее Фатма

Abstract*: The fuzzy neural networks (FNN) with different inference algorithms of Mamdani, Tsukamoto and Sugeno are considered in this paper. The learning algorithm of gradient type for Mamdani and Tsukamoto FNN is described and investigated. The application of FNN with different inference algorithms and membership functions for stocks prices forecasting was carried out and presented and their efficiency estimated.*

Keywords: *fuzzy neural networks, learning algorithms, stock prices, forecasting*

Введение

В последние годы появилось достаточно большое число публикаций, посвященных исследованиям систем с нечеткой логикой и нечетких нейронных сетей (ННС) в задачах управления, классификации и распознавания образов [1,2,3,7,8]. Их основными достоинствами по сравнению с обыкновенными ННС являются возможность работы с неполными и неопределенными данными, возможность учета знаний экспертов в виде нечетких предикатных правил вывода, «если-то». Появились также работы, посвященные исследованию ННС в задачах прогнозирования в экономике. Так, в работе [4] проведено исследование нечетких контроллеров с выводом Мамдани и Цукамото, в задачах макроєкономического прогнозирования, с треугольными функциями принадлежности. В работах [5,6] проведено исследование ННС ANFIS с выводом Сугено в задачах прогнозирования. Цель настоящей работы состоит в проведении сравнительного анализа ННС с различными алгоритмами нечеткого вывода и функциями принадлежности в задачах прогнозирования финансовых рынков с целью определения наиболее адекватного метода для класса задач прогнозирования состояния финансовых рынков, в частности, курсов акций..

Алгоритмы нечеткого логического вывода

Рассмотрим следующие наиболее употребительные алгоритмы нечеткого вывода, считая, для простоты, что базу знаний организуют два нечетких правила вида:

$$\Pi_1: \text{если } x \text{ есть } A_1 \text{ и } y \text{ есть } B_1, \text{ то } z \text{ есть } C_1$$

$$\Pi_2: \text{если } x \text{ есть } A_2 \text{ и } y \text{ есть } B_2, \text{ то } z \text{ есть } C_2$$

где x и y – имена входных переменных, z – имя переменной вывода, $A_1, B_1, C_1, A_2, B_2, C_2$ – некоторые заданные функции принадлежности. При этом четкое значение z_0 необходимо определить на основе приведенной информации и четких значений x_0 и y_0.

Алгоритм Мамдани

В рассматриваемой ситуации он математически может быть описан следующим образом:

1)Введение нечеткости. Находятся степени истинности для предпосылок каждого правила:

$$A_1(x_0), A_2(x_0), B_1(x_0), B_2(x_0).$$

2) Логический вывод. Находятся уровни "отсечения" для предпосылок каждого из правил (с использованием операции МИНИМУМ):

$$\alpha_1 = A_1(x_0) \wedge B_1(y_0);$$

$$\alpha_2 = A_2(x_0) \wedge B_2(y_0);$$

где через " $\wedge$ " обозначена операция логического минимума (min). Затем находятся "усеченные" функции принадлежности:

$$C_1' = (\alpha_1 \wedge C_1(z));$$

$$C_2' = (\alpha_2 \wedge C_2(z)).$$

3) Композиция. Производится объединение найденных усеченных функций с использованием операции МАКСИМУМ (мах, обозначенные далее как "$\vee$"), что приводит к получению итогового нечеткого подмножества для переменной выхода с функцией принадлежности:

$$\mu_\Sigma(z) = C(z) = C_1'(z) \vee C_2'(z) = (\alpha_1 \wedge C_1(z)) \vee (\alpha_2 \wedge C_2(z)) \quad (1)$$

4) Приведение к четкости. Проводится для нахождения z_0, например, центроидным методом.

Алгоритм Цукамото

Исходные посылки – как у предыдущего алгоритма, но здесь предполагается, что функции $C_1(z), C_2(z)$ монотонными.

1) Введение нечеткости (как в алгоритме Мамдани).

2) Нечеткий вывод. Сначала находятся уровни "отсечения" α_1 и α_2 (как в алгоритме Мамдани), а затем решениями уравнений:

$$\alpha_1 = C_1(z_1) \quad \text{и} \quad \alpha_2 = C_2(z_2)$$

определяются четкие значения (z_1 и z_2) для каждого исходного правила.

3) Определяется четкое значение переменной вывода (как взвешенное среднее z_1 и z_2):

$$z_0 = \frac{\alpha_1 z_1 + \alpha_2 z_2}{\alpha_1 + \alpha_2} \quad (2)$$

Алгоритм Sugeno

Sugeno и Takagi использовали набор правил в следующей форме (как и ранее, приведем пример двух правил):

$$\Pi_1: \text{если } x \text{ есть } A_1 \text{ и } y \text{ есть } B_1 \text{ то } z_1 = a_1 x + b_1 y$$

$$\Pi_2: \text{если } x \text{ есть } A_2 \text{ и } y \text{ есть } B_2 \text{ то } z_2 = a_2 x + b_2 y$$

Описание алгоритма

1) Введение нечеткости (как в алгоритме Mamdani).

2) Нечеткий вывод. Находятся $\alpha_1 = A_1(x_0) \wedge B_1(y_0)$, $\alpha_2 = A_2(x_0) \wedge B_2(y_0)$ и индивидуальные выходы правил:

$$\dot{z}_1 = a_1 x_0 + b_1 y_0$$

$$\dot{z}_2 = a_2 x_0 + b_2 y_0$$

3) Определяется четкое значение переменной вывода

$$z_0 = \frac{\alpha_1 \dot{z}_1 + \alpha_2 \dot{z}_2}{\alpha_1 + \alpha_2} \qquad (3)$$

Градиентный алгоритм обучения ННС с гауссовскими функциями принадлежностями.

Используемый в работе [3] алгоритм обучения НК Мамдани носит эмпирический характер, формулы для настройки параметров функций принадлежности теоретически необоснованны. Это связано с тем, что в НК Мамдани и Цукамото традиционно используются треугольные ФП, а пересечение условий правил берется в форме min. В результате получаемые ФП оказываются недифференцируемыми.

В связи с этим авторами разработан аналитический алгоритм обучения, сходимость которого строго доказано. С этой целью необходимо перейти к гауссовским ФП, для условий и правил.

Итак, пусть ФП i-го μ - модуля связанного с правилом R_k описывается следующим выражением:

$$\mu_{ik}(x_i) = \exp\left\{-\frac{1}{2} * \frac{(x_i - a_{ik})^2}{\sigma_{ik}^2}\right\}, \qquad (4)$$

где a_{ik}, σ_{ik} - параметры подлежащие настройки в процессе обучения и ФП ν_k - модуля имеют следующий аналогичный вид:

$$\mu_k(y_i) = \exp\left\{-\frac{1}{2} * \frac{(y_i - a_k)^2}{\sigma_k^2}\right\},$$

при этом пересечение условий правил задается в виде произведения.

$$\alpha_k = \prod_{i=1}^{n} \mu_{ik}(x_i) = \exp\left\{-\frac{1}{2} * \frac{(x_i - a_{ik})^2}{\sigma_{ik}^2}\right\}. \qquad (5)$$

Допустим, что дефаззификация происходит по центроидному методу, тогда общий выход:

$$Z_0 = \frac{\sum_k z_k \alpha_k}{\sum_k \alpha_k}$$

Пусть для определения следствия правила используется монотонные ФП и z_kопределяется путём решения следующего уравнения (контроллер Цукамото).

$$C_к(z_k)= \alpha_k \tag{6}$$

Где $C_к(z_k)= \exp\left\{-\frac{1}{2} * \frac{(z_k - a_k)^2}{\sigma_k^2}\right\}$,

Тогда решая уравнение вида: $\exp\left\{-\frac{1}{2} * \frac{(z_k - a_k)^2}{\sigma_k^2}\right\} = \alpha_k$,

находим два корня: $Z_k= a_k \pm \sqrt{2 \ln \frac{1}{\alpha}} * \sigma_k$.

Первый корень $Z_{1k}= a_k - \sqrt{2 \ln \frac{1}{\alpha}} * \sigma_k$ – на монотонно возрастающем участке кривой $z_r(a_k)$, а второй z_{2k} – на участке монотонно убывающем участке.

Пусть критерий $E(z)=\frac{1}{2}\ (z_0-z^*)^2 \rightarrow \min$, где z^* - фактический выход; а z_0 – выход НК, тогда находим производные

$$\frac{\partial E}{\partial a_k} = \frac{\partial E}{\partial z_0}\frac{\partial z_0}{\partial z_k}\frac{\partial z_k}{\partial a_k} = +\left(z_0 - z^*\right)\frac{\alpha_k}{\sum_{k=1}^{K}\alpha_k}, \tag{7}$$

$$\frac{\partial E}{\partial \sigma_k} = \frac{\partial E_0}{\partial z_0}\frac{\partial z_0}{\partial z_k}\frac{\partial z_k}{\partial \sigma_k} = -\left(z_0 - z^*\right)\frac{\alpha_k}{\sum_{k=1}^{K} a_k}\sqrt{2 \ln \frac{1}{\alpha}} \tag{8}$$

на монотонно возрастающем участке кривой ФП $\mu_к$;

$$\frac{\partial E}{\partial a_k} = +\left(z_0 - z^*\right)\frac{\alpha_k}{\sum_{k=1}^{K} a_k}\sqrt{2 \ln \frac{1}{\alpha}} \tag{9}$$

на монотонно убывающем участке кривой.

Для входных μ-модулей

$$\frac{\partial E}{\partial \sigma_{ik}} = \frac{\partial E}{\partial z_0}\frac{\partial z_0}{\partial \alpha_k}\frac{\partial \alpha_k}{\partial a_{ik}} = (z_0 - z^*)\alpha_k \frac{z_k \sum_{k=1}^{K}\alpha_k - \sum_k z_k \alpha_k}{\left(\sum_k \alpha_k\right)^2}\frac{(x_i - a_{ik})}{\sigma_{ik}^2}$$

$$\frac{\partial E}{\partial \sigma_{ik}} = \frac{\partial E}{\partial z_0}\frac{\partial z_0}{\partial \alpha_k}\frac{\partial \alpha_k}{\partial \sigma_{ik}} = (z_0 - z^*)\alpha_k \frac{z_k \sum_{k=1}^{K}\alpha_k - \sum_k z_k \alpha_k}{\left(\sum_k \alpha_k\right)^2}\frac{(x_i - a_{ik})^2}{\sigma_{ik}^3} \tag{10}$$

и тогда градиентный алгоритм обучения ННС Мамдани выглядит следующим образом для выходных модулей:

$$a_k(n+1) = a_k(n) - \gamma_n \frac{\partial E}{\partial a_k} = a_k(n) - \gamma_n(z_0 - z^*)\frac{\alpha_k}{\sum \alpha_k} \tag{11}$$

$$\sigma_k(n+1)=\sigma_k(n)-\gamma_n^{'}\frac{\partial E}{\partial \sigma_k}=\sigma_k(n)-\gamma_n^{'}(z_0-z_\Sigma^{*})\frac{\alpha_k}{\sum\alpha_k}\sqrt{2\ln\frac{1}{\alpha_k}} \quad (12)$$

Экспериментальные исследования нечетких нейронных сетей в задачах прогнозирования в финасовой сфере

С целью оценки эффективности различных алгоритмов нечеткого вывода были проведены экспериментальные исследования различных классов нечетких нейросетей в задачах прогнозирования финансового рынка. Для прогнозирования был выбран рынок акций ОАО «Лукойл», допущенных к торгам на НП «фондовая биржа Российская торговая систем» (НПРТС). Были проведены эксперименты по прогнозированию курсов акций на РТС, используя разработанный программный продукт для трех алгоритмов. Для обучения использовалась выборка из 267 ежедневных значений пользователей курсов акций ОАО «Лукойл» за период с 1.04.2005 по 30.12.2005.

В ходе тестирования экспериментально было установлено, что наиболее оптимальным является использование трех термов и пяти правил обучения, так как при таких параметрах мы имеем самую минимальную СКО и наименьшее время обучение. Обучение параметров ФП производилось градиентным методом с шагом обучения 0,4.

1.Использование НК Мамдани при прогнозировании курсов акций.

Используя НК Мамдани с треугольными и гауссовскими ФП были получены следующие результаты прогнозирования курса акций ОАО «Лукойл». Они приведены в таблице 1.

Таблица 1. Результаты прогноза с использованием НК Мамдани для ФП Гаусса.

Дата	Реальное значение	Прогнозируемое значение	Отклонение	Квадрат отклонения
01.12.2005	58,1	58,23	0,13	0,0169
02.12.2005	58,7	58,54	0,16	0,0256
05.12.2005	59,4	59,14	0,26	0,0676
06.12.2005	59	59,11	0,11	0,0121
07.12.2005	59,85	59,97	0,12	0,0144
08.12.2005	59,6	59,416	0,184	0,033856
09.12.2005	59,9	60,12	0,22	0,0484
12.12.2005	60,65	60,5	0,15	0,0225
13.12.2005	60,65	60,54	0,11	0,0121
14.12.2005	61,15	61,32	0,17	0,0289
15.12.2005	60,25	60,1	0,15	0,0225
16.12.2005	61	61,2	0,2	0,04
19.12.2005	61,01	61,24	0,23	0,0529
20.12.2005	60,7	60,54	0,16	0,0256
			0,168142857	**СКО=0,030239714**

Далее было проведено прогнозирование при использовании НК Мамдани для треугольной ФП. Как показал первый эксперимент лучшим оказался контроллер Мамдани с гауссовскими ФП (СКО на проверочной выборке из 14 точек составляет всего 0,03024, относительная средняя ошибка 3,02%).

2. Далее проведены эксперименты по прогнозированию с использованием НК Цукамото с треугольными и гауссовскими ФП. Результаты прогноза НК Цукамото для ФП Гаусса приведены в табл.2, а для треугольной ФП в табл.3

Таблица 2. Результаты прогноза НК Цукамото для ФП Гаусса.

Дата	Реальное значение	Прогнозируемое значение	Отклонение	Квадрат отклонения
01.12.2005	58,1	58,37	0,27	0,0729
02.12.2005	58,7	58,47	0,23	0,0529
05.12.2005	59,4	59,1	0,3	0,09
06.12.2005	59	59,25	0,25	0,0625
07.12.2005	59,85	60,19	0,34	0,1156
08.12.2005	59,6	59,37	0,23	0,0529
09.12.2005	59,9	60,27	0,37	0,1369
12.12.2005	60,65	60,48	0,17	0,0289
13.12.2005	60,65	60,42	0,23	0,0529
14.12.2005	61,15	61,4	0,25	0,0625
15.12.2005	60,25	60,06	0,19	0,0361
16.12.2005	61	61,22	0,22	0,0484
19.12.2005	61,01	61,28	0,27	0,0729
20.12.2005	60,7	60,48	0,22	0,0484
			0,252857143	**СКО=0,0667**

Таблица 3. Результаты прогноза НК Цукамото для треугольной ФП.

Дата	Реальное значение	Прогнозируемое значение	Отклонение	Квадрат отклонения
01.12.2005	58,1	58,48	0,38	0,1444
02.12.2005	58,7	58,37	0,33	0,1089
05.12.2005	59,4	59,05	0,35	0,1225
06.12.2005	59	59,27	0,27	0,0729
07.12.2005	59,85	60,23	0,38	0,1444
08.12.2005	59,6	59,23	0,37	0,1369
09.12.2005	59,9	60,32	0,42	0,1764
12.12.2005	60,65	60,4	0,25	0,0625
13.12.2005	60,65	60,28	0,37	0,1369
14.12.2005	61,15	61,42	0,27	0,0729
15.12.2005	60,25	59,97	0,28	0,0784
16.12.2005	61	61,27	0,27	0,0729
19.12.2005	61,01	61,34	0,33	0,1089
20.12.2005	60,7	60,38	0,32	0,1024
			0,327857143	**СКО=0,110092857**

Как показал второй эксперимент, лучшим оказался контроллер Цукамото с гауссовской ФП (СКО на проверочную выборку из 14 точек составляет всего 0,0667, а средняя относительная ошибка прогноза 6,67%)

3.Далее были проведены сравнительные исследования эффективности прогнозирования с использованием следующих методов:

- контроллер Мамдани с гауссовскими ФП;
- контроллер Цукамото с гауссовскими ФП;
- контроллер Сугено с гауссовскими ФП;
- контроллер Мамдани с треугольными ФП;
- контроллер Цукамото с треугольными ФП;
- контроллер Сугено с треугольными ФП;
- нечеткая нейронная сеть ANFIS;

В табл.4 и табл.5 приведены сравнительные результаты прогнозирования курса акций ОАО «Лукойл», полученные разными методами нечеткого логического вывода (размер проверочной выборки 10 точек).

.

Таблица 4. Сравнение контроллеров с гауссовскими ФП и ANFIS

Реальное значение	Сеть ANFIS		ННК Мамдани с гауссовской ФП		ННК Цукамото с гаусовской ФП		ННК Сугено с гауссовской ФП	
	Прогноз	Ошибка	Прогноз	Ошибка	Прогноз	Ошибка	Прогноз	Ошибка
58,1	58,62	0,52	58,23	0,13	58,37	0,27	58,42	0,32
58,7	58,23	0,47	58,54	0,16	58,47	0,23	58,34	0,36
59,4	59	0,4	59,14	0,26	59,1	0,3	59,02	0,38
59	59,57	0,57	59,11	0,11	59,25	0,25	59,29	0,29
59,85	60,51	0,66	59,97	0,12	60,19	0,34	60,24	0,39
59,6	59,02	0,58	59,416	0,184	59,37	0,23	59,29	0,31
59,9	60,66	0,76	60,12	0,22	60,27	0,37	60,3	0,4
60,65	60,04	0,61	60,5	0,15	60,48	0,17	60,41	0,24
60,65	60,07	0,58	60,54	0,11	60,42	0,23	60,4	0,25
61,15	61,78	0,63	61,32	0,17	61,4	0,25	61,43	0,28
	СКО: 0,34312		**СКО: 0,028236**		**СКО: 0,0728**		**СКО: 0,10672**	

Таблица 5. Сравнение контроллеров с треугольными ФП и ANFIS.

Реальное значение	Сеть ANFIS		ННК Мамдани с треугольной ФП		ННК Цукамото с треугольной ФП		ННК Сугено треугольной с ФП	
	Прогноз	Ошибка	Прогноз	Ошибка	Прогноз	Ошибка	Прогноз	Ошибка
58,1	58,62	0,52	58,41	0,31	58,48	0,38	58,5	0,4
58,7	58,23	0,47	58,46	0,24	58,37	0,33	58,31	0,39
59,4	59	0,4	59,11	0,29	59,05	0,35	59,01	0,39
59	59,57	0,57	59,15	0,15	59,27	0,27	59,33	0,33
59,85	60,51	0,66	60,1	0,25	60,23	0,38	60,3	0,45
59,6	59,02	0,58	59,313	0,287	59,23	0,37	59,16	0,44
59,9	60,66	0,76	60,22	0,32	60,32	0,42	60,39	0,49
60,65	60,04	0,61	60,45	0,2	60,4	0,25	60,31	0,34
60,65	60,07	0,58	60,31	0,34	60,28	0,37	60,38	0,27
61,15	61,78	0,63	61,39	0,24	61,42	0,27	61,49	0,34
	СКО: 0,34312		**СКО: 0,072077**		**СКО: 0,081787**		**СКО: 0,15134**	

На рис.1 приведены результаты прогнозирования для НК Мамдани, Цукамото и Сугено, для гауссовских ФП и ННС ANFIS, а на рис.2 для треугольной ФП.

Как демонстрируют приведенные результаты в таблицах и на рис., наилучшим, хотя и с небольшим отрывом, оказался контроллер Мамдани с гауссовской ФП. Его СКО составляет всего 0,028236. Дальше по качеству прогноза идет контроллер Цукамото, причем гауссовские ФП дают немного лучший результат, чем треугольные. Но в целом их прогнозы очень близки (СКО=0,0728 и СКО=0,0817 соответственно).

Это дает основание допустить, что подбор еще более удачного кода функций принадлежности дает возможность еще больше улучшить результаты прогноза.

И наконец, уже на последнем месте (со сравнительно большим отрывом) находятся результаты, полученные с помощью ННС ANFIS (СКО=0,34312).

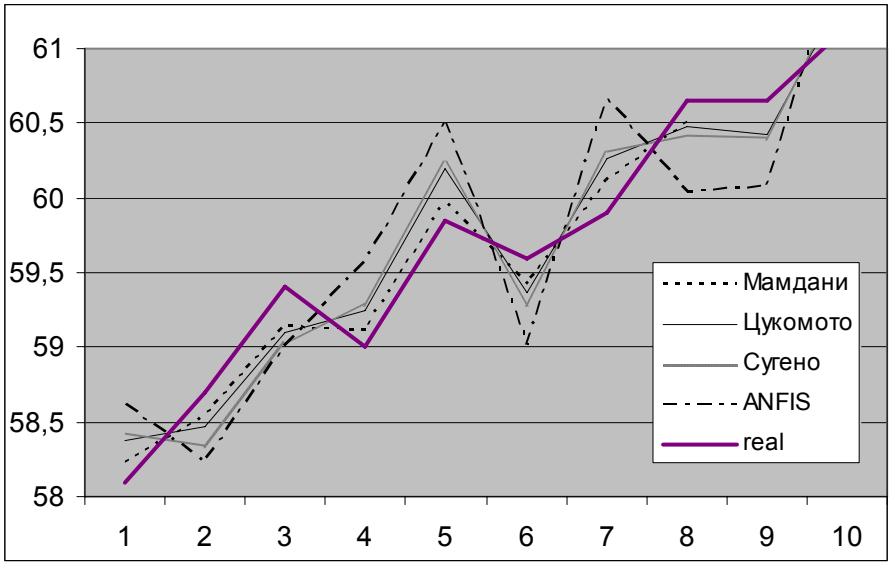


Рис.1

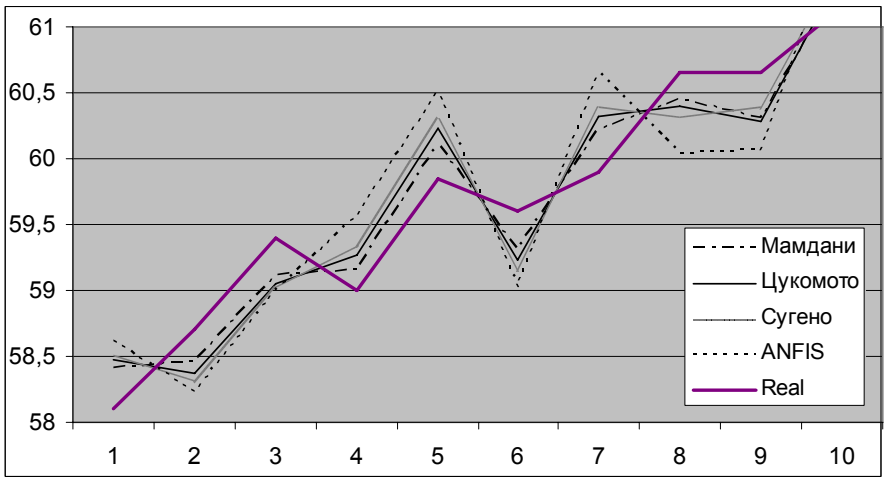


Рис.2

Выводы

1.В статье рассмотрены нечеткие нейронные сети с логическим выводом Мамдани, Цукамото и Сугено.

2.Описан алгоритм обучения гауссовского вида с выодом Мамдани и Цукамото.

3.Проведены экспериментальные исследования применения нечетких нейросетей в задачах прогнозирования финансовых показателей и выполнен анализ их эффективности.

4.Сравнительный анализ точности прогнозирования с использованием ННС Мамдани, Цукамото, Сугено и ANFIS показали, что наилучшей для прогнозирования экономических и финансовых показателей является НК Мамдани с гауссовской ФП, а наихудшей - ННС ANFIS, показатели которой существенно хуже в сравнении с НК Мамдани, Цукамото и Сугено. Этот результат можно объяснить тем, что в ННС ANFIS параметры выходных функций задаются априори и не настраиваются в процессе обучения, что является недостатком данной нейросети.

5.Проведенные экспериментальные исследования показали большие потенциальные возможности ННС и подтвердили их эффективность в задачах макроэкономического и финансового прогнозирования

Литература

1. Круглов В.В., Борисов В.В. Гибридные нейронные сети. Горячая линия – телеком, Москва, 2002.-382с.
2. Осовекий С. Нейронные сети для обработки информации. /Пер. с польского И.Д. Рудинского. – М. Финансы и статистика, 2002. – 344с.
3. Ярушкина Н.Г. «Нечеткие нейронные сети». /новости искусственного интеллекта, №3, 2001.-стр.47-51
4. Зайченко Ю.П., Севаее Фатма, Титаренко К.М., Титаренко Н.В. Исследования нечетких нейронных сетей в задачах макроэкономического прогнозирования. // Системні дослідження та інформаційні технології. -2004.-№2.-с.70-86.
5. Зайченко Ю.П., Севаее Фатма. Исследования нечеткой нейронной сети ANFIS в задачах макроэкономического прогнозирования. // В тр. 11-ой международной конференции “Knowledge-Dialogue-Solution ”. KDS.-2005. June 20-30, 2005. Varna, Bulgaria. pp.479-485.
6. Зайченко Ю.П., Севаее Фатма. Исследования нечеткой нейронной сети ANFIS в задачах макроэкономического прогнозирования. // Системні дослідження та інформаційні технології.-2005.-№1.-с.100-112
7. Зайченко Ю.П. Основи проектування інтелектуальних систем.- Вид.дім «Слово», Киев, 2004. – 352с.

Информация об авторах

Зайченко Юрий Петрович *– НТУУ Киевский политехнический институт, профессор, Киев, проспект Победы 37, тел. 38-044-2418693, e-mail: zaych@i.com.ua*

Фатма М. Севаее (Ливия) *– НТУУ Киевский политехнический институт, аспирант, Киев, проспект Победы 37, тел. 38-044-2876894*

Келестин Юрий Васильевич *– магистр, начальник объединения «Техноцентр», г. Рогатин, Ивано-Франковская обл., тел. 38-03435-21569*

RECURRENT LEARNING ALGORITHM FOR DOUBLE-WAVELET NEURON

Yevgeniy Bodyanskiy, Nataliya Lamonova, Olena Vynokurova

Abstract: *In this paper a new double wavelet neuron architecture obtained by modification of standard wavelet neuron, and its learning algorithm for its parameters are proposed. The offered architecture allows to improve the approximation properties of wavelet-neuron. Double wavelet neuron and its learning algorithm are examined for predicting non-stationary chaotic time series.*

Keywords*: wavelet, double-wavelet neuron, recurrent learning algorithm, forecasting, emulation.*

Introduction

Recently, in the analysis tasks and the non-stationary series processing under the uncertainty conditions computational intelligence techniques are widely used, particularly hybrid neural networks. The most important task related to signal processing is forecasting and emulation of dynamic non-stationary states of systems in the future.

For solving such kind of problems a variety of neural network architectures including hybrid architectures are used. However they are either bulky because of their architecture (for instance multilayer perceptron) or poorly adjusted for learning process in real time. In most cases the activation functions for these neural networks are sigmoidal functions, splines, polynomials and radial basis functions.

In addition wavelet theory is widespread [1-3] and allows with high accuracy to recognize the local characteristics of the non-stationary signals. At the confluency of the two approaches, hybrid neural networks and wavelet theory, has evolved the so-called wavelet neural networks [4-18] that has good approximating properties and sensitivity to the changes of characteristics of the analyzed processes.

Previous studies have proposed and described [19-21] attractive features of wavelet neuron such as technical realization, ensured accuracy and learning simplicity. At the same time the wavelet functions are incarnated either at the level of synaptic weights or the neuron output, and as a learning algorithm the gradient learning algorithm with constant step is used. For the improvement of approximation abilities and the acceleration of the learning process the present work introduces a new structure called double wavelet-neuron and learning algorithm that has smoothing and approximation properties.

Wavelet Activation Functions

As the activation functions of double-wavelet-neuron one can use various kind of analytical wavelets. Among them, two wavelet families, POLYWOG-wavelets and RASP-wavelets have of the most interesting properties.

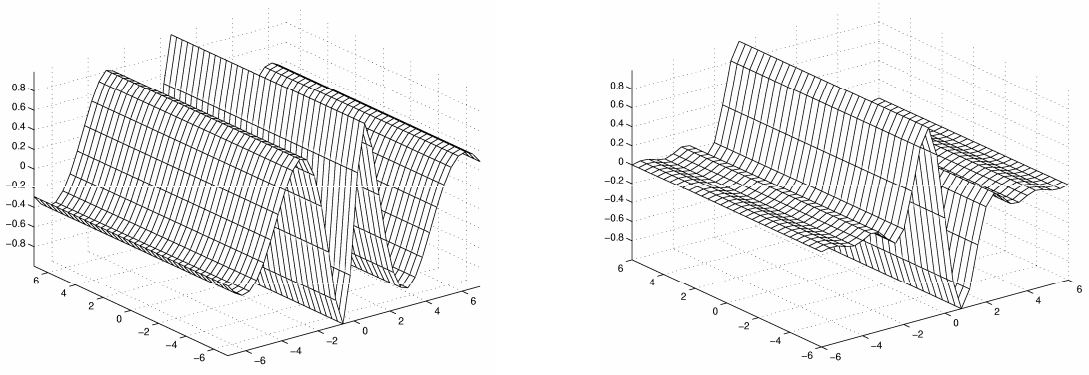
Wavelet RASP family consists of wavelets on the basis of rational functions (RAtional functions with Second-order Poles – RASP), concerned with the residue theorem of complex variables [4].

Fig. 1 shows two typical representatives of mother wavelet RASP, which can be described as following

$$\varphi_{ji}^{1}(x_i(k)) = \frac{\beta^1 \cos(x_i(k))}{x_i^2(k)+1}, \quad \beta^1 = 2.7435, \tag{1}$$

$$\varphi_{ji}^{2}(x_i(k)) = \frac{\beta^2 \sin(\pi x_i(k))}{x_i^2(k)-1}, \quad \beta^2 = 0.6111. \tag{2}$$

These wavelets are real odd functions with zero average.



a) b)

Fig. 1 – Representatives of wavelet RASP family

Another large wavelet family can be obtained from polynomials windowed with Gaussians type of functions (POLYnomials WindOwed with Gaussians type of function - POLYWOG) [4]. It is interesting to note that the derivatives from these functions are as well wavelet POLYWOG and can be used as mother wavelets.

Fig. 2 shows several typical wavelets from POLYWOG family which can be described as following

$$\varphi_{ji}^{1}(x_i(k)) = \mu^1 x_i(k) \exp\left(\frac{-x_i^2(k)}{2}\right), \quad \mu^1 = \exp\left(-1/2\right), \tag{3}$$

$$\varphi_{ji}^{2}(x_i(k)) = \mu^2 \left(x_i^3(k) - 3x_i(k)\right) \exp\left(\frac{-x_i^2(k)}{2}\right), \quad \mu^2 = 0.7246, \tag{4}$$

$$\varphi_{ji}^{3}(x_i(k)) = \mu^3 \left(x_i^4(k) - 6x_i^2(k) + 3\right) \exp\left(\frac{-x_i^2(k)}{2}\right), \quad \mu^3 = 1/3, \tag{5}$$

$$\varphi_{ji}^{4}(x_i(k)) = \left(1 - x_i^2(k)\right) \exp\left(\frac{-x_i^2(k)}{2}\right). \tag{6}$$

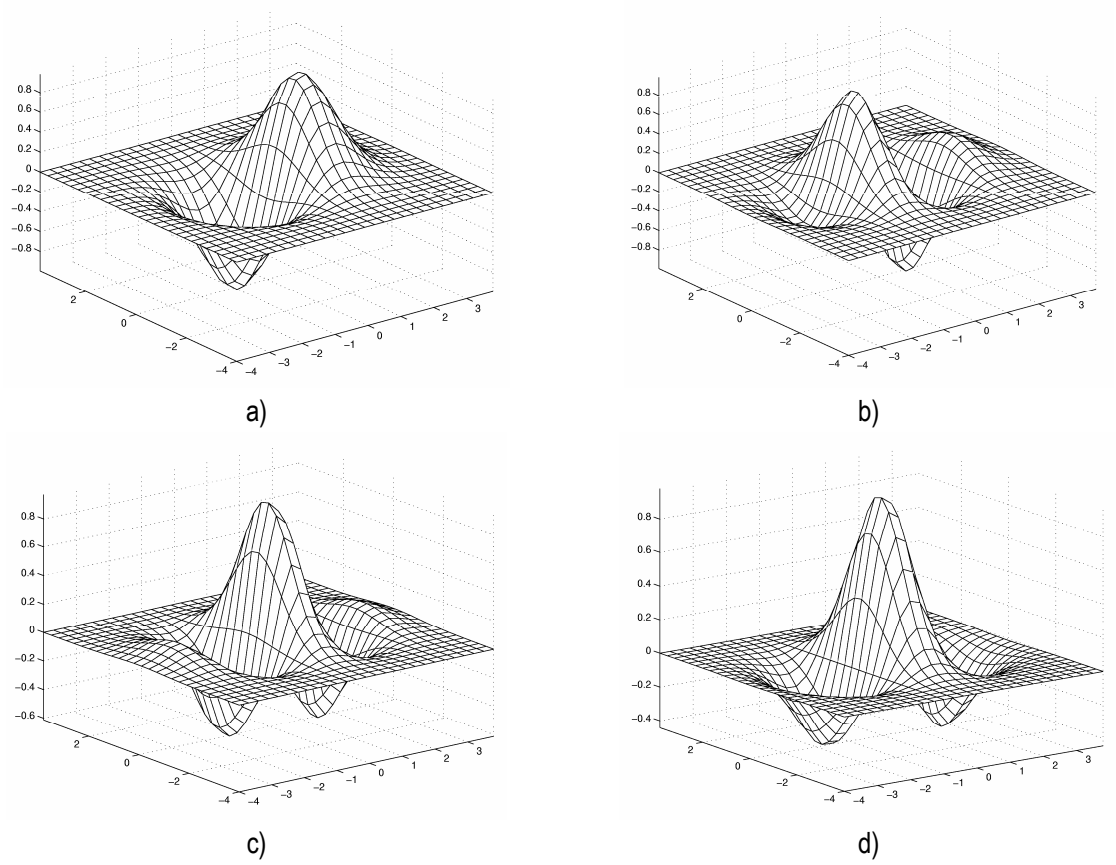


Fig. 2 – Representatives of POLYWOG family

Some wavelets of POLYWOG family can be obtained with the help of simple generators. In this way, wavelet of the family can be generated by taking into consideration hermicity properties of the polynomial derivative and Gaussian function.

Structure of Double-wavelet Neuron

Fig. 3 introduces the structure of double-wavelet neuron, and one can note that double wavelet-neuron is very close by its structure to *n*-input wavelet-neuron [19-21], however, it consists of nonlinear wavelet functions at the synaptic weights and output levels of the structure.

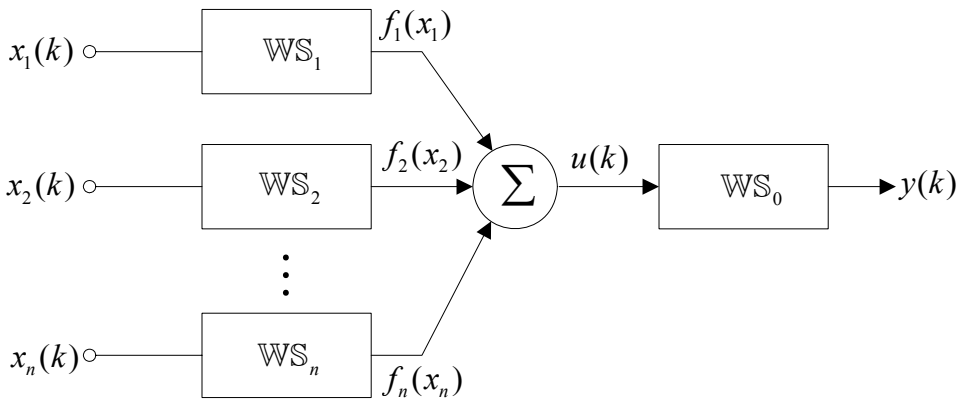


Fig. 3 – Generalized structure of double wavelet-neuron

If a vector signal $x(k) = (x_1(k),\ x_2(k),\ \ldots,\ x_n(k))^T$ (here $k = 0,\ 1,\ 2,\ \ldots$ number of sample in the training set or current discrete time) is fed to the input of he double wavelet-neuron shown in Fig. 4 then the output is determined as

$$y(k) = f_0\left(\sum_{i=1}^{n} f_i(x_i(k))\right) = f_0(u(k)) = \\ = \sum_{l=0}^{h_2} \varphi_{l0}\left(\sum_{i=1}^{n}\sum_{j=0}^{h_1} \varphi_{ji}(x_i(k)) w_{ji}(k)\right) w_{j0} = \sum_{l=0}^{h_2} \varphi_{l0}(u(k)) w_{l0}(k), \tag{7}$$

this is determined by synaptic weights $w_{ji}(k)$, w_{l0} as well as by the values of the used wavelet functions $\varphi_{ji}(x_i(k))$, $\varphi_{l0}(u(k))$, on assumption that $\varphi_{00}(\bullet) = \varphi_{0i}(\bullet) \equiv 1$.

The double wavelet-neuron is composed of two layers: hidden layer that contains *n*-wavelet synapses with h_1 wavelet-functions in each and output layer that contains one wavelet-synapse with h_2 wavelet-functions.

In each wavelet-synapse, the wavelets that differ between each other by dilation and translation factors and bias are realized.

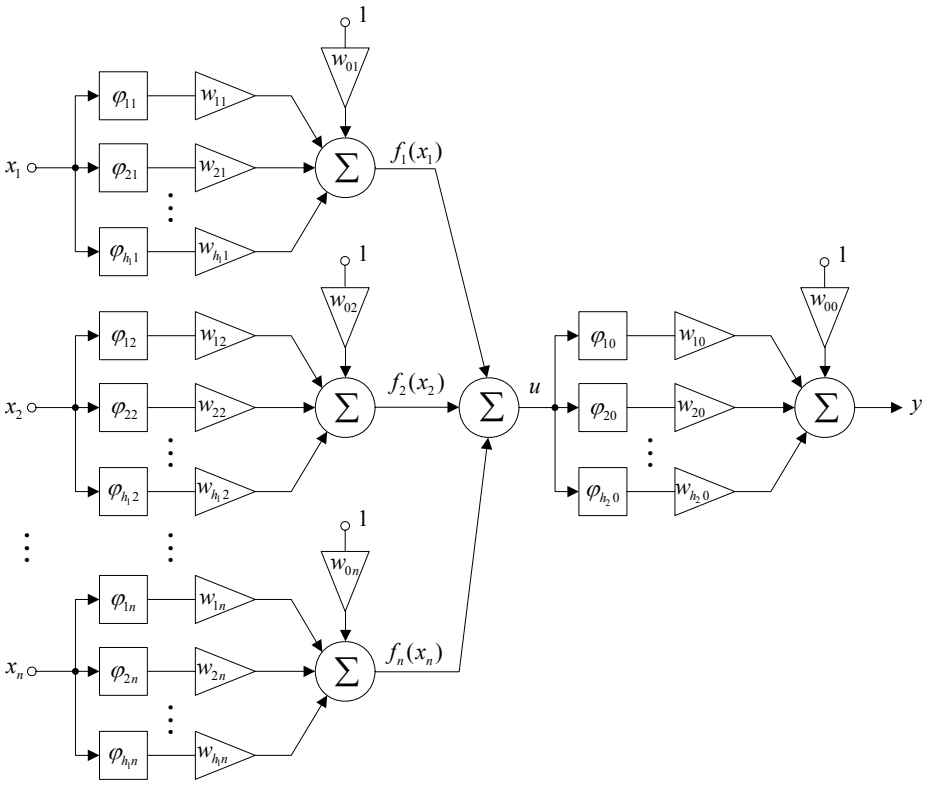


Fig. 4 - Architecture of double wavelet-neuron with nonlinear wavelet-synapses

Synthesis of Double Wavelet-neuron Learning Algorithm

In the tuning for the output layer of double wavelet-neuron as a criterion we use

$$E(k) = \frac{1}{2}(d(k) - y(k))^2 = \frac{1}{2}e^2(k), \tag{8}$$

where $d(k)$ - the external training signal.

The learning algorithm for the output layer of double wavelet-neuron on the basis of gradient approach is as

$$w_{j0}(k+1) = w_{j0}(k) + \eta_0(k) e(k) \varphi_{j0}(u(k)), \tag{9}$$

or in the vector form

$$w_0(k+1) = w_0(k) + \eta_0(k)e(k)\varphi_0(u(k)), \tag{10}$$

where $w_0(k) = (w_{10}(k), w_{20}(k), \ldots, w_{h_2 0}(k))^T$ - the vector of synaptic weights, $\varphi_0(k) = (\varphi_{10}(k), \varphi_{20}(k), \ldots, \varphi_{h_2 0}(k))^T$ - the vector of wavelet-activation functions, $e(k)$ - the learning error, $\eta_0(k)$ - learning rate parameter which is subject to determination.

To increase the rate of convergence of the training process it is necessary to turn from gradient procedures to the second-order procedures among which the Levenberg-Marquardt algorithm is widespread.

After simple transformations one can obtain the learning algorithm in the form

$$\begin{cases} w_0(k+1) = w_0(k) + \dfrac{e(k)\varphi_0(u(k))}{\gamma_i^{w_0}(k)}, \\ \gamma_i^{w_0}(k+1) = \alpha\gamma_i^{w_0}(k) + \left\|\varphi_0(u(k+1))\right\|^2, \end{cases} . \tag{11}$$

where α - the forgetting factor of out-dated information ($0 \le \alpha \le 1$).

The tuning of hidden layer is carried out in the same way on the basis of error backpropagation by using the same criterion written in the form

$$E(k) = \frac{1}{2}(d(k) - f_0(u(k)))^2 = \frac{1}{2}\left(d(k) - f_0\left(\sum_{i=1}^{n}\sum_{j=0}^{h_1} \varphi_{ji}(x_i(k))w_{ji}(k) \right) \right)^2 . \tag{12}$$

The learning algorithm for the hidden layer of double wavelet-neuron on the basis of gradient optimization has a form

$$w_{ji}(k+1) = w_{ji}(k) + \eta(k)e(k)f_0'(u(k))\varphi_{ji}(x_i(k)), \tag{13}$$

or in the vector form

$$w_i(k+1) = w_i(k) + \eta(k)e(k)f_0'(u(k))\varphi_i(x_i(k)), \tag{14}$$

where $w_i(k) = (w_{1i}(k), w_{2i}(k), \ldots, w_{h_1 i}(k))^T$ - the vector of synaptic weights, $\varphi_i(k) = (\varphi_{1i}(k), \varphi_{2i}(k), \ldots, \varphi_{h_1 i}(k))^T$ - the vector of wavelet-activation functions, $e(k)$ - the learning error, $\eta(k)$ - the learning rate.

By analogy with (11) one can introduce the procedure

$$\begin{cases} w_i(k+1) = w_i(k) + \dfrac{e(k)\, f_0'(u(k))\, \varphi_i(x_i(k))}{\gamma_i^{w_1}(k)}, \\ \gamma_i^{w_1}(k+1) = \alpha\gamma_i^{w_1}(k) + \left\|\varphi_i(x_i(k+1))\right\|^2, \end{cases} . \tag{15}$$

where $0 \le \alpha \le 1$.

Simulation Results

Effectiveness of performance for the proposed double wavelet-neuron and its learning algorithm (11), (15) were investigated in the process of solving forecasting problem and emulation of chaotic behavior of nonlinear dynamic system in the form

$$x_{n+1} = \frac{5x_n}{1+x_n^2} - 0.5x_n - 0.5x_{n-1} + 0.5x_{n-2} \tag{16}$$

with initial values $x_0 = 0.2,\ x_1 = 0.3,\ x_2 = 1.0$.

The training set contained 1000 samples, and checking set – 500 samples. Double wavelet-neuron had 5 synapses in the hidden layer corresponding to 5 inputs $x(k-4), x(k-3), x(k-2), x(k-1), x(k); (n=5)$

with 20 wavelets in each synapse ($h_i = 20, \quad i = 1 \ldots 5$). Output layer consisted of 5 wavelets in synapse WS_0 . Initial values of synaptic weights were generated in a random way from -0.1 to $+0.1$.

Several criterions were used for the quality rating of forecast:

- mean-square error (RMSE)

$$RMSE = \frac{1}{N}\sum_{k=1}^{N}(x(k) - \hat{x}(k))^2 ;$$

- Trefferquote [23, 23] represents percentage ratio of correctly predicted directions to actual direction of the signal

$$Trefferquote = \frac{N - \frac{1}{2}\sum_{k=1}^{N}\left|sign(\hat{x}(k) - x(k-1)) - sign(x(k) - x(k-1))\right|}{N} \cdot 100\% ;$$

- Wegstrecke [22, 23], represents quality rating of the predicted model (value +1 corresponds to the optimal predictive model, and -1 – to the incorrect forecast) and described by the equation

$$Wegstreke = \frac{\sum_{k=1}^{N} signal(k)(x(k) - x(k-1))}{\sum_{k=1}^{N}\left|x(k) - x(k-1)\right|},$$

where $signal(k)$ - sign-function in the form

$$signal(k) = \begin{cases} 1, & if \quad \hat{x}(k) - x(k) > 0, \\ -1, & if \quad \hat{x}(k) - x(k) < 0, \\ 0, & in \quad other \ cases, \end{cases}$$

$x(k)$ - the actual value of forecasting process, $\hat{x}(k)$ - the forecast, N - the length of training set.

Fig. 5 shows the results of forecast process on the basis of data from text set after 10 training epoch with the parameter $\alpha = 0.99$.

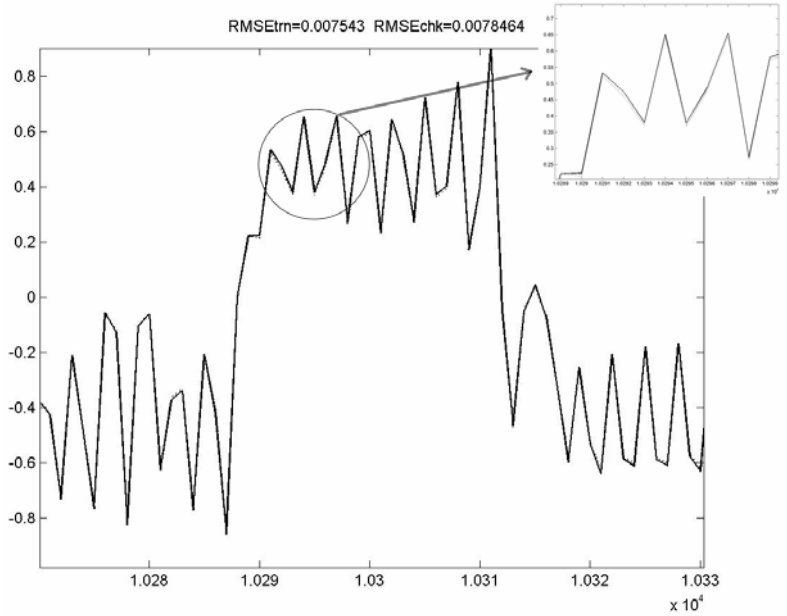


Fig. 5 – Forecasting of behavior of chaotic dynamic system with the help of the double wavelet-neuron

Table 1 shows results of forecasting process on the basis of the double wavelet-neuron compared the results of forecasting process on the basis of standard wavelet-neuron with the gradient learning algorithm, radial basis neural network and multilayer perceptron.

Table 1 - The results of time series forecasting

Neural network/ Learning algorithm	Number of adjustable parameters	Criteria		
		RMSE	Wegstrecke	Trefferquote
Double wavelet neuron / Proposed learning algorithm of parameters of wavelet-synapses (11) (15)	105	0.0078	1	99.8%
Wavelet-neuron / Gradient learning algorithm of parameters of wavelet-synapses with constant step	100	0.0101	0.98	98.8%
Radial basis neural network / RLSE	100	0.5774	0.4883	55,2%
Multilayer perceptron / Gradient learning algorithm	115	0.6132	0.5882	75,5 %

Thus as observed from experimental results the proposed double wavelet-neuron with the learning algorithm (11), (15) having the same number of adjustable parameters ensures the best quality of forecast and high learning speed in comparison with traditional architectures.

Conclusions

The double wavelet-neuron architecture and its learning algorithm which allows to adjust all parameters of network are proposed. The algorithm is very simple in the way of its numerical implementation, possesses high rate of convergence and additional smoothing and approximation properties.

Bibliography

[1]. Chui C. K. An Introduction to Wavelets. New York: Academic. 1992, 264 p.

[2]. Daubechies I. Ten Lectures on Wavelets. Philadelphia, PA: SIAM. 1992, 228 p.

[3]. Meyer Y. Wavelets: Algorithms and Applications. Philadelphia, PA: SIAM. 1993, 133 p.

[4]. Lekutai G., van Landingham H.F. Self-tuning control of nonlinear systems using neural network adaptive frame wavelets. Proc. IEEE Int. Conf. on Systems, Man and Cybernetics. Piscataway, N.J. 2, 1997, P. 1017-1022.

[5]. Bodyanskiy Ye., Lamonova N., Pliss I., Vynokurova O. An adaptive learning algorithm for a wavelet neural network. Blackwell Synergy: Expert Systems. 22 (5), 2005, P. 235-240.

[6]. Bodyanskiy Ye., Kolodyazhniy V. Pliss I., Vynokurova O. Learning wavelet neuron based on the RASP-function. Radio Electronics. Computer Science. Control. 1., 2004, P. 118-122.

[7]. Бодянский Е.В., Винокурова Е.А., Ламонова Н.С. Адаптивная гибридная вэйвлет-нейронная сеть для решения задачи прогнозирования и эмуляции. Сб. науч. трудов 12-й международной конференции по автоматическому управлению «Автоматика 2005», Т.3. – Харьков: Изд-во НТУ «ХПИ»., 2005, С. 40-41

[8]. Бодянский Е.В., Винокурова Е.А. Треугольный вэйвлет и формальный нейрон на его основе. Сб. науч. трудов 3-й Междунар. научно-практической конференции «Математическое и программное обеспечение интеллектуальных систем» (MPZIS-2005). Днепропетровск: ДНУ, 2005, С. 14-15.

[9]. Billings S. A., Wei H.-L. A new class of wavelet networks for nonlinear system identification. IEEE Trans. on Neural Networks. 16 (4), 2005, P. 862-874.

[10]. Szu H. H., Telfer B., Kadambe S. Neural network adaptive wavelets for signal representation and classification. Opt. Eng. 31, 1992, P. 1907–1916.

[11]. Zhang Q. H., Benveniste A. Wavelet networks. IEEE Trans. on Neural Networks. 3 (6), 1992, P. 889–898.

[12]. Dickhaus H., Heinrich H. Classifying biosignals with wavelet networks. IEEE Eng. Med. Biol. Mag. 15(5), 1996, P. 103–111.

[13]. Cao L. Y., Hong Y. G., Fang H. P., He G. W. Predicting chaotic time series with wavelet networks. Phys. D. 85, 1995, P. 225–238.

[14]. Oussar Y., Dreyfus G. Initialization by selection for wavelet network training. Neurocomputing. 34, 2000, P. 131–143.

[15]. Zhang J., Walter G. G., Miao Y., Lee W. N. W. Wavelet neural networks for function learning. IEEE Trans. on Signal Process. 43(6), 1995, P. 1485–1497.

[16]. Zhang Q. H. Using wavelet network in nonparametric estimation. IEEE Trans. on Neural Networks. 8(2), 1997, P. 227–236.
[17]. Casdagli M. Nonlinear prediction of chaotic time series. Phys. D. 35, 1989, P. 335–356.
[18]. Soltani S. On the use of wavelet decomposition for time series prediction. Neurocomputing. 48, 2002, P. 267–277.
[19]. Yamakawa T., Uchino E., Samatu T. Wavelet neural networks employing over-complete number of compactly supported non-orthogonal wavelets and their applications. IEEE Int. Conf. on Neural Networks, Orlando, USA., 1994, P. 1391-1396.
[20]. Yamakawa T., Uchino E., Samatu T. The wavelet network using convex wavelets and its application to modeling dynamical systems. The Trans. on the IEICE. J79-A. 12, 1996, P. 2046-2053.
[21]. Yamakawa T. A novel nonlinear synapse neuron model guaranteeing a global minimum – Wavelet neuron. Proc. 28 th IEEE Int. Symp. On Multiple-Valued Logic. Fukuoka, Japan: IEEE Comp. Soc., 1998, P.335-336
[22]. Baumann M. Nutzung neuronaler Netze zur Prognose von Aktienkursen. Report Nr. 2/96, TU Ilmenau., 1996, 113 p.

Authors' Information

Bodyanskiy Yevgeniy – *Doctor of Technical Sciences, Professor of Artificial Intelligence Department and Scientific Head of the Control Systems Research Laboratory, Kharkiv National University of Radio Electronic, Lenina av. 14, Kharkiv, Ukraine 61166, Tel +380577021890, e-mail: bodya@kture.kharkov.ua*

Lamonova Nataliya – *Candidate of Technical Sciences (equivalent Ph.D.), Senior Research Assistant of Control Systems Research Laboratory, Kharkiv National University of Radio Electronic, Lenina av. 14, Kharkiv, Ukraine 61166, Tel +380577021890, e-mail: webmaster@natashka.de*

Vynokurova Olena – *Candidate of Technical Sciences (equivalent Ph.D.), Senior Research Assistant of Control Systems Research Laboratory Kharkiv National University of Radio Electronic, Lenina av. 14, Kharkiv, Ukraine, 61166, Tel +380577021890, e-mail: vinokurova@kture.kharkov.ua*

GROWING NEURAL NETWORKS BASED ON ORTHOGONAL ACTIVATION FUNCTIONS

Yevgeniy Bodyanskiy, Irina Pliss, Oleksandr Slipchenko

Abstract*: In the paper, an ontogenic artificial neural network (ANNs) is proposed. The network uses orthogonal activation functions that allow significant reducing of computational complexity. Another advantage is numerical stability, because the system of activation functions is linearly independent by definition. A learning procedure for proposed ANN with guaranteed convergence to the global minimum of error function in the parameter space is developed. An algorithm for structure network structure adaptation is proposed. The algorithm allows adding or deleting a node in real-time without retraining of the network. Simulation results confirm the efficiency of the proposed approach.*

Keywords*: ontogenic artificial neural network, orthogonal activation functions, time-series foreccasting.*

Introduction

Artificial neural networks (ANNs) are widely applied to solving a variety of problems such as information processing, data analysis, system identification, control etc. under structural and parametric uncertainty [1, 2].

One of the most attractive properties of ANNs is the possibility to adapt their behavior to the changing characteristics of the modeled system. By adaptivity we understand not only the adjustment of parameters (synaptic weights), but also the possibility to adjust the architecture (the number of nodes). The goal of the present paper is the development of an algorithm for structural and synaptic adaptation of ANNs for nonlinear system modeling, capable of online operation, i.e. sequential information processing without re-training after structure modification.

The problem of optimization of neural network architecture has been studied for quite a long time. The algorithms that start their operation with simple architecture and gradually add new nodes during learning, are called 'constructive algorithms'. In contrast, destructive algorithms start their operation with an initially redundant network, and simplify it as learning proceeds. This process is called 'pruning'.

Radial basis function network (RBFN) is one of the most popular neural network architectures [3]. One of the first constructive algorithms for such networks was proposed by Platt and named 'resource allocation' [4]. By present time, a number of modifications of this procedure is known [5, 6]. One of the most known is the cascade-correlation architecture developed by Fahlman and Lebiere [7].

Among the destructive algorithms, the most popular are the 'optimal brain damage' [8] and 'optimal brain surgeon' [9]. In these methods, the significance of a node or a connection between nodes is determined by the change in error function that its deletion incurs. For this purpose, the matrix of second derivatives of the optimized function with respect to the tunable parameters is analyzed. Both procedures are quite complex computationally. Besides that, an essential disadvantage is the need for re-training after the deletion of non-significant nodes. This, in turn, makes the real-time operation of these algorithms impossible. Other algorithms such as [10] are heuristic and lack universality.

It should be noted that there is no universal and convenient algorithm, which could be used for the manipulation of the number of nodes and suitable for most problems and architectures. Many of the algorithms proposed so far lack theoretical justification as well as the predictability of the results of their application and the ability to operate in real time.

Network Architecture

Let's consider the network architecture, that implements the following nonlinear mapping

$$\hat{y}(k) = \hat{f}(x(k)) = \sum_{i=1}^{n} \sum_{j=1}^{h_i} w_{ji} \phi_{ji}(x_i(k)) \tag{1}$$

where $k = 1, 2, \ldots$ – discrete time or ordinal number of sample in training set, w_{ji} – tunable synaptic weights, $\phi_{ji}(\bullet)$ – j-th activation function for i-th input variable, h_i – number of activation functions for appropriate input variable, $x_i(k)$ – value of i-th input signal at time moment k (or for k-th training sample).

Proposed architecture contains $h = \sum_{i=1}^{n} h_i$ tunable parameters and it can be readily seen that the this number is between the scatter-partitioned and grid-partitioned systems.

We propose the use of orthogonal polynomials of one variable for the basis functions. Particular system of functions can be chosen according to the specificity of the solved problem. If the input data are normalized on the hypercube $[-1, 1]^n$, the system of Legendre polynomials orthogonal on the interval $[-1, 1]$ with weight $\gamma(x) \equiv 1$ [17] can be used:

$$P_n(x) = 2^{-n} \sum_{m=0}^{[n/2]} (-1)^m \frac{(2n-2m)!}{m!(n-m)!(n-2m)!} x^{n-2m}, \tag{2}$$

where $[\bullet]$ is the integer part of a number.

Among other possible choices for activation functions we should mention Chebyshev [15, 16] and Hermite [18] polynomials as well as non-sinusoidal orthogonal systems proposed by Haar and Walsh.

Synaptic Adaptation

The sum of squared errors will be used as the learning criterion:

$$E(k) = \sum_{p=1}^{p} e^2(k) = \sum_{p=1}^{p} (y(p)) - \sum_{i=1}^{n} \sum_{j=1}^{h_i} w_{ji} \phi_{ji}(x_i(p)))^2 \tag{3}$$

For the convenience of further notation, let us re-write the expression for the output of the neural network (1) in the form

$$\hat{y}(k+1) = \phi^T(k+1)W(k), \tag{4}$$

where $\phi(k) = (\phi_{11}(x(k)), \phi_{21}(x(k)), ..., \phi_{h_n n}(x(k)))^T$ is a $(h \times 1)$ vector of the values of the basis functions for the *k*-th element of the training set (or at the instant *k* for sequential processing), $W(k) = (w_1(k), ..., w_h(k))^T$ is a $(h \times 1)$ vector of estimates of synaptic weights at the iteration k.

Since the output of the proposed neural network depends on the tuned parameters linearly, we can use the least squares procedure to estimate them. For sequential processing, e.g. in the case of online identification, we can use the recursive least squares method:

$$\begin{cases} W(k+1) = W(k) + \dfrac{P(k)(y(k+1) - W^T(k)\phi(k+1))\phi(k+1)}{1 + \phi^T(k+1)P(k)\phi(k+1)}, \\ P(k+1) = P(k) - \dfrac{P(k)\phi(k+1)\phi^T(k+1)P(k)}{1 + \phi^T(k+1)P(k)\phi(k+1)}. \end{cases} \tag{5}$$

Because of the orthogonality of the basis functions, the matrix $P(k)$ will tend to diagonal form as $k \to \infty$. If the activation functions are orthonormal, $P(k)$ will tend to the unity matrix. Due to this property, the learning procedure will retain numerical stability with the increase of the number of samples in the training sequence.

Structure Adaptation

We consider sequential learning that minimizes (3). This leads to the estimate

$$W_h(k) = R_h^{-1}(k)F_h(k), \tag{6}$$

$$R_h^{-1}(k) = R_h^{-1}(k-1) - \frac{R_h^{-1}(k-1)\phi(k)\phi(k)^T R_h^{-1}(k-1)}{1 + \phi(k)^T R_h^{-1}(k-1)\phi(k)}, \tag{7}$$

$$F_h(k) = F_h(k-1) + \phi(k)y(k). \tag{8}$$

The use of the recursive least squares (RLS) method and its modifications allows to obtain an accurate and well-interpretable measure of significance of each function in the mapping (1). This mapping can be considered as an expansion of an unknown reconstructed function in the basis $\{\phi_{ji}(.)\}$. Obviously, if the absolute value of any of the coefficients in this expansion is small, then the corresponding function can be excluded from the basis without significant loss of accuracy. The remaining synaptic weights does not need to be retrained if the weight of the excluded node is close to zero. Otherwise, the network should be retrained.

Assume that a vector of synaptic weights $W_h(k)$ of a network comprising h nodes was obtained at the instant k using the formula (6), where the index h determines the number of basis functions (the dimension of $\varphi(k)$). Also assume that the absolute value of the considered parameter $w_h(k)$ is small, and we want to exclude corresponding unit function from the expansion (1). The assumption about the insignificance of the activation h is not restrictive, because we always can re-number the basis functions. This will result only in the rearrangement of the rows and columns in the matrix $R_h(k)$ and in the change of ordering of the elements of the vector $F_h(k)$. However, the rearrangement of columns and/or rows of a matrix does not influence the subsequent matrix operations.

Taking into account the fact that the matrix $R_h(k)$ is symmetric, we obtain:

$$W_h(k) = R_h^{-1}(k)F_h(k) = \begin{pmatrix} R_{h-1}(k) & \beta_{h-1}(k) \\ \beta_{h-1}^T(k) & r_{hh}(k) \end{pmatrix}^{-1} \begin{pmatrix} F_{h-1}(k) \\ f_h(k) \end{pmatrix}, \tag{9}$$

where $r_{ij}(k)$ is the element of the i-th row and j-th column of the matrix $R_h(k)$, $\beta_{h-1}(k) = (r_{1h}(k),...,r_{h-1h}(k))^T = (r_{h1}(k),...,r_{hh-1}(k))^T$, $f_i(k)$ is the i-th element of vector $F_h(k)$.

After simple transformations of (9) we obtain the expression

$$W_h(k) = \begin{pmatrix} W_{h-1}(k) - R_{h-1}^{-1}(k)\beta_{h-1}(k)w_h(k) \\ w_h(k) \end{pmatrix} \tag{10}$$

that enables us to exclude the function from (1) and obtain the corrected estimates of the remaining parameters of the ANN. For this operation, we use only the information accumulated in the matrix $R_h(k)$ and vector $F_h(k)$.

Using the same technique as above, we can obtain a procedure that can be used to add a new function to the existing basis. Direct application of the Frobenius formula [12] leads to the algorithm

$$W_{h+1}(k) = R_{h+1}^{-1}(k)F_{h+1}(k) = \begin{pmatrix} R_h(k) & \beta_h(k) \\ \beta_h^T(k) & r_{h+1,h+1}(k) \end{pmatrix}^{-1} \begin{pmatrix} F_h(k) \\ f_{h+1}(k) \end{pmatrix} =$$

$$\begin{pmatrix} W_h(k) + R_h^{-1}(k)\beta_h(k)\dfrac{\beta_h^T(k)W_h(k) - f_{h+1}(k)}{r_{h+1h+1}(k) - \beta_h^T(k)R_h^{-1}(k)\beta_h(k)} \\ \dfrac{-\beta^T(k)W_h(k) + f_{h+1}(k)}{r_{h+1,h+1}(k) - \beta_h^T(k)R_h^{-1}(k)\beta_h(k)} \end{pmatrix} \tag{11}$$

where $\beta_h(k) = (r_{1h+1}(k),...,r_{hh+1}(k))^T = (r_{h+11}(k),...,r_{h+1h}(k))^T$.

Thus, with the help of equation (11) we can add a new function (neuron) to the model (1), and exclude an existing function using the formula (10) without retraining remaining weights. In order to perform these operations in real time, it is necessary to accumulate the information about a larger number of basis functions than currently being used. E.g., we can initially introduce a redundant number of basis functions H and accumulate information in the matrix $R_H(k)$ and vector $F_H(k)$ as new data arrive, with only $h < H$ basis functions being used for the description of the unknown mapping. The complexity of the model can be either reduced or increased as required.

Analysis of equations (6), (10), and (11) shows that the efficiency of the proposed learning algorithm is directly related to the condition number of the matrix $R_h(k)$. This matrix will be non-singular if the functions $\{\varphi_i(.)\}_{i=1}^h$ used in the expansion (1) are linear-independent. The best situation is when the function system $\{\varphi_i(.)\}_{i=1}^h$ is orthogonal. In this case, the matrix $R_h(k)$ becomes diagonal, the formulas (6), (10), and (11) being greatly simplified because

$$diag(a_1,..,a_n)^{-1} = diag\left(\frac{1}{a_1},...,\frac{1}{a_n}\right), \tag{12}$$

where $diag(a_1,..,a_n)$ is an $(n \times n)$ matrix with non-zero elements $a_1,..,a_n$ only on the main diagonal.

Simulation Results

We have applied the proposed ontogenic network with orthogonal activation functions to online identification of a rat's (*Ratus Norvegius Vistar*) brain activity during sleeping phase.

The signal was measured with frequency of 64 Hz. We took a fragment of signal containing 3200 points (50 second of measuring), that was typical for sleeping phase of rat's life activity. Two neural networks of type (1) were trained in real-time. Each network had 10 inputs – delayed signal values ($y(k)$, $y(k-1)$,..., $y(k-9)$) and was trained to output one-step ahead value of the process – $y(k+1)$. First network utilized synaptic adaptation algorithm (6) while second one also involved the structure adaptation technique (10), (11). Initially both

ANNs had 5 activation functions per input, the one with synaptic adaptation only retained all 50 tunable parameters during it's work while ANN with structure adaptation mechanism had only 25 fired functions (the most significant ones chosen in real-time). For the results comparing purpose we also trained multilayer perceptron (further referred as MLP) with the same structure of inputs and training signal, having 5 units in the 1st and 4 in the 2nd hidden layers (that totals to 74 tunable parameters). As MLP is not capable of real-time data processing, all samples are used as training set and test criteria are calculated on the same data points. MLP was trained during 250 epochs with Levenberg-Marquardt algorithm. Our research showed that this is enough to achieve precision comparable to proposed ontogenic neural network with orthogonal activation functions.

Results of identification can be found in table 1. Fig. 1 shows the results of identification using proposed neural network. We used some different measures of identification quality. First, we analyse normalized root mean squared error, that is closely related to the learning criterion. Two other criteria used: "Wegstrecke" [19] characterizes the quality of the model for prediction/identification (+1 means perfect one), "Trefferquote" [20] is percent value of correctly predicted direction changes.

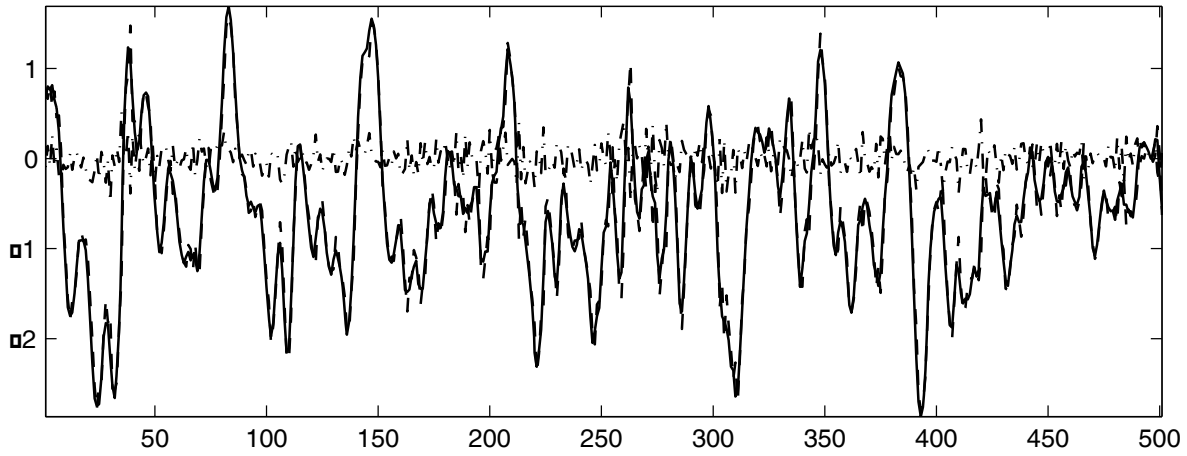


Figure 1. Identification of a rat's brain activity during sleeping phase using proposed neural network with orthogonal activation functions – brain activity signal (*solid line*), network output (*dashed line*), and identification error (*dash-dot line*)

We can see that utilizing structure adaptation technique leads to somewhat worth results. This is the tradeoff for having less tunable parameters and possibility to process non-stationary signals.

Table 1 – Identification results for different architectures

Decription	NRMSE	Trefferquote	Wegstrecke
OrthoNN, real-time processing	0.1834	82.3851	0.85221
OrthoNN, real-time processing, variable number of nodes	0.2187	77.6553	0.74872
MLP, offline learning (250 epochs), error on the training set	0.1685	83.9533	0.87192

Conclusion

A new computationally efficient neural network with orthogonal activation functions was proposed. It has a simple and compact architecture not affected by the curse of dimensionality, and provides high precision of nonlinear dynamic system identification. An apparent advantage is much easier implementation and lower computational load as compared to the conventional neural network architectures.

The approach presented in the paper can be used for nonlinear system modeling, control, and time series prediction. An interesting direction of further work is the use of the network with orthogonal activation functions as a part of hybrid multilayer architecture. Another possible application of proposed ontogenic neural network is its use as a basis for diagnostic systems.

References

1. Handbook of Neural Computation. IOP Publishing and Oxford University Press, 1997.
2. Nelles O. Nonlinear System Identification. Berlin, Springer, 2001.
3. Poggio T. and Girosi F. A Theory of Networks for Approximation and Learning. A.I. Memo No. 1140, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 1989.
4. Platt J. A resource allocating network for function interpolation. Neural Computation, 3, 1991, p. 213-225.
5. Nag A. and Ghosh J. Flexible resource allocating network for noisy data. In: Proc. SPIE Conf. on Applications and Science of Computational Intelligence, SPIE Proc. Vol. 3390, Orlando, Fl., April 1998, p. 551-559.
6. Yingwei L., Sundararajan N. and Saratchandran P. Performance evaluation of a sequential minimal radial basis function (RBF) neural network learning algorithm. IEEE Trans. on Neural Networks, 9, 1998, p. 308-318.
7. Fahlman S. E. and Lebiere C. The cascade-correlation learning architecture. Technical Report CMU-CS-90-100, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, 1990.
8. Cun Y. L., Denker J. S., Solla S. A. Optimal Brain Damage. Advances in Neural Information Processing Systems, 2, 1990, p. 598-605.
9. Hassibi B. and Stork D. G. Second-order derivatives for network pruning: Optimal brain surgeon. In: Advances in Neural Information Processing Systems, Hanson et al. (Eds), 1993, p. 164-171.
10. Prechelt L. Connection pruning with static and adaptive pruning schedules. Neurocomputing, 16, 1997, p. 49-61.
11. Takagi T. and Sugeno M. Fuzzy identification of systems and its application to modeling and control. IEEE Trans. on System, Man and Cybernetics. 15, 1985, p. 116-132.
12. Gantmacher F. R. The Theory of Matrices. Chelsea Publ. Comp., New York, 1977
13. Narendra K. S. and Parthasarathy K. Identification and control of dynamic systems using neural networks. IEEE Trans. on Neural Networks, 1, 1990, p. 4-26.
14. Scott I. and Mulgrew B. “Orthonormal function neural network for nonlinear system modeling”. In: Proceedings of the International Conference on Neural Networks (ICNN-96), June, 1996.
15. Patra J.C. and Kot A.C. Nonlinear dynamic system identification using Chebyshev functional link artificial neural network. IEEE Trans. on System, Man and Cybernetics – Part B, 32, 2002, p. 505-511.
16. Bodyanskiy Ye.V., Kolodyazhniy V.V., and Slipchenko O.M. “Forecasting neural network with orthogonal activation functions” In: Proc. of 1st Int. conf. “Intelligent decision-making systems and information technologies”, Chernivtsi, Ukraine, 2004, p. 57. (in Russian)
17. Bateman, H., Erdelyi, A.: Higher Transcendental Functions. Vol.2. McGraw-Hill (1953)
18. Liying M., Khorasani K. Constructive Feedforward Neural Network Using Hermite Polinomial Activation Functions. IEEE Trans. On Neural Networks, 16, No. 4, 2005, p.821–833.
19. Baumann M. Nutzung neuronale Netze zur Prognose von Aktionkursen. – Report Nr. 2/96, TU Ilmenau, 1996. – 113 S.
20. Fueser K. Neuronale Neteze in der Finanzwirtshaft. – Wiesbanden: Gabler, 1995. – 437 S.

Authors' Information

Yevgeniy Bodyanskiy - *Dr. Sc., Prof., Head of Control Systems Research Laboratory, Kharkiv National University of Radio Electronics, Lenin Av., 14, Kharkiv, 61166, Ukraine, e-mail: bodya@kture.kharkov.ua*

Irina Pliss - *Ph.D., Senior research scientists, Control Systems Research Laboratory, Kharkiv National University of Radio Electronics, Lenin Av., 14, Kharkiv, 61166, Ukraine, e-mail: pliss@kture.kharkov.ua*

Oleksandr Slipchenko - *Ph.D., Senior research scientists, Control Systems Research Laboratory, Kharkiv National University of Radio Electronics, Lenin Av., 14, Kharkiv, 61166, Ukraine, e-mail: slipchenko@kture.kharkov.ua*

DISTRIBUTED REPRESENTATIONS IN CLASSIFICATION TASKS

Ivan S. Misuno, Dmitri A. Rachkovskij, Sergey V. Slipchenko

***Abstract**: Binary distributed representations of vector data (numerical, textual, visual) are investigated in classification tasks. A comparative analysis of results for various methods and tasks using artificial and real-world data is given.*

***Keywords**: Distributed representations, binary representations, coarse coding, classifiers, perceptron, SVM, RSC*

***ACM Classification Keywords**: C.1.3 Other Architecture Styles - Neural nets, I.2.6 Learning - Connectionism and neural nets, Induction, Parameter learning*

Introduction

Classification tasks consist in assigning input data samples to one or more classes from a predefined set [1]. Classification in the inductive approach is realized on basis of training set that contains data samples with predefined class labels. Usually, input data samples are represented as numeric vectors. Vector elements are real numbers (e.g., some measurements of object characteristics or their function) or binary values (indicators of some features in the input data).

This vector information often doesn't contain information relevant to the classification explicitly, therefore some kind of transformation is necessary. We created methods for transformation of input information of various kinds (such as numerical [2], textual [3], visual [4]) to binary distributed representations. Those representations can then be classified by linear classifiers – such as *SVM* [5] or more computationally effective and naturally handling multiple classes perceptron-like ones [4, 6]. An objective of this research is the investigation of efficiency of the proposed methods for distributed information representation and classification using real and artificial data of different modalities.

Numeric Vector Data Classification

For an experimental research of abovementioned methods on numeric data the following well-known test problems have been selected: Leonard-Kramer *LK*, *XOR*, *Double Spiral* [6]; datasets generated by *DataGen* [6]; and sample data from the *Elena* database [7]. The dimensionality A of data vectors varied from 2 to 36, number of classes C varied from 2 to 11, and the number of samples in the training and test sets varied from 75 to 3218.

All selected problems have essentially non-linear class boundaries. Therefore, non-linear transformation of input numeric vectors has been used - i.e., *RSC* and *Prager* [2] methods of encoding. Those methods extract binary features – indicators of input A-dimensional vector presence in s-dimensional ($s<A$) hyperrectangle receptive fields with random position and size.

To investigate the impact of code parameters on the classification quality, the following experimental scheme was selected. Input vectors were converted to *RSC* and *Prager* codes. Those codes were used as input data for training and testing linear classifiers. The number (or percent) of test errors was chosen as a classification quality criterion. We used *SVM* [5] and modifications of perceptron-like classifiers [4] as linear classifiers for the obtained distributed representations. Besides, classification experiments with (non-linear) kernel *SVM* using *Prager*, *RSC* [2] and standard (Gaussian and polynomial) kernels were conducted.

It is well known [5] that *SVM* doesn't support online learning, requires solving computationally expensive non-linear programming problems, and constructs optimal separating hyperplane for two-class problems only. In this work we have also used a perceptron with an enlarged margin and multi-class learning rule which has been developed by us in order to overcome *SVM* drawbacks. In the developed perceptron outputs of neurons that correspond to classes are determined as $y_c = \Sigma_i x_i w_{ic}$, where w_{ic} are weights of modifiable connections, x_i is the i-th element value of the vector that is input to those connections (in the present context it was the binary vector of

distributed representations, but the original data vector could be used for linear tasks as well). For the “true” class neuron $y_{c\text{-}true} = y_{c\text{-}true}(1–T)$, where $0<T<1$ is the “defence margin parameter". The classification result is the index c^* of neuron with the maximum activation: $c^* = \text{argmax}_c\, y_c$. In case of error ($c^* \neq c_{true}$) connections are modified in the following way: $w_{ic} = w_{ic} + \Delta w$ for $c=c_{true}$ and $w_{ic} = w_{ic} – f(\Delta w)$ for c: $y_c > y_{c\text{-}true}$, where c_{true} is the index of the correct class. E.g., $f(\Delta w) = \Delta w/|c|$. Our previous version of the enlarged margin perceptron had single-class (not multi-class) learning rule: unlearning with single class $c^* = \text{argmax}_c\, y_c$ was performed in case of error, and $f(\Delta w) = \Delta w$. For T=0 and single-class learning rule one obtains usual percepton, while for T=0 and multi-class learning rule one obtains usual percepton with multi-class learning. Multi-class learning extracts and uses more information from a single error and so provides a potential for faster learning and better generalization for essentially multi-class tasks, especially at early learning iterations of the training set. This could be critical for on-line learning tasks.

Experimental Results for Numerical Data

Figure 1 demonstrates Leonard-Kramer problem results: dependencies of classification errors percent %*err*, elementary cell size *cell* (the smaller is the *cell,* the larger is resolution), and average fields dimensionality *E*{*s*} vs the code density *p* (the fraction of 1s in the code). For *Prager* and *RSC* coding, the results of *SVM* and of the enlarged margin perceptron (*T*=0.75) with multi-class learning were averaged by 10 realizations of codes at *N*=100. Results for *SVM* with kernels (*Kernel*) are also shown. For all cases (as well as for large *N*s Figure 2) classification error reaches its minimum near *p*=0.25, which corresponds to the minimum *cell* and *E*{*s*} = 2.

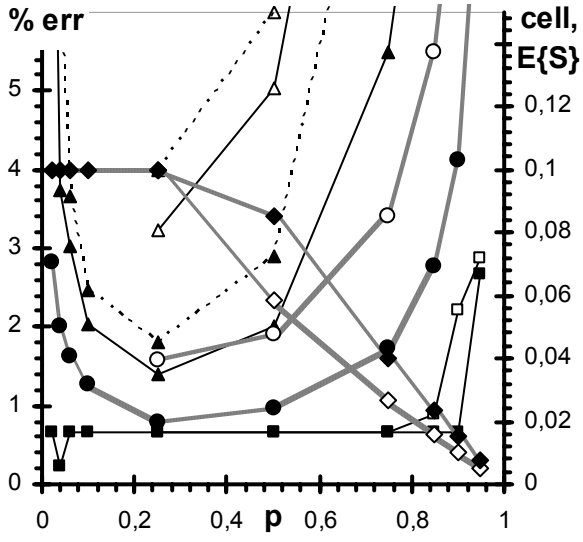


Figure 1

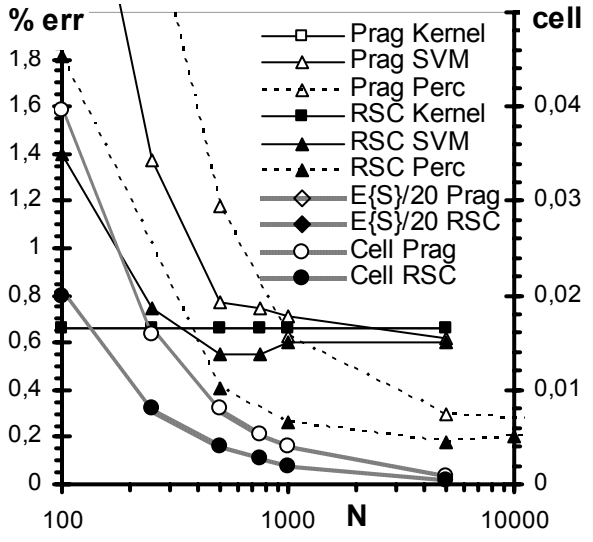


Figure 2

Figure 2 demonstrates %*error* and *cell* vs *N* at *p*=0.25. The results were averaged by 10 realizations of codes. At *N*=500 the *SVM* result has already been close to the kernel result. For the enlarged margin perceptron (*T*=0.75) with multi-class learning the error for *N*>(300–1000) was lower than the *SVM* one. Training for perceptron was faster than that for *SVM* by 20 times, while testing was >100 times faster.

The experimental results for the *DataGen* data are given in Figure 3 (*A*=4, *S*=3, *C*=4, *R*=4, where *R* determines the complexity of the class regions [6]) and the number of samples per class is equal to 100. Averaging was conducted through 5 realizations of the *DataGen* samples and 5 realizations of codes. For those parameters the minimum cell value corresponds to *p*~0.3 (and close to it for *p*=0.125...0.5) and the error minimum for both *SVM* and the enlarged margin perceptron is also reached in this interval. For *N*=100 it is biased to the larger *p* values (which ensures a more stable number of 1s). For *N*=1000 the minimum is biased to the smaller *p* which corresponds to a larger mean dimensionality of receptive fields, while the number of 1s remains large enough and the *cell* is small enough. The training time for the perceptron is ~20 times less than for *SVM*, and the testing time is ~500 times less.

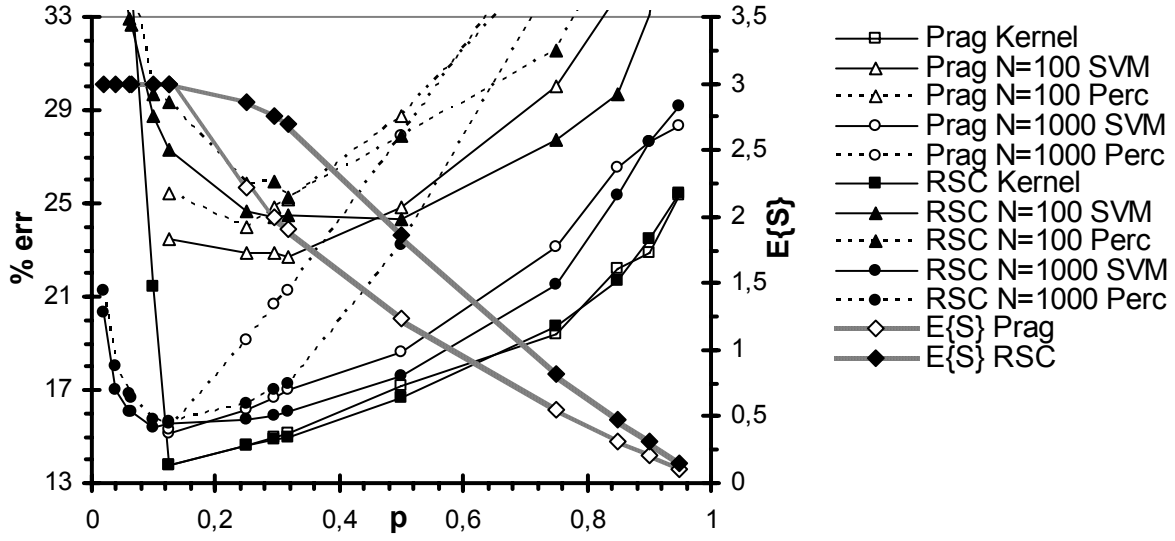


Figure 3

We have also obtained and compared experimental results for multi-class and single class learning perceptrons. The error for multi-class learning perceptron was up to 1.5 times lower than for the usual one, whereas the error for multi-class perceptron with the enlarged margin was still lower and comparable with the error for single-class perceptron with the enlarged margin. The learning curves (test error vs training iteration number) were typically lower for multi-class learning than for single-class learning, and best results for multi-class perceptrons were typically higher than those for multi-class perceptrons.

For the artificial data of the *Elena* database the code parameters were N=1000, A=S=2, p=0.25; for the real data (*Iris*, *Phoneme*, *Satimage*, *Texture*) N=10000, S=2,5(4), p=0.1 and 0.25. Table 1 demonstrates percent of classification errors. For *SVM* and perceptron the results were obtained by averaging over 10 realizations of *RSC* and *Prager* codes. The best results of the known methods *kNN*, *MLP*, *IRVQ* are also given [7]. The comparison of results shows that *RSC* and *Prager* coding provided the best result to *Concentric*, *Phoneme*, *Texture* and the second best result for *Satimage* and *Gaussian 7D*. Perceptron training time is (on the average) several times less, and test time is dozens of times less than that for SVM.

Classification of Texts and Images

Traditional approaches to text classification use functions of word occurrence frequencies as elements of their vector representations. To reduce vectors' dimensionality, methods for informative feature selection can be used [4]; however, even simplified methods that consider features as independent have quadratic computational complexity. We propose and investigate usage of distributed representations for dimensionality reduction of vector text representation. N-dimensional binary code with m 1s in random positions is used to represent each word. N-dimensional text representation is formed by summation of its word vectors, and transformation back to binary space is performed by a threshold operation, or by context-dependent thinning *CDT* (see [3]).

Table 1

Database	RSC SVM	RSC kern.	RSC perc.	Prager SVM	Prager kern.	kNN	MLP	IRVQ
Clouds	12.68	14.84	–	12.4	14.8	11.8	12.2	11.7
Concentric	1.36	1.2	–	1.17	1.04	1.7	2.8	1.5
Gaussian2 S=2	28.12	35.12	–	27.83	35.64	27.4	26.8	27.2
Gaussian7 S=2	14.35	15.68	–	14.36	15.76	15.9	15.3	11.5
Gaussian7 S=5	14.69	14.64	–	13.36	15.12	–	–	–
Iris S=2	6.53	6.67	5.33	5.59	6.67	4	4.3	6.7
Iris S=4	4.27	6.67	5.73	6.13	6.67	–	–	–
Phoneme S=2	14.12	11.51	13.7	15.79	14.47	12.3	16.3	16.4
Phoneme S=5	13.61	11.62	13.19	14.82	12.62	–	–	–
Satimage S=2	10.06	10.13	9.15	10.82	10.79	9.9	12.3	11.4
Satimage S=5	10.11	–	9.1	10.64	–	–	–	–
Texture S=2	0.82	0.76	1.13	0.82	0.80	1.9	2.0	3.1
Texture S=5	0.73	–	1.07	0.74	–	–	–	–

Testing in the classification task has been conducted using *Reuters*-21578 text collection [3] by means of *SVM*. For the *TOP*-10 categories *BEP* (break even point of recall/precision characteristic) for the initial vector representation of N^*=20000 was 0.920/0.863 (micro/macro averaging). Using of the distributed representations with N=1000, m=2 made it possible to obtain 0.861/0.775 (micro/macro averaging), and usage of *CDT* in some experiments increased it by several percents.

The analogously formed distributed representations have been studied for classification of handwritten digit images of the *MNIST* database [4]. Images have been coded by the extracting binary features. The presence of each feature corresponded to the combination of white and black points in some positions of retina (*LIRA* features [4]). As a result, a "primary" binary code was obtained. Then it was transformed to the "secondary" representation using the same procedures as for text information.

Classification results with dimensionality reduction from N^* to N are shown in Table 2. Line “sel” contains the error percent obtained using selection of informative features [4]. Line “distr” contains classification results for the "secondary" binary distributed representations. Results for the distributed representations considerably exceed the results of initial representations for the same N and are similar to the results of feature selection methods [4].

We have also obtained and compared *MNIST* experimental results for multi-class and single class learning perceptrons. Here we used original *LIRA* features without transformation to secondary distributed representations, for N={1000, 10000, 50000}, and both with and without feature selection. We observed the same tendencies as for numerical data, however the advantage of multi-class learning was more pronounced for weaker classifiers (at N=1000) than for better ones (at N=50000).

Table 2

N (err)	5000(667)	10000 (407)		50000 (195)			128000 (160)		
*N**	1000	1000	5000	1000	5000	10000	1000	5000	10000
sel	820	578	420	492	264	242	474	261	218
distr	904	727	**415**	632	274	**213**	826	264	**204**

Conclusions

The developed binary distributed representations of vector data (numeric, text, images) were investigated in the classification tasks. A comparative analysis of various method results for the tasks with artificial and real data was carried out. The study showed that analytical expressions for the characteristics of the *RSC-Prager* codes of the numerical vectors obtained in [2] make it possible to select code parameters that provide high results in the non-linear classification tasks using linear classifiers. Results obtained with the proposed perceptron with an enlarged margin are comparable with the results of the state-of-art *SVM* classifiers, however a significant decrease in training and recognition time has been observed. The results obtained with the *RSC-Prager* kernels also make it possible to reduce training and test time for small S.

Application of distributed encoding for representation of binary features in texts and images also made it possible to obtain computationally effective solutions to classification tasks preserving classification quality. A promising direction of further studies could consist in developing computationally efficient *RSC* and *Prager* kernels, as well as developing distributed representations and kernels that provide a more adequate account for structural information in the input data.

Bibliography

[1] R.. Duda, P. Hart, D. Stork. Pattern Classification, 2nd ed. – New York: John Wiley & Sons, 2000.

[2] S.V. Slipchenko, I.S. Misuno, D.A. Rachkovskij. Properties of coarse coding with random hyperrectangle receptive fields. Mathematical machines and systems, N 4, pp. 15-29, 2005 (in Russian).

[3] I.S. Misuno. Distributed vector representation and classification of texts. USIM, N 1, pp. 85-91, 2006 (in Russian).

[4] S.V. Slipchenko, D.A. Rachkovskij, I.S. Misuno. The experimental research of handwritten digits classification. System technologies, N 4 (39), pp. 110–133, 2005 (in Russian).

[5] V.N. Vapnik. Statistical Learning Theory. – New York: John Wiley & Sons, 1998.

[6] I.S. Misuno, D.A. Rachkovskij, E.G. Revunova, S.V. Slipchenko, A.M. Sokolov, A.E. Teteryuk. Modular software neurocomputer SNC - implementation and applications. USiM, N 2, pp. 74–85, 2005 (in Russian).

[7] D. Zhora. Evaluating Performance of Random Subspace Classifier on ELENA Classification Database. Artificial Neural Networks: Biological Inspirations – ICANN 2005 – Springer–Verlag Berlin Heidelberg, pp. 343–349, 2005.

Authors' Information

Ivan S. Misuno – *e-mail: i.misuno@longbow.kiev.ua*
Dmitri A. Rachkovskij – *e-mail: dar@infrm.kiev.ua*
Sergey V. Slipchenko – *e-mail: slipchenko_serg@ukr.net*
International Research and Training Center of Information Technologies and Systems; Pr. Acad. Glushkova, 40, Kiev, 03680, Ukraine

SELECTING CLASSIFIERS TECHNIQUES FOR OUTCOME PREDICTION USING NEURAL NETWORKS APPROACH

Tatiana Shatovskaya

Abstract*: This paper presents an analysis of different techniques that is designed to aid a researcher in determining which of the classification techniques would be most appropriate to choose the ridge, robust and linear regression methods for predicting outcomes for specific quazistationarity process.*

Keywords*: classification techniques, neural network, composite classifier*

ACM Classification Keywords*: F.2.1 Numerical Algorithms and Problems*

1. Introduction

There are a lot of approaches to building mathematical models for kvazistationarity process with multicollinearity and noisiness. For example, ridge regression is a linear-regression variant that is used for highly correlated independent variables, as is often the case for a set of predictors that are designed to approximate the same function [1]. Ridge regression adds a constraint that the sum of the squares of the regression coefficients be equal to a constant λ. Varying this parameter produces a set of predictors. Robust methods estimation parameters of mathematical model have stability in relation to infringement of requests normality the rests of model. They are insensitive not only to mistakes in a dependent variable, but also take into account a degree of influence of points of factorial space, that is reveal emissions in independent variables that allows to receive effective estimations of the coefficients regression models. For all methods a necessary condition of a solvency of their estimations is symmetry of allocating of mistakes of regression model.

But the main problem for the researcher is how to select an appropriate method for given task. In some cases using only one classification method for choosing the estimation method could not the solve problem. A multitude of techniques exists for modeling process outcomes. But the selection of modeling techniques to use for a given class of process is a nontrivial problem as there are many techniques from which to choose. It could be that the modeling technique used is not the most appropriate for the task and that accuracy can be increased through the use of a more appropriate model. There are many reasons why a model may have low predictive value.

This paper presents an analysis of different techniques that is designed to aid a researcher in determining which of the classification techniques would be most appropriate to choose the ridge, robust and linear regression methods for predicting outcomes for specific kvazistationarity process. We shall try to see that success can be attained with particular architectures on commonly used data for such process.

According to goal of our researching it is suggesting to create two-layer architecture in which the classifiers to be combined are called level-0 classifiers, and the combining classifier is the level-1 classifier. The layering may be iterated to create level-2 classifiers, and so on. Such architecture is a framework for classifier combination in which each layer of classifiers is used to combine the predictions of the classifiers at the immediately preceding layer. A single classifier at the top-most level outputs the ultimate prediction. The classifier at each layer receives as input a vector of predictions of the classifiers in the layer immediately below. While the information passed

from layer to layer may take the form of vectors of predictions, confidence values, or other data, we will limit our attention to systems in which only predictions of estimation methods class are passed from layer to layer. We will also limit ourselves to two-layer generalizes, consisting of a set of component classifiers and a single combining classifier that combines the predictions of the component classifiers.

In effect, such combining classifiers are an attempt to minimize generalization error by using the classifiers in higher numbered layers to learn the types of errors made by the classifiers immediately below. The task of the level-1 (and higher) classifiers is to learn to use the contestant predictions to predict more accurately.

Such combining classifiers framework diagram looks like a multilayer neural network diagram (Fig. 1).

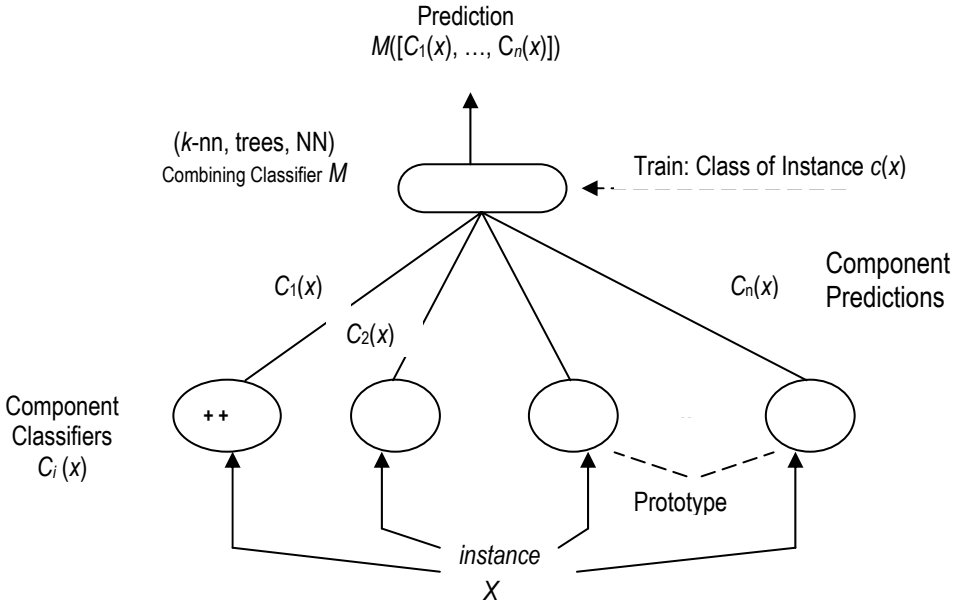


Fig. 3. Classifier architecture

There are certainly analogous aspects to the two frameworks. The distinction between them appears to lie partially in the type of information that is passed from the input layer to the succeeding layer and in the granularity of the classifier nodes themselves. In a neural network, an activation value is passed to forward layers, which may or may not be an ultimate prediction or even have some recognizable interpretation. Generally, in the stacked generalization framework, a “full-fledged” class prediction is passed to the combining classifier, and not just a scalar that somehow contributes to a prediction. Also, in other implementations of such classifiers, the classifiers to be stacked are complex, and may be neural networks themselves.

2. Architecture and Algorithm

We have been given a set of n level-0 (component) learning algorithms, a level-1 learning (combining) algorithm, and a training set of classified instances, T_0. The *n* level-0 learning algorithms should be distinct, so that diverse level-0 classifiers are obtained. Otherwise, no synergy will result from their combination. How to create diverse component classifiers is a fundamental problem for composite classifier construction. Our algorithm has the two phases, training and application.

Training Phase:

1. Train the component classifiers as follows. For each instance in the data set, train each of the *n* level-0 classifiers using the remaining instances. After training, classify the held-out instance using each of the trained level-0 classifiers. Form a vector from the predictions of each of the level-0 classifiers and the actual class of that instance. These vectors have length *n* + 1, since they have as components the predictions of each of the n level-0 component classifiers and a class label.

2. Train the level-1 classifier, using as the level-1 training set the collection of vectors of the level-0 classifier predictions and the actual classes. This collection has cardinality $|T_0|$, since there is one level-1 training instance corresponding to each level-0 training instance.

3. Since the level-0 classifiers have not been trained on the entire training set, re-train the level-0 classifiers on the entire training set.

Application Phase:

When presented with a new instance whose class is unknown, classify the instance using each of the level-0 classifiers, deriving an input vector for the level-1 classifier. The derived vector is then classified by the level-1 classifier, which outputs a prediction for the new instance. Leave-one-out cross validation is applied in the training phase to ensure that the level-1 algorithm is trained on the generalizations made for unseen data by the level-0 classifiers. Since "generalization" refers to data outside the training set, this observation is memorialized in the name "composite generalization", as opposed to "stacked classification".

In an experiment with combining linear, ridge, robust regression function showed that using 10-fold cross validation to create the level-1 training data yielded slightly more accurate stacked generalizes than when we applied only leave-one-out cross validation. Also in our experiment has been used decision-tree to generate classifiers that make diverse prediction. We combines a set of trees that have been pruned to the *k*-node trees that displayed the smallest training set error, for various choices of *k*. Investigation of the effect of the combination of neural networks with different numbers of units have been performed too. The accuracies of a given model will vary for the different prediction, so have opportunity to compare it on commonly used data.

In our study we used a commonly used data and compare prediction as follow:

Maximal accuracy prediction: predicted value must lie within a narrow range of actual value.

Minimal level prediction: actual value is no less than 5 point below predicted value.

Significant assistance prediction.

Table 1. Accuracy prediction

Model	Accuracy
Combination of Decision trees	55.7%
Combination of Linear discriminant function	68.9%
Combination of Neural network	76.5%
Linear regression	45.8%

The accuracy for each model for the minimal level prediction is higher than those for the same model for the maximal accuracy prediction. Obtained results shows that combined classifier of neural network have the best accuracy prediction. Does this suggest that artificial neural network models should be used for all outcome predictions in class of kvazistationarity process?

For check-up such situation the experiment was designed to test "whether such composite classifier of combination of neural network can be used to separate ridge and robust estimation methods for incomplete input information" using a set of neural network.

As income information from quasistationarity process with multicolinearity and noisiness for level-0 classifiers used: volume of sample, number of independent variables, degree of multicollinearity, dispersion of a mistake in a dependent variable, ratio of scales of "littering" and basic distributions of the "polluted" distribution of mistakes of model, degree of pollution of independent variables, the form of emissions in independent variables, length of a tail of the "polluted" distribution of independent variables. As a level-0 classifier we used a Probabilistic neural network, Multiple Perceptron Layers, Radial Basis Function for prediction a class or subclass of methods. When an input task is given, the allocator determines which module (neural network) should be used to fulfill this task. Generally, many modules might be selected to fulfill the task together. Each of these selected modules outputs a result based on local computation. The coordinator then gives the final result based on outputs of the modules. If the allocator is so strong that a single module can always be correctly selected to perform a given task, the coordinator can be removed. If, on the other hand, the allocator is so weak that all modules must be used to fulfill a task, a strong coordinator would be useful to make the final judgment. Interesting enough, most existing nets are different from each other simply because their allocators or coordinators are stronger or weaker.

Bibliography

1. Лесная Н.С., Репка В.Б., Шатовская Т.Б. Метод выбора эффективных процедур оценивания параметров моделей квазистационарных процессов в нейросетевой экспертной системе // Радиотехника. Всеукраинский межведомственный научно-технический сборник. - Харьков. 2001. - № 119. - С. 195-198.
2. Skalak D.B. Prototype selection for composite nearest neighbour classifiers. Neurological Research 2001; 20. Pp. 116-328.
3. Breiman L., Friedman J.H., Olshen R.A., Stone C.J. Classification and regression tress. Belmont, CA: Wadsworth. 1984.
4. Lang E.W., Pitts L.H., Damron S.L., Rutledge R. Outcome after severe head injury: Analysis of prediction based upon comparison of neural network versus logistic regression analysis. Neurological Research 1997; 19. P. 274-280.
5. Grisby J., Kooken R., Hershberger J. Simulated neural network to predict outcomes, cost and length of stay among orthopaedic rehabilitation patients. Arch. Phys. Med. Rehabil. 1994. Vol. 75. P. 1077-1082.
6. Manchester Metropolitan University. Department of Computing. Report, September 1997. A Modular Neural Network Architecture with Additional Generalization Abilities for High Dimensional Input Vectors.
7. Happle H. and Murre. Design and Evolution of Modular Neural Network Architectures. 2000. Vol. 75. P. 256-277

Authors' Information

Tatiana Shatovskaya *- Department of Software Engineering, Kharkiv National University of Radioelectronics, Computer Science Faculty, 61166, Kharkiv, Lenin avenue 14, e-mail: mywork@kture.kharkov.ua*

ИНТЕЛЛЕКТУЛАЬНАЯ ОПТИМИЗАЦИЯ ИСКУССТВЕННЫХ НЕЙРОННЫХ СЕТЕЙ

Кирилл Юрков

Аннотация*: Рассматриваются возможности применения достижений в области искусственного интеллекта (ИИ) для оптимизации искусственных нейронных сетей (ИНС) под конкретную задачу. Особое внимание уделено мультиагентным системам (МАС). Предложен ряд метафор миров агентов для применения при оптимизации.*

Ключевые слова*: Мультиагентные системы, искусственные нейронные сети.*

ACM Classification Keywords*: I.2: Artificial Intelligence: I.2.11 Distributed Artificial Intelligence - Multiagent systems, I.2.6 Learning – Connectionism and neural nets*

Введение

Во многих книгах по искусственным нейронным сетям (ИНС) написано, что нейронные сети не могут считаться панацеей [Уоссермен, 1992], [Gonzalez, 2000]– в теории они могут почти все, а вот на практике – только как получится. Тем не менее, нередко от ИНС ждут, что поставленная задача будет решена как по мановению волшебной палочки, на необработанных данных первая попавшаяся сеть уловит все внутренние закономерности данных и будет выдавать только правильные результаты. После первых опытов с ИНС вполне может наступить горькое разочарования, способное заставить отказаться от дальнейшего применения ИНС. С другой стороны, после тщательного штудирования литературы по ИНС, можно прийти в ужас от количества исследовательской работы, которую необходимо проделать, для того чтобы выбрать правильную, в смысле почти оптимальную и наиболее адекватную задаче, сеть. Исключения могут составлять те узкие области, где ИНС давно и с успехом применяются, например, распознавание. Но распознавание распознаванию рознь, и если сеть таки показывает неудовлетворительные результаты, разработчик опять сталкивается с необходимостью проведения исследовательской работы.

Поэтому вполне естественным является желание облегчить разработчику работу, избавив его от проведения большого количества рутинных экспериментов, и возложить обязанности по нахождению оптимальной сети на вычислительную систему. Не менее естественным является вопрос, можно ли найти зависимость (в идеале отображение) позволяющую по описанию задачи определить тип и параметры оптимальной сети. То есть, в какой уже раз мы хотим перенести работу с наших плеч на виртуальные плечи искусственного интеллекта (ИИ), другими словами, нам бы хотелось найти подход, который позволил бы по заданным данным, критерию оптимальности (в данном случае речь идет о более общем критерии, чем целевая функция сети) и описанию задачи получать (суб)оптимальную сеть.

В данной статье рассмотрена проблема оптимизации ИНС. Мы обсуждаем возможности, предоставляемые для решения этой неформальной задачи современные достижения в области ИИ. Особое внимание уделено многоагентной парадигме как наиболее современной и в определенном смысле, наиболее адекватной, и предложено ряд концепций, которые могут быть использованы для решения поставленной задачи нахождения оптимальной ИНС в рамках поставленной задачи.

Оптимизация ИНС в рамках когнитивистской парадигмы

Как известно, экспертные системы (далее ЭС) способны решать неформализованные задачи, каковой и является подбор ИНС под задачу. Однако для создания такой ЭС нужен высококвалифицированный эксперт в данной проблемной области (выбора оптимальной ИНС под задачу), которого в настоящее время трудно найти, так как область не является устоявшейся и эксперты слишком сильно расходятся во мнениях. Как следствие, мы не можем применить все наработки когнитивистской парадигмы напрямую, однако, как будет показано, это не делает данную парадигму абсолютно бесполезной.

База данных экспериментов как источник знаний об оптимальной сети

В процессе работы с ИНС каждый разработчик создает, обучает и тестирует сотни ИНС. На данный момент научное сообщество накопило информацию о тысячах экспериментов, и большая часть этих данных находится в свободном доступе. Заметим, что каждый исследователь склонен работать в своей проблемной области, а значит, наиболее релевантные примеры он может почерпнуть даже из своей практики, не прибегая к внешним источникам информации. Из этого следует, что вполне разумным является создание базы данных экспериментов, которая может стать источником знаний об оптимальной сети для конкретной задачи. На данный момент Data Mining является развитой отдельной наукой, и его методы позволят создать набор правил или даже ЭС для создания оптимальных ИНС при наличии достаточно полной и репрезентативной базы данных.

На данный момент нами разработана и спроектирована нормализованная база данных экспериментов, которую планируется пополнять в ходе продолжающихся экспериментов по оптимизации ИНС под конкретные задачи.

Оптимизация ИНС в рамках бионической парадигмы

В рамках теории ИНС было разработано множество алгоритмов оптимизации параметров топологии. Их можно разделить на конструктивные и деструктивные. Конструктивные алгоритмы начинают работу с минимально возможной сетью (без скрытых слоев, а иногда и без выходного слоя, как в алгоритме расширяющегося нейронного газа) и продолжают наращивать нейроны, а подчас и слои, вплоть до достижения желаемого результата.

Очевидным недостатком этих методов является неспособность учесть возможность применения другой топологии, другого алгоритма обучения и т.д. Плюсом же является тот факт, что разработчик остается в рамках коннекционизма.

Генетическая оптимизация ИНС

Наибольшую эффективность как конструктивным, так и деструктивным подходам придают генетические алгоритмы (ГА) [Ragg, 1996]. Однако подобный подход не решает основной проблемы деструктивных и конструктивных алгоритмов, которая обозначена выше. Для того чтобы для определенного типа сетей решить все задачи оптимизации (выбор параметров топологии, алгоритма обучения, параметров

алгоритма обучения) используются метагенетические алгоритмы. Заметим, что, так как практика показывает, что ГА хорош в нахождении области притяжения глобального минимума, а нахождение самого минимума лучше возложить на алгоритмы обучения ИНС, оптимизироваться должны как веса ИНС, так и алгоритм обучения. Пример метагенетического алгоритма можно найти в работе [Abraham, 2002].

Вполне очевидно, что подобный подход позволяет находить (суб)оптимальные ИНС, но он может применяться лишь в случае, если процесс поиска оптимальной сети отделен от процесса применения в связи с большой вычислительной сложностью. Вторым недостатком является необходимость создания нескольких таких алгоритмов, если планируется сравнивать несколько различных типов сетей. Другими словами оптимизация по типу сети является невозможной в рамках генетической парадигмы.

Применение многоагентных систем для оптимизации ИНС

Поставим некоторую особь в рамки окружающей среды, которая соответствует поставленной задаче. Затем дадим особи знания об этой окружающей среде, своем месте в ней и других особях. Определим конкретную цель для особи. Важно также предоставить ей возможность обучаться, и делиться накопленными знаниями. Можно также добавить возможность случайных изменений, для решения проблемы локальных минимумов. Теперь мы можем определить приз (например, выживание) для лучших, и получим модель, позволяющую в рамках данной конкретной задачи получить оптимальный результат, оптимальную особь. При этом по желанию, мы можем все знания представлять в терминах проблемной области (далее ПО), а значит, мы получим и компоненту объяснения, и результат опять таки в терминах конкретной ПО. В данном случае речь пойдет не о функции фитнеса, а скорее о целой «задаче фитнеса», т.е. особь, решающая данную задачу быстрее, точнее, качественнее и будет искомой.

Проделав все вышеперечисленное, из простого ГА мы получим многоагентный алгоритм оптимизации. Таким образом, мы заменяем разработчиков агентами, условия задачи – средой и ставим перед агентами цель в рамках данной среды найти возможные решения задачи.

Метафора возможных сред обитания агентов

Рассмотрим мир А. Большое количество агентов живут в мире, и каждый агент имеет целью создать лучшую ИНС (все приведенные примеры основаны на принципе конкуренции, не исключающей ограниченной кооперации). Агент обладает знаниями о построении ИНС и о взаимосвязи различных компонент.

Предпочтения. Агент изначально имеет (обладает знаниями) всю информацию о мире, включая информацию об ассортименте каждого из магазинов. В мире есть

1) магазин входных данных (здесь получают данные, можно добавить выбор размерности вектора входов или выбор предобработчика);
2) магазин типов сетей (здесь выбирают тип сети);
3) магазин параметров топологии (здесь выбирают параметры топологии);
4) магазин алгоритмов обучения (здесь выбирают алгоритм обучения и его параметры);
5) арена тестирования (здесь проходит тестирование получившейся сети);
6) лаборатория облучения (с определенной вероятностью случайно изменяет предпочтения).

Ассортимент каждого из магазинов определяется условиями задачи. На нулевой итерации определяются «врожденные» предпочтения агента (его знания об оптимальной сети). В простейшем случае можно создавать агента без предпочтений или со случайными предпочтениями. С другой стороны можно изначально привить агенту «любовь», например, к РБФ сети. На каждой итерации каждый агент пробегает по всем магазинам, приходит на арену, ждет оставшихся агентов, проходит тестирование и разочаровывается или убеждается в своих выборах (значит в будущем он, скорее всего, не сделает такой же выбор (те же ошибки)). Результаты каждой итерации фиксируются. Возможно увеличение доверия к компонентам сетей победителей (5%-10% от популяции). После тестирования агент попадает в лабораторию. Компоненты в магазинах бесплатны.

После определенного числа итераций или по нахождению (суб) оптимальной сети работа системы останавливается, и результатами будут

1. База данных экспериментов.
2. Лучшая из найденных сетей.
3. Несколько близких конкурентов.
4. База знаний самых успешных агентов.

О возможности применения первой речь шла выше. Лучшую из найденных сетей можно

1. Объявить оптимальной и наиболее пригодной для данной задачи.
2. Исследовать на «устойчивость» – сравнить с близкими конкурентами, если среди них много сходных сетей, значит, данная сеть, скорее всего, будет хорошо работать на новых данных из проблемной области, в противном случае необходимо с большей долей недоверия относится к результатам эксперимента.
3. Включить в ансамбль вместе с ближайшими конкурентами и применять данный ансамбль.

База знаний лучшего агента может быть использована для генерации компоненты объяснения и для выбора оптимальной сети в схожей проблемной области в будущем.

Антирыночные цены. Рассмотрим мир В. Агент изначально имеет (обладает знаниями) всю информацию о мире, включая информацию об ассортименте каждого из магазинов и о ценах в каждом из магазинов. Более того, при изменении цен в любом магазине, агент сразу же узнает об этом. Мир В отличается тем, что любой товар в магазине имеет цену и агент не просто берет, а покупает товар. В мире В нет лаборатории облучения. Изначально каждый компонент в каждом магазине имеет свою цену. При отсутствии априорных знаний об оптимальной сети следует назначать одинаковые цены эквивалентным компонентам. В начале каждой итерации у каждого агента есть определенное количество ресурсов (это количество может уменьшаться от итерации к итерации), никак не связанное с прошлой итерацией. Агенты, решив задачу, о том, как потратить ресурсы, чтобы получилась сеть, проходят тестирование (можно добавить возможность обучения агентов, для оптимизации принимаемых ими решений). Чем лучше зарекомендовал себя компонент в тестировании, тем дешевле он становится в магазине (можно ввести систему скидок, для того чтобы учесть взаимосвязь между компонентами, то есть, если в наилучшей известной сети использовались два определенных компонента, то покупка второго будет дешевле номинала, при наличии первого). Таким образом, в будущем агенты будут более склонны покупать именно приносящие лучшие плоды компоненты. Время от времени цена случайных компонентов изменяется. Одним из результатов работы системы будет список цен, позволяющий определить наиболее адекватные компоненты.

Буржуазный мир. В мире С к объектам мира В добавлено казино, где случайным образом с определенной вероятностью изменяется капитал агентов. Возможно применение лаборатории облучения. На нулевой итерации определяются «врожденные» предпочтения агента (его знания об оптимальной сети) и начальный капитал каждого агента. Изначально каждый компонент в каждом магазине имеет свою цену. При отсутствии априорных знаний об оптимальной сети следует назначать одинаковые цены эквивалентным компонентам. Агенты, решив задачу, о том, как потратить ресурсы, чтобы получилась сеть, проходят тестирование (можно добавить возможность обучения агентов, для оптимизации принимаемых ими решений). Чем лучше результаты агента, тем больше денег он зарабатывает. Изменение предпочтений аналогично миру А. Чем лучше зарекомендовал себя компонент в тестировании, тем дороже он становится в магазине. Таким образом, одни агенты (богатые) будут развивать свой успех, двигаясь в изначально выбранном направлении, другие (бедные) будут вынуждены искать принципиальные новые решения (возможно лучшие), а значит, уменьшается вероятность «застрять» в локальном минимуме. После тестирования агент попадает в казино. Результаты работы системы аналогичны мирам А и В.

Заключение

В данной статье мы рассмотрели различные возможности оптимизации ИНС. Каждый разработчик вправе выбирать подход более адекватный его задаче. Однако, по выше названным причинам, мы считаем, что многоагентный подход является универсальным и при правильном применении способен находить (суб)оптимальное решение.

На очереди создание соответствующего программного инструментария, способного применить всю мощь многоагентных систем. Выбор из приведенных концепции также должен стать предметом будущих исследований. В силу отмеченной пользы, которую способна принести база данных экспериментов, будущий инструментарий планируется снабдить возможностями по автоматическому ведению и применению подобной базу. Подобная база данных сама по себе представляет интерес, так на ее основе может быть создана база знаний для оптимизации ИНС под конкретную задачу.

Библиографический список

[Abraham, 2002] A. Abraham. Optimization of Evolutionary Neural Networks using Hybrid Learning Algorithms // IEEE 2002 Joint International Conference on Neural networks. 2002. Volume 3. P. 2797-2802

[Gonzalez, 2000] S. Gonzalez. Neural Networks for Macroeconomic Forecasting: A Complementary Approach to Linear Regression Models, Working Papers, Department of Finance Canada. Available at: “http://www.machinelearning.net/ann/Gonz00.pdf”. 07/2000.

[Ragg, 1996] T. Ragg, H. Braun, H. Landsberg. A Comparative Study of Neural Network Optimization Techniques // In 13th International Conference on Machine Learning: Workshop Proceedings on Evolutionary Computing and Machine Learning, 1996, P. 111–118.

[Уоссермен, 1992] Ф. Уоссермен. Нейрокомпьютерная техника, М.: Мир, 1992

Информация об авторах

Кирилл Юрков *– Пермский Государственный Университет, студент; Россия, 614990, Пермь, ул. Букирев, д.15; e-mail: forfin@mail.ru*

NEAREST STRING BY NEURAL-LIKE ENCODING

Artem Sokolov

Abstract*: We analyze an approach based on distributed representations in the neural network paradigm to similarity preserving coding of symbol sequences and showed that it can be viewed as an embedding process. We also give conditions for obtaining binary and ternary vectors at the output that can be useful for a unified approach to representation and processing of various data types.*

Keywords *sequence similarity, edit distance, metric embeddings, distributed representations*

ACM Classification Keywords*: I.2.6 Connectionism and neural nets, E.m Miscellaneous*

Introduction

Edit distance (Levenshtein distance) [Levenshtein, 1966] is used in a large number of research areas from genetics and web-search to anomaly detection in network traffic and voice recognition. Taking into account the contemporary data lengths (millions and billions of symbols) that have to be dealt with in the mentioned areas, the classic $O(n^2)$-algorithm [Vintsyuk, 1968; Wagner, 1974] of its calculation is not applicable in practice.

These circumstances gave birth to a branch of information theory concerned with the acceleration of edit distance calculation or its approximation (see survey [Navarro, 2001]). An exponential increase in the characteristic

lengths of sequences, which are subject to comparison (the genome assembly, the need to compare data flows in information systems, etc.) urged interest to applications of the metric embedding theory (see survey [Indyk, 2004]). This theory is concerned with space mappings that simplify distance calculation [Indyk, 2001]. Levenshtein edit distance embedding to a vector space is an actual known open problem [Matoušek, 2002].

Independently, within the framework of the neural network paradigm of AI several approaches were proposed to the task of distributed representation and comparison of strings and other structured objects [Kussul, 1991; Rachkovskij, 2001]. Some approaches aimed at finding similarity of strings were presented in [Sokolov, 2005]. Here we develop one of them, namely, the approach based on the position-dependent thinning of vector representations, giving a theoretical grounding to the obtained scheme with the aid of probabilistic embedding of the edit metrics into Manhattan space.

Task Description

We seek for a way to effectively calculate Levenshtein edit distance with the help of vector representations or, more specifically, by embedding edit metrics to a vector space.

Our method belongs to the group of the so-called q-gram edit distance approximation methods (q-gram is a substring of length q), started by [Ukkonen, 1992]. We noted that the approach based on the distributed representations [Sokolov, 2005] partly resembles embedding or sketching methods [Cormode, 2000; Bar-Yossef, 2004; Batu, 2004] and therefore we attempted to combine both presenting the neural coding approach as an edit distance embedding into Manhattan space l_1. In order to show that the proposed method realizes one of the possible embedding definitions [Indyk, 2001], we will give the proofs of:

1) «upper bound», i.e. statements like $ed(x,y) \le k_1 \Rightarrow P[d(v(x),v(y) \le d_1] \ge p_1$ (1a)
2) «upper bound», i.e. statements like $ed(x,y) > k_2 \Rightarrow P[d(v(x),v(y) > d_2] \ge p_2$. (1b)

Mapping Description

For two input strings x,y of length n, we independently and equiprobably select a window of width w in both strings: $x[i,i+w-1]$ and $y[i,i+w-1]$. Using fixed parameters of q-gram length q_1 and q_2, a q-gram vector $v_{w,q}$ (vector of quantities of each q-gram) is composed for each window for each $q=q_1,\ldots,q_2$. The obtained vectors are concatenated for each string. The Manhattan distance d^{Σ} between them is the sum of Manhattan distances between q-gram vectors of the windows:

$$d^{\Sigma}(x[i,i+w-1], y[i,i+w-1]) = \sum_{q=q_1}^{q_2} \left\| v_{w,q}(x[i,i+w-1]) - v_{w,q}(y[i,i+w-1]) \right\|_{l_1} \quad (2)$$

Using the defined distance it is possible to show that necessary embedding properties (1a) and (1b) hold.

Upper Bound

For $x,y \in \Sigma^n$ let a q-gram $x[i,i+q-1]$ *be "good"*, if there is a q-gram $y[i,i+q-1]$ such that $|j-i| \le s$, where s is the relative shift of the q-grams. If all q-grams in a window $x[i,i+w-1]$ are «good» and are located successively in $x[i,i+w-1]$ as well as in $y[i,i+w-1]$, then $\|v_q(x[i,i+w-1]) - v_q(y[i,i+w-1])\| \le 2s$. As s does not exceed $ed(x,y)$, so $\|v_q(x[i,i+w-1]) - v_q(y[i,i+w-1])\| \le 2ed(x,y)$.

For the independent and equiprobable chose of the window we get for the property (1a):

Lemma 1 *For* $x,y \in \Sigma^n$, *if* $ed(x,y) < k_1$, $q_1 \le q_2 \le w \le (n+1)/(k_1+1)$, *then* $P[d^{\Sigma}(x,y) \le 2k_1(q_2-q_1+1)] > 1-k_1 w/(n-w+1)$.

Lower Bound

De Bruijn graphs. For a string x and a parameter q de Bruijn graph [Bruijn, 1946] $G[x;q]$ is a graph, whose vertices are all $q-1$-grams ($q-1$-spectrum) of the string x. An edge $a^1a^2\ldots a^q$ connects vertices labeled $a^1a^2\ldots a^{q-1}$ and $a^2a^3\ldots a^q$. Such graphs are widely used in genetics [Pevzner, 1989] and in cryptographic stream ciphers' analysis. Let a de Bruijn graph built from the union of $q-1$-spectra of two strings x,y be $G[x,y;q]$, and a path corresponding to a string x be π_x.

Let us consider possible local mutual configurations of paths π_x *and* π_y on *G[x,y;q]* that correspond to two symbolic string being compared. Let us call the right and left branching points the vertices where the ways, correspondingly, diverge or converge. To disambiguate their definition it is necessary that there is only one way of continuing a path from any graph vertex (Euler paths on *G[x;q]* and *G[y;q]* should be unique). Let a "loop" be a right branching point together with the nearest following left branching point in the direction of passing paths. Because of its definition, a loop does not contain identical arcs, otherwise it could be possible to divide it into smaller loops. The situation when there are no loops, when in *G[x,y;q]* is only one left or right branch point, or a left one and a right point following it, is called a "fork". With this configuration, a left folk compulsorily contains as least one of the starting arcs of at least one of the paths, and a right folk contains terminating arcs of at least one of the ways. For some π_x, π_y, let a "shift" be a special case of a fork, when there is a non-empty subpath $\pi_c, w \geq |\pi_c| \geq 0$, that $\pi_x = \pi'_x \pi_c$ and $\pi_y = \pi_c \pi'_y$, where π'_x, π'_y are some, possibly empty, subpaths.

Concepts of "transposition" and "rotation" are used to discriminate between ways of obtaining identical spectra from different strings ([12], [11]). A transposition is the following situation: for $t,s \in \Sigma^{q-1}$: x=… t … s…t s … ⇒ y=… t s…t … s … or x=… t … t t … ⇒ y=… t t … t … A rotation is a situation when *q-1*-grams on the edges of a string are identical. It this case a corresponding path on the de Bruijn graph is a cycle, and, from any of its vertices one can get different strings with the same spectrum.

The following theorem was conjectured in [Ukkonen, 1992] and proven in [Pevzner, 1995]

Theorem 1 *Any two strings with the same q-spectrum can be transformed into each other by transpositions and rotations.*

Corollary 1 *For a string* $x \in \Sigma^w$ *and graph G[x;q], if q>(w+1)/3 then the corresponding Euler path on G[x;q] is unique or is an Euler cycle.*

Corollary 2 *For two strings* $x,y \in \Sigma^w$ *such that they can transformed into each other* by rotations only*, holds* $ed(x,y) \leq 2\lfloor (w-q+1)/2 \rfloor$.

Our aim is to determine such a distance measure between two *w*-wide windows and such a threshold that strings with a distances less than this threshold would represent a shift and can be aligned in a fixed number of operations, by simply editing them at the beginning and the end of the window, namely, by eliminating forks at the edges of the windows. The usual q-gram distance (with a fixed *q*) cannot provide the desired result since for any *q* there can be found two strings *x,y* where on the graph *G[x,y;q]* there will be a loop.

Therefore we propose distance (2) and in the following lemma show that with its aid it is possible to determine whether it is possible to align windows with a small number of edit operations.

Lemma 3 *Let* $x,y \in \Sigma^w$, $q_2 > q_1 > (w+1)/3$, $\lfloor (w-q_1+1)/2 \rfloor \leq (\Delta q+1)(\Delta q+1)/2$, $\Delta q = q_2 - q_1$, $\Delta q \leq (((8w+11)/3)^{1/2}-3)/2$,

$$d^{\Sigma}(x,y) < (\Delta q+1)(\Delta q+2), \qquad (3)$$

then $ed(x,y) < (\Delta q+1)(\Delta q+2)$.

We checked lemma 3 experimentally for those values of parameter *w* that still allowed for brute force string comparison. For a binary alphabet, all pairs of 2^w strings were compared for *w = 8,…,17* (experiment ran for 4 days). For a ternary alphabet all pairs of 3^w strings were compared for *w = 8,…,10* (2 days). None of the experiments has found a pair of strings violating lemma.

Let there be two types of pairs of windows *x[i,i+w-1],y[i,i+w-1]*: «good» and «bad» – correspondingly those for which condition (3) holds or not. The next lemma says that it is possible to simultaneously align simultaneously successive «good» windows. Conditions of lemma 4 (with corollaries 1 and 2) exclude non-unique determination of branching points.

Lemma 4 *Let conditions of lemma 3 hold, w=8,…,n, if for all i=1,…,n-w+1 holds* $d^{\Sigma}(x[i,i+w-1],y[i,i+w-1]) < (\Delta q+1)(\Delta q+2)$*, than* $ed(x,y) < 2(\Delta q+1)(\Delta q+2)$.

Let *N* be the number of bad windows. Let us find the upper limit (lemma 5) on the edit cost for all possible arrangements of *N* windows, using the following string edit algorithm. Assume we have aligned strings up to position *j-1*. If all consecutive pairs of windows, beginning from position *j* and to *j+r*, are "good", then we align them with not more than *2Q* operations, using the result of lemma 4, and continue from position *j+r+w-1*. If the next pair of windows in position *j* is bad, we use one edit operation to replace symbol *x[j]* with symbol *y[j]*, thus aligning one symbol and continuing to the next pair of windows in position *j+1*.

Lemma 5 *Let conditions of lemma 4 hold. Let* $T=\lfloor (n-1)/w \rfloor$. *The cost of aligning* strings *y and x with the help of the above algorithm is upper bounded with*

$$(2Q-1)\min(T,N)+\min(N,n-1-(w-1)\min(T,N))+2Q \le 2Q(N+1)$$

So for the independent and equiprobable window choice we get the following lemma that specifies property (1b):

Lemma 6 *For* $x,y \in \Sigma^n$ *and holding conditions of lemmas 3-5, if* $ed(x,y)>k_2$, *then*

$$P[d^{\Sigma}(x,y) \ge (\Delta q+1)(\Delta q+2)] > (k_2/2Q-1)/(n-w+1).$$

Nearest Neighbor Search

In this part we consider one of the possible procedures for the nearest string search (NNS) with the help of edit distance approximation described above. Let $P=\{p_1,\ldots,p_P | p_i \in P\}$ be a collection of strings and p_0 be the input probe string, to which it is necessary to find the nearest one (by the edit distance) from P. Searching for the exact nearest neighbor is often a laborious task. Namely, for large dimensionalities dimensions of input space (in our case it is $d=|\Sigma|^n$) the existing NNS algorithms are reduced to the linear search on P. On the other side, the "approximately" nearest neighbor is often sufficient in applications and it is often much easier to find it. This caused a large number of works concerned with NNS approximations.

First we transform vectors $v_{w,q}$ into hash-values distributed around $||v_{w,q}||$ with the help of a p-stable distribution, and a modified scheme from [Datar, 2004] (see subsection "Modification"). Then we apply a known scheme of locality-sensitive hashing (LSH) [Indyk, 1998].

LSH. Let us describe the original [Indyk, 1998] LSH scheme applied to strings with the classic edit metrics. Define a ball with radius r containing points distanced from its center not farther then r: $B(t,r)=\{q: ed(q,s) \le r\}$.

Definition of locality-sensitive functions. A family of hash-functions $H=\{h: \Sigma \to X\}$ is called (r_1,r_2,p_1,p_2)-sensitive, if for any $x,y \in \Sigma^n$ and any independently an equiprobably chosen $h \in H$ holds the following:

$$t \in B(s,r_1) \Rightarrow P[h(t)=h(s)] \ge p_1 \quad \text{and} \quad t \notin B(s,r_2) \Rightarrow P[h(t)=h(s)] \le p_2, \quad r_1<r_2 \text{ и } p_1<p_2 \tag{4}$$

Compose random hash-vectors $g_j=(h_1,\ldots,h_K)$, $j=1,\ldots,L$ from functions h. Additionally, we create cells where we put a string $p_i \in P$ based on the value of the hash-vector $g(p_i)$: a string p_i is put into a cell with an identifier equal to the hash-vector value. The aim is to get high collision probability between nearby strings, and low probability between distant ones. Then, applying the same hash to the probe we check whether it equals one of the previously stored hashes of vectors from P: for probe p_0 we calculate all hash-vectors $g_i(p_0)$, $i=1,\ldots,L$ and examine corresponding cells. If some cell contains a string $p^*_i \in B(p,r_2)$, the algorithm returns YES and p^*_i and NO otherwise (thus representing a solution to the so called (r_1,r_2)-PLEB task [Indyk, 1998]). Algorithm terminates after checking $2l$ cells. For such procedure holds the following theorem. For such a procedure, the following theorem holds:

Theorem 2 [Indyk, 1998] *Let H be a* (r_1,r_2,p_1,p_2)*-sensitive family of functions,* $K=-\log|P|/\log(p_2)$, $L=|P|^{\rho}$, *where* $\rho=\ln(p_1/p_2)$. *Then the above algorithm solves* (r_1,r_2)*-PLEB task and takes* $O(|\Sigma|^q|P|+|P|^{1+\rho})$ *space,* $O(|P|^{\rho})$ *distance calculations, and* $O(|P|^{\rho}K)$ *calculations of hash functions.*

LSH with a 1-stable distribution. In [Datar, 2004], it is proposed to use the property of stable distributions that linear combinations of theirs random values ϕ_i are distributed as one random variable multiplied by the norm of linear combination's coefficients. Due to linearity of scalar product $(v_1,\phi)-(v_2,\phi) \sim ||v_1-v_2||_p\phi$. Hash-functions are defined as:

$$h(v)=\lfloor ((v,\boldsymbol{\phi})+b)/r \rfloor, \tag{5}$$

where b is an equiprobably distributed random variable on $[0,r]$, $\boldsymbol{\phi}$ is a vector with elements taken from Cauchy distribution.

If one divides a real axis into equal intervals, then, intuitively, vectors with the similar norm will likely fall into the same interval. It is possible to show [Datar, 2004] that for two fixed vectors the hash-function (5) is

$$p(c) = \int_0^r \frac{1}{c} f(c)(1 - \frac{t}{c})dt = \frac{1}{\pi}\left(2\arctan\left(\frac{r}{c}\right) - \frac{c}{r}\ln\left(1 + \left(\frac{r}{c}\right)^2\right)\right) \quad (6)$$

where *f(.)* is probability density function of the absolute values of ϕ, and $c=||v_1-v_2||$ is the distance between the hashed vectors. As *p(c)* is a monotonically decreasing function, the family of such functions is locality-sensitive (see definition 1). So, hash-functions (5) can be used in the LSH scheme.

Our modification. Let us propose a method of using hash-functions (4) in a different way from that described above, providing analogy to the distributed approaches [Sokolov, 2005]. Instead of multiplying a fixed vector *v* by a number of random vectors ϕ to form hash-vectors g_j, we take *K* random vectors $v^i_{w,q}(s)$ obtained by random and independent sampling with a window of width *w* from string *s* (see "Mapping description"). For each of them, we generate a separate random vector ϕ_i whose elements are taken from the Cauchy distribution. Let us designate the scalar product of these two vectors $h'_i=(v^i_{w,q}(s),\phi_i)$ and let us fix the hash-functions of form $h'(v)=\lfloor (h'_i+b)/r \rfloor$. Taking into account lemmas 1 and 6, holds the following lemma the following lemma holds indicating that the family of such functions is also locality-sensitive.

Lemma 7 *$P[h'(x)=h'(y)|ed(x,y)\le k_1]\ge p(2k_1(\Delta q+1))(1-k_1 w/(n-w+1))$, $P[h'(x)=h'(y)|ed(x,y)\ge k_2]\le 1-(k_2/(2(\Delta q+1)(\Delta q+2))-1)(1-p((\Delta q+1)(\Delta q+2)))$, where function p(.) is defined in (6).*

With this method of hash-function formation, it is possible to get output vectors without matrix multiplying of intermediate q-gram representations by random vectors ϕ, thus obtaining a scheme consistent with a neural network approach [Sokolov, 2005].

Binarization and ternarization. Parameter *r* in (5) can be chosen to minimize p_1/p_2 (see [Datar, 2004]) and to speed up the NNS procedure (theorem 2). It is also attractive to have either binary or ternary vectors at the output that are more beneficial than integer-valued ones because of a more efficient implementation. On the other side (sparse) binary or ternary vectors are widely used in distributed processing models [Rachkovskij, 2001].

Hash-function (5) will take binary values *{0,1}* if $0<h(v)<2$ and so $0<(\phi,v)<r$, and ternary values *{-1,0,1}* if $1<h(v)<2$ and so $-r<(\phi,v)<r$. Integrating $(\pi(1+x^2))^{-1}$ with limits from 0 to $r / ||v||$ and from $-r / ||v||$ to $r / ||v||$ we get for the probability of binary and ternary output, correspondingly, $1/\pi \arctan(r / ||v||)$ and $2/\pi \arctan(r / ||v||)$. Density of zero elements is an important parameter in many schemes of distributed coding, for ternary vectors it is defined by $1/\pi \arctan(r / ||v||-||v||/2r \ln(1+(r / ||v||)^2))$ and is increasing with the increasing of *r*.

Conclusion

We analyzed the concept of the distributed representations of sequences [Sokolov, 2005] from the point of view of metric embeddings, presented a new q-gram approximation method of the edit distance, and proved the possibility of constructing locality-sensitive functions. Thus we showed that the distributed representations used for the comparison of sequential data in the neural network paradigm could be justified with the aid of the methods from the embedding theory. This approach can also be considered as the substantiation of the Broder approach [Broder, 1995] who takes for the document similarity measure the degree of the coincidence of the sets of their q- grams and also other bag-of-grams methods. We also gave conditions for obtaining binary and ternary vectors at the output that can be useful for a unified approach to representation and processing of various data types and modalities [Rachkovskij, 2001].

Bibliography

[Bar-Yossef, 2004] Z. Bar-Yossef, T.S. Jayram, R. Krauthgamer, R. Kumar: Approximating Edit Distance Efficiently. *FOCS*, pp. 550-559, 2004

[Batu, 2004] T. Batu, F. Ergun, J. Kilian, A. Magen, S. Raskhodnikova, R. Rubinfeld, R. Sami. A sublinear algorithm for weakly approximating edit distance, In *Proc. 36th STOC*, 2004

[Broder, 1998] A. Broder. On the resemblance and containment of documents. In *SEQS: Sequences '97*, 1998.

[Bruijn, 1946] N. G. de Bruijn. A combinatorial problem. In *Koninklijke Nederlandsche Akademie van Wetenschappen*, volume 49, 1946.

[Cormode, 2000] G. Cormode, M. Paterson, S. C. Sahinalp, U. Vishkin. Communication complexity of text exchange. In *Proc. of the 11th ACM-SIAM Annual Symposium on Discrete Algorithms*, pp. 197--206, San Francisco, CA, 2000

[Datar, 2004] M. Datar, N. Immorlica, P. Indyk, V. Mirrokni. Locality-sensitive hashing scheme based on p-stable distributions. In *20-th annual symposium on Computational geometry*, pages 253–262, Brooklyn, New York, USA, 2004.

[Indyk, 1998] P. Indyk, R. Motwani. Approximate nearest neighbors: towards removing the curse of dimensionality. In *Proc. of 30th STOC*, pages 604–613, 1998.

[Indyk, 2001] P. Indyk. Algorithmic aspects of geometric embeddings. In *FOCS*, 2001.

[Indyk, 2004] P. Indyk. Embedded stringology. Talk at *15-th Annual Combinatorial Pattern Matching Symposium*, July 2004.

[Kussul, 1991] Kussul, E. M., & Rachkovskij, D. A. (1991). Multilevel assembly neural architecture and processing of sequences. In A. V. Holden & V. I. Kryukov (Eds.), Neurocomputers and Attention: Vol. II. Connectionism and neurocomputers (pp. 577-590). Manchester and New York: Manchester University Press.

[Levenshtein, 1966] V. I. Levenshtein. Binary codes capable of correcting deletions,insertions, and reversals. *Soviet Physics - Doklady*, 10(8):707–710, February 1966.

[Matoušek, 2002] Open problems. In J. Matoušek, editor, *Workshop on Discrete Metric Spaces and their Algorithmic Applications*, Haifa, March 2002.

[Navarro, 2001] G. Navarro. A guided tour to approximate string matching. *ACM Computing Surveys*, 33(1):31–88, 2001.

[Pevzner, 1989] P. A. Pevzner,P. L-tuple DNA sequencing: computer analysis. *J. Biomol. Struct. Dyn.*, 7, 63–73,1989

[Pevzner, 1995] P. A. Pevzner. Dna physical mapping and alternating eulerian cycles in colored graphs. *Algorithmica*, 13(1/2):77–105, 1995.

[Rachkovskij, 2001] D. A. Rachkovskij, E. M. Kussul. Binding and Normalization of Binary Sparse Distributed Representations by Context-Dependent Thinning, *Neural Comp*. 13: 411-452, 2001.

[Shamir, 2004] R. Shamir. Lecture notes in Analysis of Gene Expression Data, DNA chips and Gene Networks: Sequencing by hybridization. www.cs.tau.ac.il/~rshamir/ge/04/scribes/lec02.pdf, 2004.

[Sokolov, 2005] A. Sokolov, D. Rachkovskij. Some approaches to distributed encoding of sequences. In *Proc. of XI-th International Conference Knowledge-Dialogue-Solution*, volume 2, pages 522–528, Varna, Bulgaria, June 2005.

[Ukkonen, 1992] E. Ukkonen. Approximate string-matching with q-grams and maximal matches. *Theor. Comput. Sci.*, 92(1):191–211, 1992.

[Vintsyuk, 1968] T. K. Vintsyuk. Speech discrimination by dynamic programming. *Kibernetika (Cybernetics)*, (4):81–88, 1968.

[Wagner, 1974] R. A. Wagner, M. J. Fischer. The string-to-string correction problem. *Journal of the ACM*, 21(1):158–173, January 1974.

Authors' Information

Artem M. Sokolov – *International Research and Training Center of Information Technologies and Systems; Pr. Acad. Glushkova, 40, Kyiv, 03680, Ukraine; e-mail: sokolov (at) ukr.net*

DOMAIN ONTOLOGIES AND THEIR MATHEMATICAL MODELS[1]

Alexander S. Kleshchev, Irene L. Artemjeva

Abstract: *In this article the notion of a mathematical model of domain ontology is introduced. The mathematical apparatus (unenriched logical relationship systems) is essentially used. The representation of various elements of domain ontology in its model is considered. These elements are terms for situation description and situations themselves, knowledge and terms for knowledge description, mathematical terms and constructions, auxiliary terms and ontological agreements. The notion of a domain model is discussed. The notions of a precise ontology and precise conceptualization are introduced. The structures of situations and knowledge and also their properties are considered. Merits and demerits of various classes of the domain ontology models are discussed.*

Keywords*: Domain ontology, domain ontology model, ontology language specification, kernel of extendable language of applied logic, unenriched logical relationship systems, enriched logical relationship systems, enrichment of logical relationship system.*

ACM Classification Keywords*: I.2.4 Knowledge Representation Formalisms and Methods, F4.1. Mathematical Logic*

Introduction

A few different definitions for the notion of domain ontology have been suggested by now. But every definition has certain flaws. Because different interpretations of the notion of a domain ontology are used when different problems related to domain ontologies are solved, it may be deduced that now there is no universally accepted definition of the notion. This article suggests another definition of the notion of domain ontology. As this takes place, the mathematical apparatus (unenriched logical relationship systems) introduced in [1-3] is essentially used.

A Mathematical Model of a Domain Ontology

An unenriched logical relationship system [3] can be considered as a domain ontology model, if each of its logical relationship has a meaningful interpretation that a community of the domain agrees with, and the whole system is an explicit representation of a conceptualization of the domain understood both as a set of intended situations and as a set of intended knowledge systems of the domain. Some examples of unenriched logical relationship systems and their meaningful interpretations as models of simplified domain ontologies were given in [1-2]. Models of ontologies for medicine close to real notions of the domain were described in [5]. Models of ontologies for physical and organic chemistry and also for roentgen fluorescent analysis were described in [6-9]. Model of ontology for classical optimizing transformations is described in [10-12].

Information concerning a finite (real or imaginary) fragment of a real or imaginary reality (the fragment may be related to a finite part of the space and to a finite time lapse) will be called a situation (a state of affairs in terms of [4]), if this fragment contains a finite set of objects and a finite set of relations among them.

Objects and relations (including unary ones) among them depending on situations are designated by special domain terms which will be called terms for situation description. Objects in situation models can be represented:

[1] This paper was made according to the program № 14 of fundamental scientific research of the Presidium of the Russian Academy of Sciences, the project "Intellectual systems based on multilevel domain models".

by elementary mathematical objects (numbers and so on); by names having neither sort nor value [1] (such a name is a designation of an object); by structural mathematical objects (sets, n-tuples, and so on) constructed of elementary or structural mathematical objects or names having neither sort nor value by composition rules defined in the language of applied logic.

The set of names having neither sort nor value and used as designations of objects (and their components) in situation models can be determined explicitly or implicitly in a domain ontology model. In the former case, all these names appear in sort descriptions for unknowns. In the latter case, all these names are constituents of parameter values. In the domain ontology model the names of these parameters are used for describing sorts of unknowns. If a domain ontology model determines some names having neither sort nor value then these names have the same meaning in every situation of the domain. A domain ontology model can determine only some of the names having neither sort nor value and used in situations for designating objects (and their components). In this case these names are determined by a model of situation and may have different meaning in different situations.

Unknowns represent relations among objects depending on situations. In different situations the relations corresponding to the same unknown can be different. Every objective unknown designates a role that in each situation an (unique) object of the situation plays, and also in every situation there is its own object playing the role. Every functional unknown designates a set of functional relations. For each situation this functional relation is the one among objects of the situation. For different situations these relations corresponding to the same unknown can be different. Analogously, every predicative unknown designates a set of nonfunctional relations. For each situation this nonfunctional relation (it may be empty) is the one among objects of the situation. For different situations these relations corresponding to the same unknown can be different.

Thus, every unknown can be considered as a designation of a one-to-one correspondence between situations and the values of the unknown in these situations.

The sort description for an unknown determines the set of value models for the unknown. In any (real or imaginary) situation only an element of this set can be a value of the unknown. Thereby, the sort description for an unknown determines a model of the capacity for the concept designated by the unknown. A model of the capacity for a concept can be both a finite and infinite set.

A model of a (real or imaginary) situation is a set of values of the unknowns for the unenriched logical relationship system representing a domain ontology model. A model of a situation can be represented by a set of value descriptions for the unknowns.

Knowledge Models and Terms for Knowledge Description

If an unenriched logical relationship system is a model of a domain ontology then any of its enrichments is a model of a knowledge system for the domain. If a model of a domain ontology is an unenriched logical relationship system O without parameters, then the ontology model introduces all the terms for description of the domain. In this case any enrichment k of the system O is a set of logical relationships – restrictions on the interpretation of names representing empirical or other laws of the domain. Since this enrichment does not introduce any new names it cannot contain any sort descriptions for names [3].

If a model of a domain ontology is an unenriched logical relationship system with parameters, then the parameters of the system are the domain terms which are used for knowledge description.

If a model of domain ontology is a pure unenriched logical relationship system O with parameters then any enrichment k of the system O is a set α_P of the parameter values for the system O [3]. A value of an objective parameter determines a feature of the domain, a set of names for situation description, or a set of parameter names. Every enrichment (a knowledge base) can introduce new names as compared with the ontology – terms for situation and knowledge description. Functional and predicative parameters represent empirical or other laws of the domain. The value of every functional or predicative parameter is some relation among terms and/or domain constants. In this case domain knowledge is described at a higher level of abstraction than in the case when a domain ontology model is an unenriched logical relationship system without parameters. The values of parameters can be represented by a set of propositions – value descriptions for names.

If a model of a domain ontology is a mixed unenriched logical relationship system O with parameters, then any enrichment k of the system O is a pair $<\Phi', \alpha P>$, where Φ' is a set of logical relationships (restrictions on the interpretation of names) representing a part of empirical or other domain laws, and αP is a set of parameter

values for the system O representing the other domain laws [3]. In this case domain knowledge is represented at two levels of abstraction: as logical relationships among unknowns of the system O and as relations among terms of the domain (as parameter values of the system O).

The sort description for a parameter determines the set of value models for the parameter. In any knowledge model only an element of this set can be a value of the parameter. Thereby, the sort description for a parameter determines a model of the capacity for the concept designated by the parameter. A model of the capacity for a concept can be both a finite and infinite set.

Mathematical Terms and Constructions. Auxiliary Terms

The language of applied logic [1] determines mathematical terms and constructions used for domain description in that the unenriched logical relationship system which is an ontology model for the domain is represented. The kernel of the applied logic language [1] determines a minimal set of logical means for domain description. The standard extension of the language [1] apart from additional logical means introduces arithmetic and set-theoretic constants, operations and relations. Every specialized extension [2] of the language gives us a possibility to define both additional logical means and constants, operations and relations of other divisions of mathematics. The specialized extensions Intervals and Mathematical quantors of language [2] introduce integer-valued and real-valued intervals, and also mathematical quantifiers. Other examples of mathematical terms which can be introduced by specialized extensions are operations of differentiation and integration, predicates of optimization, and the like.

Mathematical objects (names, numbers, sets, n-tuples, and the like) serve to represent models of elementary and combined domain objects. Mathematical functions and relations represent the properties of domain objects which are kept with mathematical models in place of domain objects. In every domain a specific mathematical apparatus is used, as a rule. This property of domains is represented by specialized extensions of the language in domain ontology models. At the same time, the practice shows that the same mathematical apparatus can be used for description of different domains. In this case, for description of ontology models of these domains the same specialized extensions of the applied logic language given by the names of these extensions can be used.

Thus, mathematical terms and constructions have more or less universally accepted designations, syntax and semantics. They are separated from a domain ontology by their definition in the applied logic language (in its kernel and extensions) rather than in the unenriched logical relationship system representing the ontology model. They are associated with the domain ontology by the fact that the name of the logical theory representing the set of logical relationships contains the names of all the extensions used for description of this theory. Using mathematical terms and constructions with this interpretation does not constrain the possibility of unenriched logical relationship systems application for representation ontologies of different domains, and mathematics is among them. In the latter case mathematical terms and constructions play the role of elements of the metalanguage with completely defined syntax and semantics, and the other terms play the role of terms of (the domain) mathematics, their semantics being defined by an ontology.

Auxiliary terms are introduced to make a domain ontology description more compact. A value of an auxiliary term is defined by the values of other domain terms: of mathematical terms, terms for situation descriptions, terms for knowledge descriptions and other auxiliary terms. The definitions of auxiliary terms are represented by a set of value descriptions for names in a domain ontology model.

Ontological Agreements

Ontological agreements about a domain are represented by a set of restrictions on the interpretation of names of the unenriched logical relationship system which is an ontology model of the domain. Ontological agreements are explicitly formulated agreements about restrictions on the meanings of the terms in which the domain is described (additional restrictions on capacity of the concepts designated by these terms).

If a domain ontology model is an unenriched logical relationship system without parameters, then all the ontological agreements are only constraints of situation models. The set of ontological agreements, in this case, can be empty, too. If a domain ontology model is an unenriched logical relationship system with parameters then the set of ontological agreements can be divided into three nonintersecting groups: constraints of situation models, i.e. the agreements restricting the meanings of terms for situation description; constraints of knowledge models, i.e. the agreements restricting the meanings of terms for knowledge description; agreements setting up

a correspondence between models of knowledge and situations, i.e. the agreements setting up a correspondence between the meanings of terms for situation and knowledge description. Every proposition of the first group must contain at least one unknown or a variable whose values are unknowns and cannot contain any parameters; every proposition of the second group must contain at least one parameter or a variable whose values are parameters and cannot contain any unknowns; every proposition of the third group must contain at least one parameter or a variable whose values are parameters and at least one unknown or a variable whose values are unknowns. In doing so, the definitions of auxiliary terms should be taken into account.

Now let us define the informal notion of domain ontology using the formal notion of a domain ontology model. The part of information about a domain, which is represented by an ontology model of the domain, will be called an ontology of the domain. It immediately follows that a domain ontology contains a set of capacity concept definitions for situations description (it cannot be empty), a set of capacity concept definitions for knowledge description (it can be empty), characteristics of mathematical apparatus for domain description, a set of auxiliary term definitions (it can be empty), a set of restrictions on the meaning of terms for situation description (it can be empty), a set of restrictions on the meaning of terms for knowledge description (it can be empty), and a set of agreements setting up a correspondence between meanings of terms for situation description and for knowledge description (it can be empty).

A Domain Model

If an unenriched logical relationship system O is a domain ontology model and $k \in En(O)$ is a knowledge model of the domain, then the enriched logical relationship system <O, k> [3] is a model of the domain. In this case the set of solutions A(<O,k>) is a model of the domain reality. Thus, the domain ontology model O determines a class of the domain models $\{<O,k>|k \in En(O)\}$. Every domain model consists of two parts: an ontology model O that is the same for the whole class and a knowledge model k, which is specific for a particular domain model <O,k>.

The set of all possible situations in a domain which have ever taken place in the past, are taking place now and will take place in the future will be called the reality of the domain. Thus, the reality has the property that the persons studying the domain, the developers of its conceptualization and its models do not know the reality completely. Only a finite subset of situations forming the reality and having taken place in the past is known (although the information forming these situations also can be not completely known). We will suggest that relative to any conceptualization of a domain the hypothesis on its adequacy is true: the reality is a subset of the set of intended situations. In view of the reality definition it is evident that this hypothesis cannot be verified. Hence, every adequate conceptualization imposes certain limitations on the notion of the reality.

A(<O, k>) is an approximation of the unknown set of models of all situations which are members of the domain reality. It is apparent that the better A(<O, k>) approximates the reality the more adequate the domain model <O, k> is. A model of a domain is adequate to the domain, if the set of models of all the situations forming the domain reality is equal to the solution set of the enriched logical relationship system which is a model of the domain, i.e. the reality approximation is precise.

We will consider only such domain ontology models O that there is the adequate model <O, k> of the domain in the class of models of the domain determined by the ontology model O (the hypothesis on existence of the adequate domain model). The hypothesis on existence of the adequate domain model is stronger than the hypothesis on conceptualization adequacy. The first hypothesis states that there is such a knowledge base (an element of the set En(O)) that A(<O, k>) is the same as the set of models of all the situations of the domain reality. The second one states only that the latter set is a subset of the set of models of all the intended situations. Inasmuch as the reality is not completely known (not all the situations which took place in the past and take place at present are known, and no future situation is known either), it is unknown for any domain model how well the reality model approximates the reality. Thus, it is impermissible to hold about any domain model that it is an adequate model of the domain. At the same time, a criterion of inadequacy can be formulated: a model of a domain represented by an enriched logical relationship system is an inadequate model of the domain, if such a situation is known which took place in the reality that its model is not a solution of the logical relationship system. If a domain ontology model O is given, and inadequacy of a domain model <O, k> is revealed, then experts usually look for some other model of the domain knowledge $k' \in En(O)$, so that the domain model <O, k'> won't be inadequate with respect to the available data (known situations). If in the process of storing empirical data (extending the set of known situations) it becomes clear that inadequacy of the current domain model is

sufficiently often found, and that the model has to be permanently modified, and that this process leads to constant increasing of the number of empirical laws and/or to constant growth of complexity of the knowledge model, then an aspiration may arise for finding another conceptualization of the domain and an ontology representing it (changing the paradigm) and for finding an adequate model of the domain within the restrictions of the new conceptualization.

A Precise Ontology and Conceptualization

A domain ontology will be called precise, if the set of situation models forming the conceptualization represented by the ontology is equal to the set $\bigcup_{k \in En(O)} A(< O, k >)$, where O is an unenriched logical relationship system that is a model of the ontology, i.e. the approximation of the conceptualization determined by the ontology is precise.

A conceptualization will be called precise, if it is the same as the domain reality. It is apparent that precise conceptualizations are impossible for the domains related to the real world. But conceptualizations are possible for theoretical (imaginary) domains (mathematics, theoretical mechanics, theoretical physics and so on) for which their precision is postulated.

If an ontology and conceptualization are precise, then the unenriched logical relationship system O being a model of this ontology must have the following property: if <O, k> is the adequate model of the domain where $k \in$ En(O), then A(<O, k'>) $\subseteq$ A(<O, k>) for any $k' \in$ En(O). If O is an unenriched logical relationship system without parameters, then the empty set of propositions is this k.

The question arises of whether in the case of precise conceptualization the empty knowledge base is always consistent with the adequate domain model. Let us discuss this question using the example of an ontology of mathematics. An ontology of mathematics (or any one of its branches) consists of definitions and axioms. Any conceptualization of mathematics is assumed to be precise. At the same time, mathematical knowledge consists of theorems (lemmas, corollaries and so on) and their proofs. Since in mathematics any theorem is a logical consequence of the ontology, the theorems impose no additional restrictions on the reality model. Thus, both the empty knowledge base and a knowledge base containing any set of theorems determine adequate (and equivalent [1]) models of mathematics. The role of theorems is to make explicit the properties implicitly given by the ontology, and the role of proofs is to make evident the truth of theorems. Some theorems can have inflexible form (identities, inequalities and so on). So a mixed unenriched logical relationship system with parameters can be a natural ontology model for mathematics where terms identities, inequalities and others describe knowledge.

The Structure of Situations and Knowledge

The set of the unknowns whose values form a model of a situation will be called the structure of the situation model. We will say that models of two situations have the same structures, if the sets of the unknowns forming the structures of these situations are the same. From this point of view, the models of all the situations belonging to the reality model of any domain model have the same structures, if this domain model is an enriched logical relationship system. As for the structures of intended situation models determined by a domain ontology model that is an unenriched logical relationship system, three cases are possible.

1. A domain ontology model is an unenriched logical relationship system without parameters. In this case all intended situation models have the same structures.

2. A domain ontology model is an unenriched logical relationship system with parameters, none of parameter values being able to contain unknowns. In this case all intended situation models also have the same structures.

3. A domain ontology model is an unenriched logical relationship system with parameters, values of some parameters being able to contain unknowns. In this case the models of the situations belonging to the reality models of different models of the domain (consistent with different knowledge models) can have different structures depending on knowledge models.

The structures of all the situations determined by the ontology model of example 1 of article [3] are the same. They are formed by the unknowns diagnosis, partition for a sign, moments of examination, blood pressure, strain of abdomen muscles, and daily diuresis. The structures of all the situations determined by the ontology model of

example 6 of article [3] also are the same. They are formed by the unknowns cubes, balls, rectangular parallelepipeds, length of an edge, volume, substance, and mass.

The parameter signs in example 2 of article [2] contains unknowns (see propositions 2.2.1 and 2.2.13 in [2]). Thus, situation models determined by this ontology model can have different structures. In [2] an example of a knowledge model for this ontology model was given (see example 3, propositions from 3.1.1 to 3.1.9). The structure of situation models corresponding to that knowledge model is formed by the unknowns diagnosis, partition for a sign, moments of examination, strain of abdomen muscles, blood pressure and daily diuresis. If in another knowledge model of the same ontology model the parameter signs has the different value

signs ≡ {pain, temperature, discharge},

and the other parameters have some proper values, then the structure of situation models corresponding to this knowledge model is formed by the unknowns diagnosis, partition for a sign, moments of examination, pain, temperature and discharge, i.e. these structures differ from one another.

Using parameters whose values contain unknowns makes it possible “to hide” some terms used for situation description in domain ontology model description. At the same time, the meanings of these unknowns are completely determined by the propositions describing the sorts of these unknowns (see proposition 2.2.13 of example 2 of [2]): models of concepts designated by these unknowns are determined, for any unknown its meaning in a situation is determined (either the unknown is a name of a role, a functional relation or an unfunctional one), for every name of relation the number of its arguments, the sorts of its arguments and the sort of its result are determined.

The set of parameters of the unenriched logical relationship system being a domain ontology model will be called the structure of domain knowledge model. It follows from this definition that if an unenriched logical relationship system without parameters is a domain ontology model, then any knowledge model of this domain has no structure. If a mixed unenriched logical relationship system with parameters is a domain ontology model, then a part of any knowledge model has a structure but the other its part has no structure. If a pure unenriched logical relationship system with parameters is a domain ontology model, then all parts of any knowledge model of the domain have a structure. Let a domain ontology model be an unenriched logical relationship system with parameters. If no parameter value in its turn contains a parameter, then all domain knowledge models for this conceptualization have the same structures. If values of some parameters in their turn contain parameters, then different knowledge models of the domain can have different structures.

Using parameters whose values contain parameters makes it possible “to hide” some terms used for knowledge description in domain ontology model description. At the same time, the meanings of these terms are completely determined by the propositions describing the sorts of these terms.

A Comparison between Different Ontology Model Classes

Now let us discuss the question about capabilities of domain models and domain ontology models, which are enriched and unenriched logical relationship systems of different classes.

Let us consider several aspects of the term “domain ontology”.

1. If a conceptualization contains intended situations of different structures, then any ontology representing this conceptualization cannot have a model in the class of unenriched logical relationship systems without parameters but can have a model in the class of the systems with parameters.

2. If a conceptualization contains concepts designated by terms for knowledge description, then no ontology representing this conceptualization can have a model in the class of unenriched logical relationship systems without parameters, but it can have a model in the class of the systems with parameters.

3. If a conceptualization contains concept classes and determines properties of the concepts belonging to these classes, and concepts themselves are introduced by domain knowledge, then no ontology representing this conceptualization can have a model in the class of unenriched logical relationship systems without parameters but can have a model in the class of the systems with parameters.

4. If in a conceptualization some restrictions on meanings of terms for situation description depend on the meaning of terms for knowledge description, then any ontology representing this conceptualization cannot have a model in the class of unenriched logical relationship systems without parameters, but it can have a model in the class of the systems with parameters.

5. The more compactly and clearly domain ontology models of a class describe agreements about domains, the better the class is. In this regard unenriched logical relationship systems without parameters require for every term for situation description to appear explicitly in these agreements. For real domains (such as medicine) the models of their ontologies turn out immense because of large number of these terms. At the same time, the systems with parameters describe agreements about domains for groups of terms, rather then only for isolated terms through using terms for knowledge description. In doing so the majority of the terms for situation description and some terms for knowledge description do not appear explicitly in agreement descriptions (they are replaced by the variables whose values are terms from appropriate groups). As a result, a model of agreements becomes compact and agreements themselves become more general.

6. The more understandable knowledge bases represented in terms of an ontology are for domain specialists, the better the class of domain ontology models is. In this respect unenriched logical relationship systems without parameters represent knowledge bases as sets of arbitrary logical formulas. The more complex these formulas are, the more difficult it is to understand them. At the same time, the systems with parameters introduce special terms for knowledge description. The meanings of these terms are determined by ontological agreements, and their connection with terms for situation description among them. In real domains these terms are commonly used to ease mutual understanding and to make communication among domain specialists economical. The meanings of these terms are, as a rule, understood equally by all domain specialists. The role of these terms is to represent domain knowledge as relation tables (sets of atomic formulas, of simple facts). It is considerably easier for domain specialists to understand the meanings of these simple facts than the meanings of arbitrary formulas.

7. The more precise approximation of a conceptualization model a class of domain ontology models assumes, the better it is.

First, let us remark that it follows from the theorem about eliminating parameters of enriched logical relationship systems [3] that if there is a domain model represented by an enriched logical relationship system with parameters which determines an approximation of the domain reality, then there is a model of the domain represented by an enriched logical relationship system without parameters which determines the same approximation of the domain reality. In this regard domain models represented by enriched logical relationship systems with parameters offer no advantages over domain models represented by enriched systems without parameters.

As for domain ontology models, every one represented by an unenriched logical relationship system determines some approximations for both the set of intended domain situation models and for the set of intended domain knowledge models. If a model O_P of a domain ontology represented by an unenriched logical relationship system with parameters determines an approximation $\bigcup_{k\in En(O_P)} A(< O_P, k >)$ of the set of intended domain situation models, then the unenriched logical relationship system O_X without parameters quasiequivalent to O_P determines the approximation $\bigcup_{k\in En(O_X)} A(< O_X, k >)$ of the same set of intended situation models [1]. Let h: En(OP)→En(O_X) be the map defined by the theorem about eliminating parameters of unenriched logical relationship systems and H = {h(k) | k ∈ En(O_P)}. Then $\bigcup_{k\in En(O_X)} A(< O_X, k >) = \bigcup_{k\in En(O_P)} A(< O_X, h(k) >) \cup \bigcup_{k\in En(O_X)\setminus H} A(< O_X, k >)$; but by the theorem about eliminating parameters of enriched logical relationship systems $\bigcup_{k\in En(O_P)} A(< O_X, h(k) >) = \bigcup_{k\in En(O_P)} A(< O_P, k >)$, i.e. $\bigcup_{k\in En(O_X)} A(< O_X, k >) = \bigcup_{k\in En(O_P)} A(< O_P, k >) \cup \bigcup_{k\in En(O_X)\setminus H} A(< O_X, k >)$. Thus, the approximation of the set of intended situation models determined by the system O_X, is less precise than the approximation represented by the system O_P.

If a model O_P of a domain ontology represented by an unenriched logical relationship system with parameters determines an approximation En(O_P) of the set of intended domain knowledge models, then the unenriched logical relationship system O_X without parameters determines an approximation En(O_X) of the same set of intended knowledge models. In this case H is a subset of En(O_X), i.e. the approximation of the set of intended knowledge models determined by the system O_X also is less precise than the approximation determined by the system O_P. In what follows we show some reasons of this fact.

Let us consider the case when a domain ontology model is a pure unenriched logical relationship system O_P with parameters. First, the constraints of knowledge models represented by O_P determine the set $En(O_P)$ as a proper subset of the set of all possible interpretations of the system O_P's parameters, whereas, if the system O_X without parameters is a domain ontology model, then this ontology model contains practically no restrictions on the set $En(O_X)$. Second, for the theorem about eliminating parameters of enriched logical relationship systems a set of formulas representing empirical and other domain laws can be deduced from every proposition setting up a correspondence between knowledge models and situation models and from parameter values. These formulas contain no parameters. It is obvious that the forms of these formulas are restricted and determined by the forms of propositions setting up a correspondence between knowledge models and situation models. At the same time, if a domain ontology is an unenriched logical relationship system O_X without parameters, then this system imposes no restrictions on the form of formulas entering its enrichments.

Let us consider the case when a domain ontology is a mixed unenriched logical relationship system $O_P = <\Phi, P>$ with parameters. In this case, if $k \in En(O_P)$, then $h(k) = \Phi' \cup \Phi''$ where the propositions belonging to Φ' are deduced from every proposition of Φ setting up a correspondence between knowledge models and situation models and from parameter values (taking into account the parameter constraints) and Φ'' is such a set of propositions that $\Phi_X \cup \Phi' \cup \Phi''$ is a semantically correct applied logical theory where Φ_X is the set of all the propositions of Φ which contain no parameters, i.e. $H \subset En(O_X)$.

Domain ontology models represented by unenriched logical relationship systems with parameters are thus seen to offer certain advantages over domain ontology models represented by unenriched logical relationship systems without parameters (see also [13]).

Conclusions

In the article a notion “a mathematical model of a domain ontology” has been introduced, the representation of different elements of a domain ontology in this model – of terms for situation description and situations themselves; of knowledge and terms for knowledge description; of mathematical terms and constructions; of auxiliary terms, and ontological agreements has been considered. The structures of situations and knowledge and their properties have been considered. The notion "a domain model” has been discussed. Definitions of the notions “precise ontology” and “precise conceptualization” have been presented. Some merits and demerits of different domain ontology model classes have been discussed in details.

References

1. Kleshchev A.S., Artemjeva I.L. A mathematical apparatus for domain ontology simulation. An extendable language of applied logic // Int. Journal on Inf. Theories and Appl., 2005, vol 12, № 2. PP. 149-157. – ISSN 1310-0513.
2. Kleshchev A.S., Artemjeva I.L. A mathematical apparatus for ontology simulation. Specialized extensions of the extendable language of applied logic // Int. Journal on Inf. Theories and Appl., 2005, vol 12, № 3. PP. 265-271. – ISSN 1310-0513.
3. Kleshchev A.S., Artemjeva I.L. A mathematical apparatus for domain ontology simulation. Logical relationship systems // Int. Journal on Inf. Theories and Appl., 2005, vol 12, № 4. PP. 343-351. – ISSN 1310-0513.
4. Guarino N. Formal Ontology and Information Systems. In Proceeding of International Conference on Formal Ontology in Information Systems (FOIS’98), N. Guarino (ed.), Trento, Italy, June 6-8, 1998. Amsterdam, IOS Press, pp. 3- 15/
5. Kleshchev A.S., Moskalenko Ph. M., Chernyakhovskaya M.Yu. Medical diagnostics domain ontology model. Part 1. An informal description and basic terms definitions. In Scientific and Technical Information, Series 2, 2005, №12. PP. 1-7.
6. Artemjeva I.L., Tsvetnikov V.A. The fragment of the physical chemistry domain ontology and its model. In Investigated in Russia, 2002, 5, pp.454-474. http://zhurnal.ape.relarn.ru/articles/2002/042.pdf
7. Artemjeva I.L., Visotsky V.A., Restanenko N.V. Domain ontology model for organic chemistry. In Scientific and technical information, 2005, №8, pp. 19-27.
8. Artemjeva I.L., Restanenko N.V. Modular ontology model for organic chemistry. In Information Science and Control Systems, 2004, №2, pp. 98-108. – ISSN 1814-2400.
9. Artemjeva I.L., Miroshnichenko N.L. Ontology model for roentgen fluorescent analysis. In Information Science and Control Systems, 2005, №2, pp. 78-88. – ISSN 1814-2400.
10. Artemjeva I. L., Knyazeva M.A., Kupnevich O.A. Processing of knowledge about optimization of classical optimizing transformations // International Journal on Information Theories and Applications. 2003. Vol. 10, №2. PP.126-131. – ISSN 1310-0513.

11. Artemjeva I. L., Knyazeva M.A., Kupnevich O.A. A Model of a Domain Ontology for "Optimization of Sequential Computer Programs". The Terms for the Description of the Optimization Object. In Scientific and Technical Information, Series 2, 2002, № 12, pp. 23-28.(see also http://www.iacp.dvo.ru/es/)

12. Artemjeva I. L., Knyazeva M.A., Kupnevich O.A. A Model of a Domain Ontology for "Optimization of Sequential Computer Programs". Terms for Optimization Process Description. In Scientific and Technical Information, Series 2, 2003, № 1, pp. 22-29. (see also http://www.iacp.dvo.ru/es/)

13. Kleshchev A.S., Artemjeva I.L. Domain Ontologies and Knowledge Processing. Technical Report 7-99, Vladivostok: Institute for Automation & Control Processes, Far Eastern Branch of the Russian Academy of Sciences, 1999. 25p. (see also http://www.iacp.dvo.ru/es/).

Authors' Information

Alexander S. Kleshchev – *kleschev@iacp.dvo.ru*

Irene L. Artemjeva – *artemeva@iacp.dvo.ru*

Institute for Automation & Control Processes, Far Eastern Branch of the Russian Academy of Sciences

5 Radio Street, Vladivostok, Russia

ОНТОЛОГИИ И МУЛЬТИЛИНГВИСТИЧЕСКИЕ ТЕЗАУРУСЫ КАК ОСНОВА СЕМАНТИЧЕСКОГО ПОИСКА ИНФОРМАЦИОННЫХ РЕСУРСОВ ИНТЕРНЕТ

Юлия В. Рогушина, Анатолий Я. Гладун

Аннотация*: Определены подходы к анализу различных информационных ресурсов, пертинентных потребностям пользователей, на семантическом уровне – при помощи тезаурусов соответствующих доменов. Приведены алгоритмы формирования и нормализации мультилингвистических тезаурусов, а также методы их сравнения.*

Ключевые слова*: информационный ресурс, онтология, тезаурус, поиск информации.*

Введение

За последние годы Интернет превратился в одно из основных средств публикации информации. Это динамично изменяющаяся распределенная среда, а информационные ресурсы (ИР), представленные в ней, крайне разнородны. Эффективный поиск ИР в Интернет по мере увеличения объема и сложности сети становится все более сложным и трудоемким. При этом критичным является не столько время поиска, сколько отбор ИР, удовлетворяющих реальным информационным потребностям пользователей.

Оценка качества работы информационно-поисковых систем (ИПС) является достаточно сложным вопросом [1]. Проблема заключается в том, по каким параметрам оценивать ИПС. Большинство существующих методик анализируют такие параметры работы ИПС, как релевантность, полнота, точность и различные их соотношения. *Релевантность* – тематическое соответствие полученной в результате поиска информации запросу. *Полнота* поиска – это отношение количества правильно найденных документов к общему количеству релевантных запросу документов, известных ИПС. *Точность* поиска – отношение количества правильно найденных документов к общему количеству документов, выданных ИПС в ответ на запрос.

Однако следует учитывать, что формальный запрос к ИПС является попыткой пользователя формализовать свою информационную потребность и, к сожалению, не всегда точно отражает последнюю, что и приводит к снижению эффективности использования Интернета.для пользователя Поэтому важнее такой параметр оценки качества функционирования ИПС, как *пертинентность* – соотношение объема полезной для него информации к общему объему полученной информации. Для этого ИПС надо иметь сведения об области интересов пользователя, чтобы выбирать среди доступных ресурсов те, которые интересны ему, а не только формально соответствуют запросу. Такие сведения должны быть представлены в форме, пригодной для автоматической обработки и повторного использования, а их формирование необходимо автоматизировать.

Информационные ресурсы Интернет

Среди информационных ресурсов (ИР), потенциально доступных пользователям Интернет, по-прежнему преобладает текстовая информация, в основном, в формате HTML и XML, однако ее доля информации постоянно уменьшается за счет увеличения доли мультимедийных ИР. Составить представление о предметной области (ПрО), которую характеризуют эти ИР, можно двумя способами: анализируя непосредственно полнотекстовую информацию и рассматривая метаописания этих ИР.

Метаданные – информация о документе, которая понятна компьютеру. На сегодняшний день наиболее перспективной и общеупотребительной моделью описания метаданных является система описания ресурсов RDF (Resource Description Framework), созданная на основе XML. С помощью RDF можно описывать как структуру сайта, так и связанную с ним ПрО. RDF описывает ресурсы в виде ориентированного размеченного графа – каждый ресурс может иметь свойства, которые в свою очередь также могут быть ресурсами или их коллекциями. Наиболее распространённым наборов элементов для создания метаданных является Dublin Core Metadata Elements. Метаданные могут быть встроены в сам ИР либо хранится и обновляться независимо от ресурсов.

Мультимедийные данные. В последнее время в ИР, представленные в Интернет, наряду с текстовой информацией в них включается графика, видео, звук. На сегодняшний день существует значительное количество широко распространенных форматов для хранения аудио- и видеоинформации, 3D-сценариев и изображений. Мультимедийные ресурсы значительно хуже, чем текстовая информация, поддаются индексации. Если информация о мультимедийных ресурсах не представлена их поставщиками явным образом в каком-либо формате, известном средствам индексирования, то возникает необходимость в применении сложных и трудоемких операций (по распознаванию образов, речи и т.д.). В настоящее время группой MPEG разработан ряд стандартов для представление метаописаний мультимедийной информации (например, MPEG7 "Multimedia Content Description Interface" и MPEG21. Несмотря на значительные отличия мультимедийных ИР от текстовых, наиболее приемлемым для осуществления информационного поиска (с учетом времени его выполнения и объемов хранимой в индексной БД информации) представляется их описание с помощью тех же средств, что и текстовой информации: ключевые слова, размер и дата создания файла и т.д.

Web-сервисы. Изначально технология World Wide Web была ориентирована на работу со статичными гипертекстовыми документами, представленными в Интернет. Но затем в сети стали появляться сайты, предлагающие клиентам не только документы, но и услуги (например, сайты электронной коммерции). Многие такие сайты используют серверы приложений, которые не просто возвращают документ, а могут обрабатывать данные, введенные пользователем (запросы, заполненные формы и т.д.) и динамически генерировать документы в зависимости от указанных пользователем параметров. Такая динамическая составляющая Интернет растет значительно быстрее статичной и требует применения более сложных информационных технологий. В связи с этим можно рассматривать отдельный класс ИР – Web-сервисы.

Web-сервис – набор логически связанных функций, которые могут быть программно вызваны через Интернет. Это программа, идентифицируемая по URI, интерфейс которой может быть определен в виде XML-конструкций. Web-сервисы базируются на трех основных Web-стандартах: *SOAP* (Simple Object Access Protocol) — протоколе для посылки сообщений по протоколу HTTP и другим Интернет-протоколам; *WSDL* (Web Services Description Language) – языке для описания программных интерфейсов Web-сервисов; *UDDI* (Universal Description, Discovery and Integration) – стандарте индексации Web-сервисов.

Постановка задачи

Чтобы эффективно осуществлять поиск информации, необходимой пользователю (текстовых и мультимедийных документов, информационных услуг и т.д.), необходимо сформировать модель ПрО, интересующей пользователя (например, в виде онтологии), и использовать ее при выполнении ИПС запросов этого пользователя.

Тезаурусы и онтологии – средства представления знаний предметных областей

В каждой ПрО имеются явления, которые люди выделяют как концептуальные или физические объекты, связи и ситуации. С помощью различных языковых механизмов такие явления связываются с определенными дескрипторами (например, названиями, именными группами и т.д.).

Для успешного решения задачи поиска информации необходимо представить знания пользователя о той ПрО, которая его интересует, в некоторой форме, пригодной для автоматической обработки. Понятие ПрО относительно: спецификации ПрО более высокого уровня образуются путем интеграции схем ПрО более низких уровней. Важно достигнуть интероперабельности знаний ПрО. Онтологии являются именно такой формой представления знаний. Онтология – соглашение об общем использовании понятий, которое содержит средства представления предметных знаний и договоренности о методах соображений. Она может рассматриваться как определенное описание взгляда на мир в конкретной сфере интересов, который состоит из набора терминов и правил использования этих терминов, которые ограничивают их значение в рамках конкретной ПрО [2].

Онтология – БЗ специального вида с семантической информацией о некоторой ПрО. Это набор определений (на формальном языке) фрагмента декларативных знаний, ориентированный на совместное многократное использование различными пользователями в своих приложениях. В онтологии вводятся термины, типы и соотношения (аксиомы), описывающие фрагмент знания.

Онтологические обязательства - это соглашения относительно того, как согласованно и последовательно использовать общий словарь. Агенты (люди либо, например, программные агенты), совместно использующие словарь, не испытывают потребность в общей базе знаний: один агент может знать то, чего не знает другой агент, и агент, который обращается к онтологии, не требуется ответы на все вопросы, которые могут быть сформулированы с помощью общего словаря.

Любая ПрО с определенным предметом исследования имеет собственную терминологию, своеобразный словарь, использующийся для обсуждения характерных объектов и процессов, которые включают область. Библиотека, например, вовлекает собственный словарь, имеющий отношение к книгам, ссылкам, библиографиям, журналам и т.д. Таким образом, характер ПрО раскрывается в ее словаре – множестве слов, которые в ней используются. Ясно, однако, что характер области показан не только в соответствующем словаре. Кроме этого, необходимо (i) обеспечивать строгие определения грамматики, управляющей тем, как могут быть объединены термины словаря для формирования утверждений, и (ii) прояснить логические связи между такими утверждениями. Только когда эта дополнительная информация доступна, можно понять как природу объектов ПрО, так и важные отношения, установленные между ними. Онтология - структурированное представление этой информации [3].

Формальная модель онтологии O представляет собой упорядоченную тройку O=<X,R,F>, где X - конечное множество концептов (понятий, терминов) предметной области, которую представляет онтология O; R - конечное множество отношений между концептами заданной предметной области; F - конечное множество функций интерпретации, заданных на концептах и отношениях онтологии O.

До недавнего времени термин "тезаурус" использовался как синоним онтологии, однако теперь в ИТ с помощью тезауруса чаще описывают лексику ПрО в проекции на ее семантику, а онтологию применяют для моделирования семантики и прагматики в проекции на язык представления [4]. Модели как онтологий, так и тезаурусов включают в качестве основных понятия терминов и связей между терминами.

Слово "*тезаурус*" впервые использовал еще в XIII-м веке Б.Датини как название энциклопедии. В переводе с греческого "thesaurus" – сокровище, богатство. Согласно "Современному словарю иностранных слов": "*тезаурус – … полный систематизированный набор данных о какой-либо области знаний, позволяющий человеку или вычислительной машине в ней ориентироваться*".

Тезаурус – это словарь, в котором представлены дескрипторы определенной области знаний с систематизацией их иерархических и коррелятивных отношений; дескрипторы подаются в алфавитном порядке, но сгруппированные они по семантическому принципу; поиск осуществляется от понятия к слову.

Тезаурус можно рассматривать как частный случай онтологии. *Тезаурус* – это пара Th=<T,R>, где T - множество терминов, а R – множество отношений между этими терминами. Множества T и R конечны. Совокупность терминов, описывающих ПрО, с указанием семантических отношений между ними, является тезаурусом ПрО.

Мультиязычный тезаурус представляет собой согласованную совокупность одноязычных тезаурусов, содержащая эквивалентные дескрипторы на языках-компонентах, необходимые и достаточные для межъязыкового обмена, и включающая средства для указания их эквивалентности. При установлении эквивалентности дескрипторов различных одноязычных версий необходимо различать на разных языках-компонентах следующие степени эквивалентности терминов: 1) полная; 2) неполная; 3) частичная; 4) отсутствие эквивалентного термина. Неполными эквивалентами являются термины, для которых объемы выражаемых ими понятий пересекаются. Частичными эквивалентами являются термины, для которых объем понятия, выражаемого одним эквивалентом, входит в объем понятия, выражаемого другим эквивалентом. Одним из средств установления эквивалентности различной степени является использование онтологии соответствующей ПрО: каждое слово, входящее в один из одноязычных тезаурусов,должно ссылаться на один из терминов онтологии, что и помогает установить связи между словами различных тезаурусов. Так, слова "*книга*" (рус.), "*book*" (англ.) и "*buch*" (нем.) ссылаются на термин онтологии "книга", поэтому они эквивалентны. Слово "*book*" (англ.) ссылается на термин онтологии "*книга*", а слово "*manual*" (анл.) – на термин онтологии "*учебник*", являющийся подклассом "*книга*", поэтому между словами "*book*" и "*manual*" устанавливается отношение неполной эквивалентности.

Использование тезаурусов для поиска ИР

Для того, чтобы при поиске ИР, удовлетворяющим информационным потребностям пользователя, учитывать семантику интересующей его области, необходимо (рис.1):

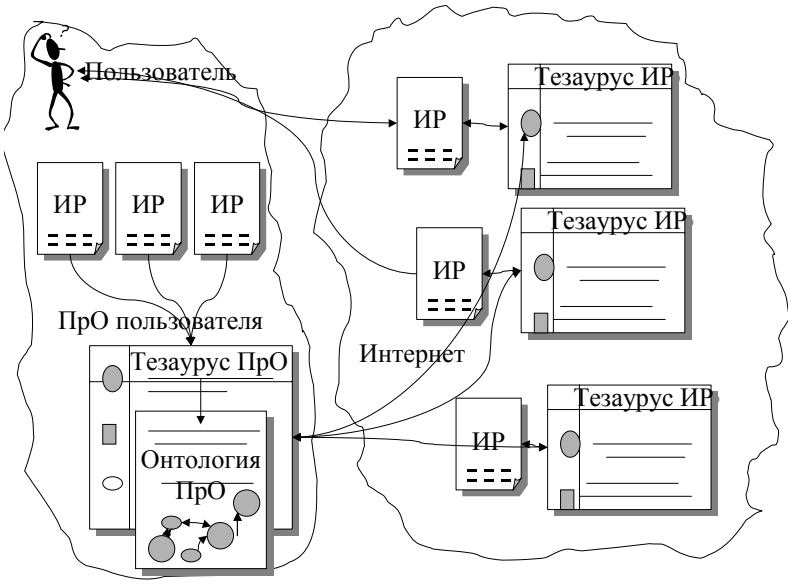


Рис.1. Процедура информационного поиска на основе нормализованных тезаурусов

1. сформировать тезаурус ПрО, соответствующей информационным потребностям пользователя (на основе анализа ИР, которое пользователь считает релевантными этой ПрО) [5];
2. для каждого ИР, известного ИПС, построить тезаурус (в данном случае – простой словарь, не содержащий стоп-слов);

3. провести сравнение тезауруса ПрО с тезаурусами ИР, релевантных запросу пользователя к ИПС (например, по ключевым словам) и найти те из них, в которых встречается наибольшее количество соответствий значимых слов.

При построении тезаурусов необходимо использовать онтологии соответствующих областей (более высокого уровня по сравнению с ПрО пользователя, чтобы нормализовать мультиязычные тезаурусы). Так как все эти тезаурусы строятся с точки зрения пользователя (которая отражена в онтологии интересующей его ПрО), то их построение является его задачей.

Построение тезауруса ПрО

Вначале пользователь должен самостоятельно отобрать множество ИР, которые он считает релевантными ПрО, которая его интересует. Каждый ИР характеризуется непустым множеством связанных с ним текстовых документов – метаописаний, результатов индексирования, своим контентом и т.д. Тезаурус ПрО формируется в результате автоматизированного анализа этих документов (действия пользователя сводятся к тому, чтобы построить *семантические пучки* – указать, на какой термин онтологии ПрО ссылается каждое из слов формируемого тезауруса). Алгоритм построения тезауруса ПрО состоит из следующих шагов:

1. Формирование исходного множества текстовых документов, характеризующих ПрО. На вход алгоритма поступает множество текстовых документов A, характеризующих выбранные ИР (каждый из них может иметь свой коэффициент значимости и коэффициент релевантности ИР, что позволяет по-разному определять вес слов из этих документов для характеристики ИР).

2. *Создание информационного пространства ПрО.* Для каждого документа из A $a_i \in A, i = \overline{1,n}$ строится тезаурус $T(a_i)$ – словарь, в котором содержатся все слова, встречающиеся в документе a_i. Тезаурус ИР строится как объединение тезаурусов $a_i : T_{ИР} = \bigcup_{i=1}^{n} T(a_i)$, а тезаурус ПрО – как объединение тезаурусов ИР.

3. Очистка тезаурусов. Пользователь должен указать для каждого $a_i \in A, i = \overline{1,n}$, словарь, содержащий стоп-слова (например, предлоги и союзы языка, на котором написан документ, являются для него стоп-словами, но предлоги и союзы другого языка, используемые как примеры, к ним не относятся) $s_j, s_j \in Voc$. Слова, содержащиеся в $s_j, s_j \in Voc$, необходимо удалить из тезаурусов. Затем отбрасывается вся служебная информация (для гипертекста, например, это теги разметки). Таким образом формируются очищенные тезаурусы $T`(a_i), \forall p \in T(a_i) \Rightarrow p \in T`(a_i) \vee p \in s_j, T`(a_i) \cap s_j = \varnothing$. Очищенный тезаурус ИР строится как объединение очищенных тезаурусов $a_i : T_{ИР} = \bigcup_{i=1}^{n} T(a_i)\ T`_{ИР} = \bigcup_{i=1}^{n} T`_{ИР}(a_i)$, а очищенный тезаурус ПрО – как объединение тезаурусов ИР.

4. Связывание тезауруса с онтологией ПрО. Чтобы интегрировать обработку слов, имеющих одинаковую семантику (например, синонимы, переводы термина на различные языки, разнообразные виды написания), тезаурусу ПрО ставится в соответствие некоторая онтология O (пользователь может формировать ее самостоятельно, использовать готовую онтологию либо ее модификацию и т.д.).

Для каждого слова из тезауруса надо установить ссылку на один из терминов онтологии (если связь отсутствует, то слово считается стоп-словом либо элементом разметки и должно быть отброшено): $\forall p \in T`(a_i) \exists t = Term(p, O) \in T_O$. В дальнейшем группа терминов тезаурусов ИР, связанная с одни термином онтологии, будет называться *семантическим пучком* $R_j, j = \overline{1,n}$ и рассматриваться как единое целое. $\forall p \in T`_{ИР} \exists R_j = \{r : r \in T`_{ИР}, Term(p, O) = Term(r, O)\}$

Это позволяет интегрировать обработку семантики документов, информация в которых представлена на различных языках, и, таким образом, обеспечить мультилингвистический анализ информационных ресурсов в сети Интернет.

5. *Расширение онтологии.* Если в тезаурусе обнаружены слова, для которых невозможно установить ссылку в онтологии, однако пользователь считает эти слова значимыми, то необходимо расширить онтологию, введя в нее соответствующие термины, указать их связи с другими терминами онтологии и вновь вернуться к шагу 4.

6. *Построение нормализованного тезауруса ПрО,* т.е. объединения всех терминов онтологии ПрО, с которыми установлена связь слов из нормализованного тезауруса ИР (рис.2.):

Нормализованный тезаурус представляет собой проекцию множества слов ИР на множество терминов ПрО. $L_{ИР} = \{t : p \in T(a_i), i = \overline{1,n}, t = Term(p, O) \in T_O\}$, а нормализованный тезаурус ПрО – объединение нормализованных тезаурусов ИР (рис.3.).

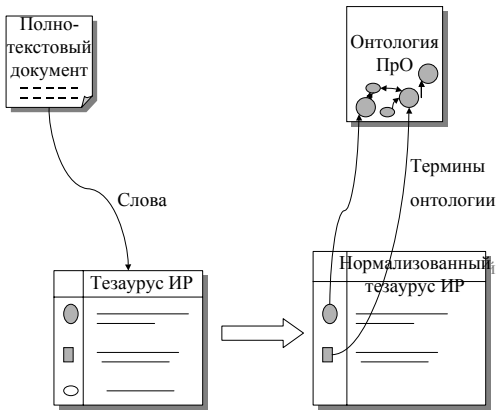


Рис.2. Построение нормализолванного тезауруса ИР

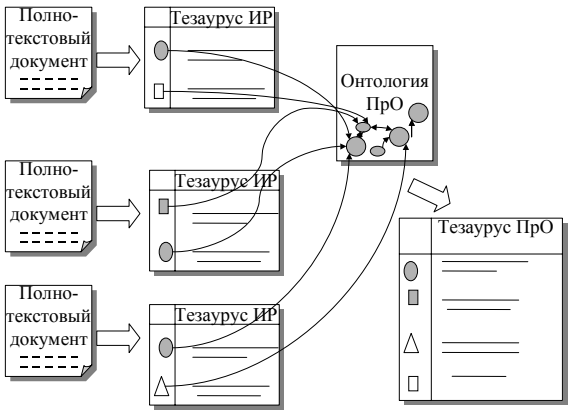


Рис.3. Формирование тезауруса ПрО

Построение тезауруса ИР

Тезаурус ИР, найденного ИПС в результате выполнения запроса пользователя, является упрощенным. Это простой словарь, который не содержит отношений между словами (извлечение таких связей из текста достаточно сложно и в данном случае не оправданно).

Алгоритм построения тезауруса $ИР_i$ состоит из следующих шагов:

1. *Формирование исходного множества ИР* U, $U = \{ИР_j, j = \overline{1,m}\}$.
2. *Формирование тезаурусов ИР* из U. Для каждого из ИР строится и очищается тезаурус,
3. *Построение нормализованных тезаурусов ИР:* при нормализации используются семантически пучки, сформированные пользователем при формировании тезауруса ПрО.

Алгоритм сравнения тезауруса ИР с тезаурусом ПрО

Нормализованные тезаурусы ИР и тезаурус ПрО представляют собой подмножества терминов онтологии ПрО O, выбранной пользователем: $L_{ИР} \subseteq Term(O)$, $L_{ПрО} \subseteq Term(O)$. Можно предположить, что тот ИР, описания которого содержат больше слов, для которых удалось установить соответствие с терминами ПрО, которые в данный момент интересуют пользователя (что отражено в нормализованном тезаурусе ПрО), в большей степени может удовлетворить информационные потребности пользователя, чем другие ИР, также релевантные тому же формальному запросу к ИПС. Таким образом, необходимо найти ИР q, такой, что $f(q, L_{ПрО}) = \max f(L_{ИР}, L_{ПрО})$, где функция f определяется как количество

элементов в пересечении множеств $L_{ИР}$ и $L_{ПрО}$: $f(A,B)=|A\cap B|$. Если различные термины нормализованных тезаурусов имеют для пользователя различное значение, то можно использовать соответствующие весовые коэффициенты, позволяющие учитывать их значимость. Тогда оценочная функция имеет следующий вид: $f(A,B)=\sum_{j=1}^{z} y(t_j)$, где функция у определена для всех терминов онтологии ПрО и $y(t_j)=\begin{cases}0, t_j\notin A\vee t_j\notin B\\ w_j, t_j\in A\wedge t_j\in B\end{cases}$.

Заключение

Предложенный в работе подход к использованию онтологии предметной области для создания и нормализации тезаурусов информационных ресурсов позволяет производить поиск интересующей пользователя информации на семантическом уровне, абстрагируясь от языка описания ресурсов. При этом использование тезаурусной меры информации позволяет предлагать пользователю только те сведения, которые будут ему понятны, что обеспечивает пертинентность информационного поиска.

Литература

1. Методика оценки эффективности систем информационного поиска / Выборнова О.Е., Завьялова О.С., Осипов Г. С., Смирнов И.В., Тихомиров И.А. // Сб.трудов VI международн.конф. "Интеллектуальный анализ информации ИАИ-2006", К.: Просвіта, 2006. – С.215-226.
2. Гаврилова Т.А., Хорошевский В.Ф. Базы знаний интеллектуальных систем. - Спб.: Питер, 2001.
3. IDEF5 Method Report. Knowledge Based Systems, Inc.1408 University Drive East College Station, Texas 77840, 1994. – 175 pp.
4. Нариньяни А.С. Кентавр по имени ТЕОН: Тезаурус + Онтология. – http://www.artint.ru/articles/narin/teon.htm.
5. Рогушина Ю.В., Гладун А.Я. Онтологический поход к мультилингвистическому анализу информационных ресурсов в сети Интернет // Сб.трудов VI международн.конф. "Интеллектуальный анализ информации ИАИ-2006", К.: Просвіта, 2006. – С.237-246.

Информация об авторах

Рогушина Юлия Витальевна – *Институт программных систем НАНУ, Киев-187 ГСП 03680, просп. акад. Глушкова, 40, e-mail: jjj_@ukr.net*

Гладун Анатолий Ясонович – *Международный научно-учебный центр информационных технологий и систем НАНУ, Киев-187 ГСП 03680, просп. акад. Глушкова, 40, e-mail: glanat@yahoo.com*

ЯЗЫК МНОГОУРОВНЕВОГО ОНТОЛОГИЧЕСКОГО МОДЕЛИРОВАНИЯ

Сергей Шаврин

Резюме*: В данной статье предлагается язык многоуровневого онтологического моделирования O_2ML, ориентированный на использование в системах, управляемых метаданными. В статье рассматриваются существующие языки и подходы к онтологическому моделированию, после чего предлагается рассмотреть новый язык, объединяющий в себе их сильные стороны.*

Ключевые слова*: Метамоделирование, информационные системы, языки моделирования.*

Классификация АСМ*: H.0 Information Systems - General.*

Введение

При разработке информационных систем коллективом разработчиков создается целый ряд артефактов, который обычно включает модель предметной области, документацию пользователя, программный код, набор тестов и т.д. Эффективность работы компании в краткосрочной перспективе зависит от наличия инструментальных средств, позволяющий облегчить и по возможности автоматизировать процесс создания и использования этих артефактов. Однако средне- и дальнесрочная эффективность во многом зависит от того, насколько универсальны создаваемые компанией артефакты.

Общепринятым способом универсализации, а, следовательно, и продления срока жизни создаваемых артефактов, является повышение уровня абстракции. Однако абстрагирование увеличивает семантический разрыв между артефактом и машиной, что приводит к необходимости выполнения трансляции. Как известно, существует два типа трансляторов: компиляторы и интерпретаторы. По принципу компиляторов работает подавляющее большинство современных CASE-средств. Преимуществом такого подхода является то, что процесс трансляции выполняется один раз до начала эксплуатации системы, что позволяет сэкономить ресурсы компьютера. Однако, системы, построенные по принципу интерпретатора, обладают большей гибкостью, что в современных условиях представляется более ценным свойством.

Естественным кандидатом на роль «управляющей программы» информационной системы, построенной по принципу интерпретатора, является модель предметной области. В этом случае необходимо, чтобы система понимала и могла исполнять модели, описанные на некотором языке моделирования. Наиболее распространенным на сегодняшний день языком является UML [7]. В данный момент в OMG (Object Management Group) ведутся работы по созданию второй версии этого языка и сопутствующих стандартов. Опубликованы еще не все необходимые спецификации, однако уже сейчас можно говорить о том, что была проделана огромная работа по формализации семантики UML, что существенно облегчает создание виртуальной UML-машины. На Рис. 1 приведен пример UML-модели.

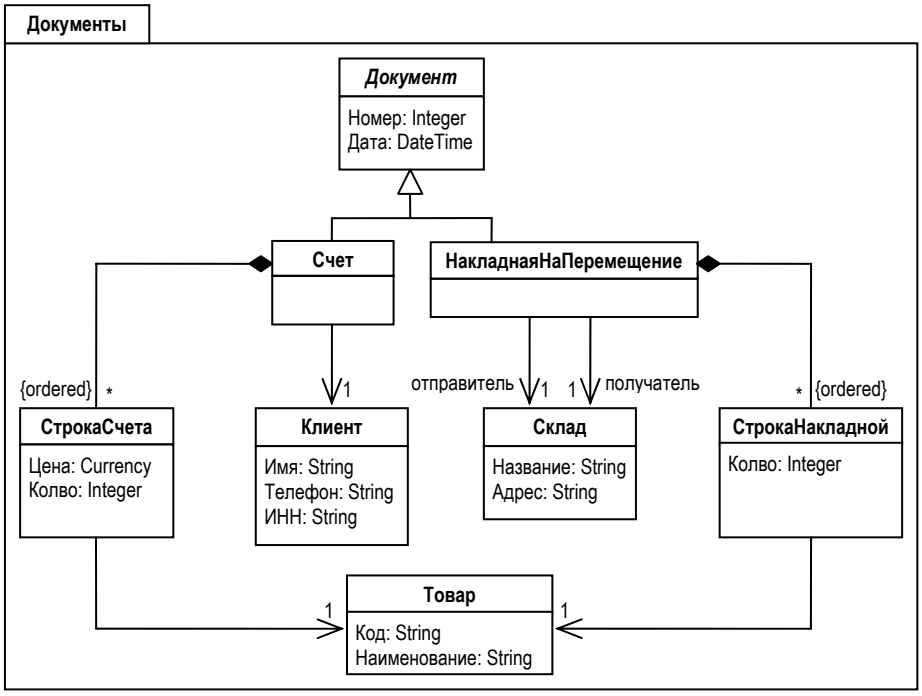


Рис. 1. Пример UML-модели

Использование модели предметной области в качестве основного артефакта позволяет, при соответствующей инструментальной поддержке, существенно повысить эффективность работы компании. Однако, как показывает опыт, модель предметной области так же, как и любой другой артефакт, подвержена изменениям, которые, впрочем, гораздо проще выполнить, нежели внести изменения в программный код. С другой стороны, для смежных предметных областей зачастую используются похожие модели, отличающиеся лишь в деталях. В данной ситуации компания может повысить свою

эффективность в средне- и долгосрочной перспективе используя метамодели, описывающие в меньшей степени подверженные изменениям метааспекты, общие для целого ряда смежных предметных областей.

С точки зрения метамоделирования UML предоставляет весьма ограниченные возможности, основанные на использовании стереотипов и помеченных значений (tagged values). В данном случае необходим язык, позволяющий описывать полноценные метасущности и поддерживающий произвольное число метауровней.

Проект OMEGA

OMEGA [4] – Ontological Metamodeling Extension for Generative Architectures – это проект по расширению MOF [6] (Meta Object Facility – метамодель языка UML) с целью поддержки онтологического метамоделирования. OMEGA ориентируется на генерацию кода.

В контексте рассматриваемой проблемы проект OMEGA интересен тем, что определяет ряд понятий, делающих возможным полноценное онтологическое метамоделирование. Основными среди них являются метакласс, метаатрибут и метаассоциация. Следует отметить, что метаатрибут в данном случае понимается не как атрибут метакласса, а как полноценная метасущность, экземплярами которой являются традиционные атрибуты. Кроме того, существует возможность управления допустимым количеством экземпляров метаатрибута. Это позволяет моделировать такие особенности предметной области, как, например, «Документ каждого типа имеет ровно один числовой атрибут, представляющий номер документа; не менее одного атрибута-даты и несколько атрибутов-реквизитов» (см. Рис. 2 и Рис. 3).

Однако OMEGA имеет два недостатка, важных с точки зрения рассматриваемой проблемы. Во-первых, в силу того, что OMEGA базируется на MOF, этот проект наследует все его особенности. В частности, MOF ориентируется на описание языков, таких как UML и CWM [5], и не обладает рядом возможностей (которые, впрочем, имеются в UML), полезных при моделировании предметных областей информационных систем. А именно, MOF и, следовательно, OMEGA не поддерживают множественной классификации и ортогональной специализации, каковые, по мнению автора, являются весьма полезными инструментами моделирования.

Другим существенным недостатком является то, что семантика OMEGA определена в большинстве своем неформально, что существенно осложняет построение OMEGA-машины.

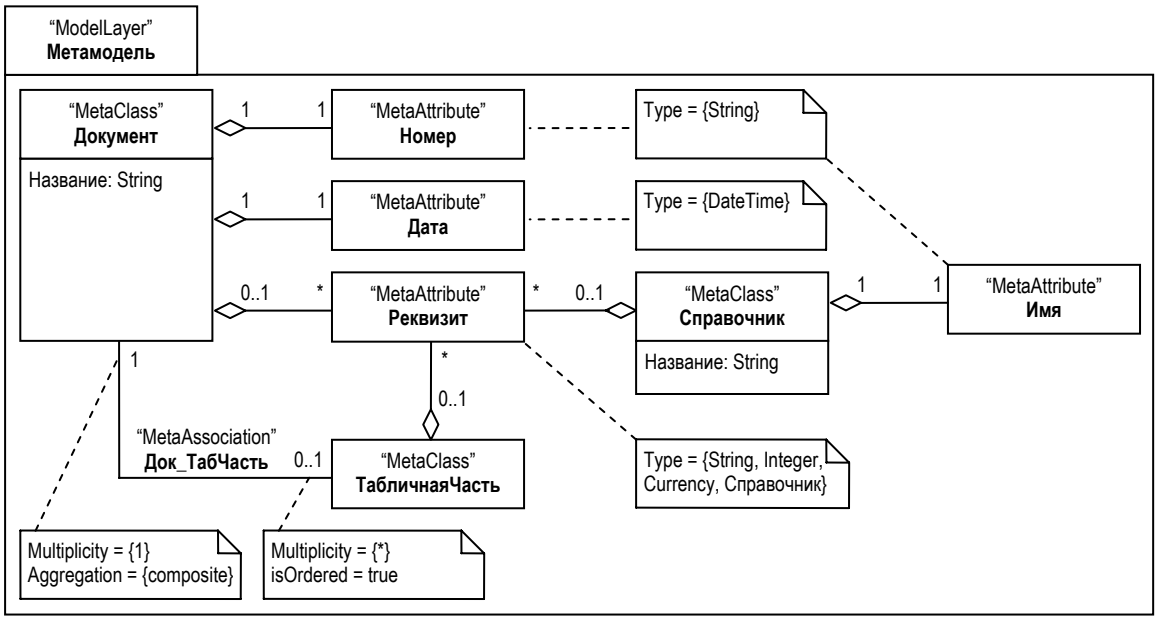


Рис. 2. Пример OMEGA-метамодели

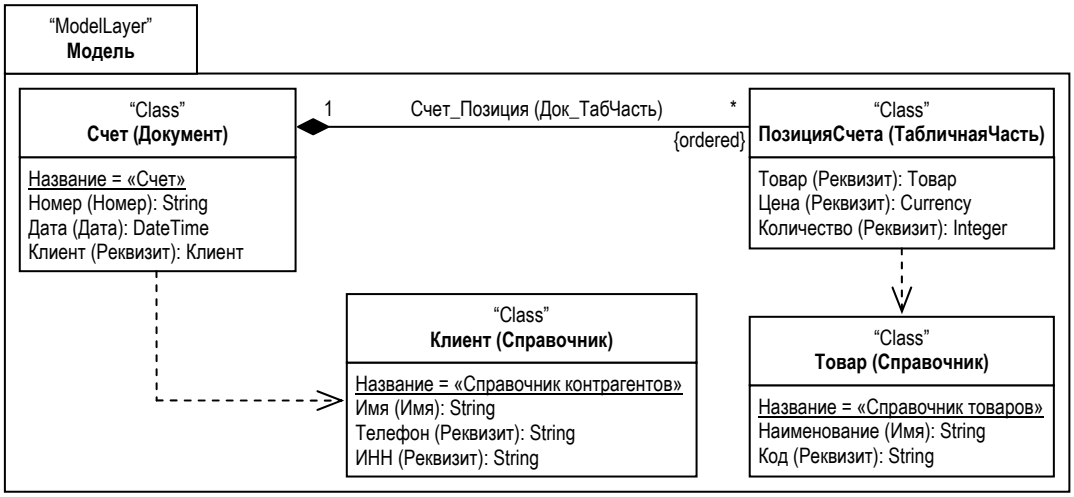


Рис. 3. Пример OMEGA-модели

Глубокое порождение Аткинсона и Кюхне

Говоря о порождении (создании экземпляров) обычно подразумевают мелкое порождение (shallow instantiation). Т.е. экземпляр создается в соответствие с определением своего класса. Другими словами, при определении класса делаются утверждения относительно его экземпляров. Очевидно, что в рамках двухуровневой модели «класс-экземпляр» другого понимания порождения и быть не может. Однако, распространение мелкого порождения на многоуровневый случай может привести к ряду проблем. В частности возникают проблемы дублирования понятий и неоднозначной классификации [1,2].

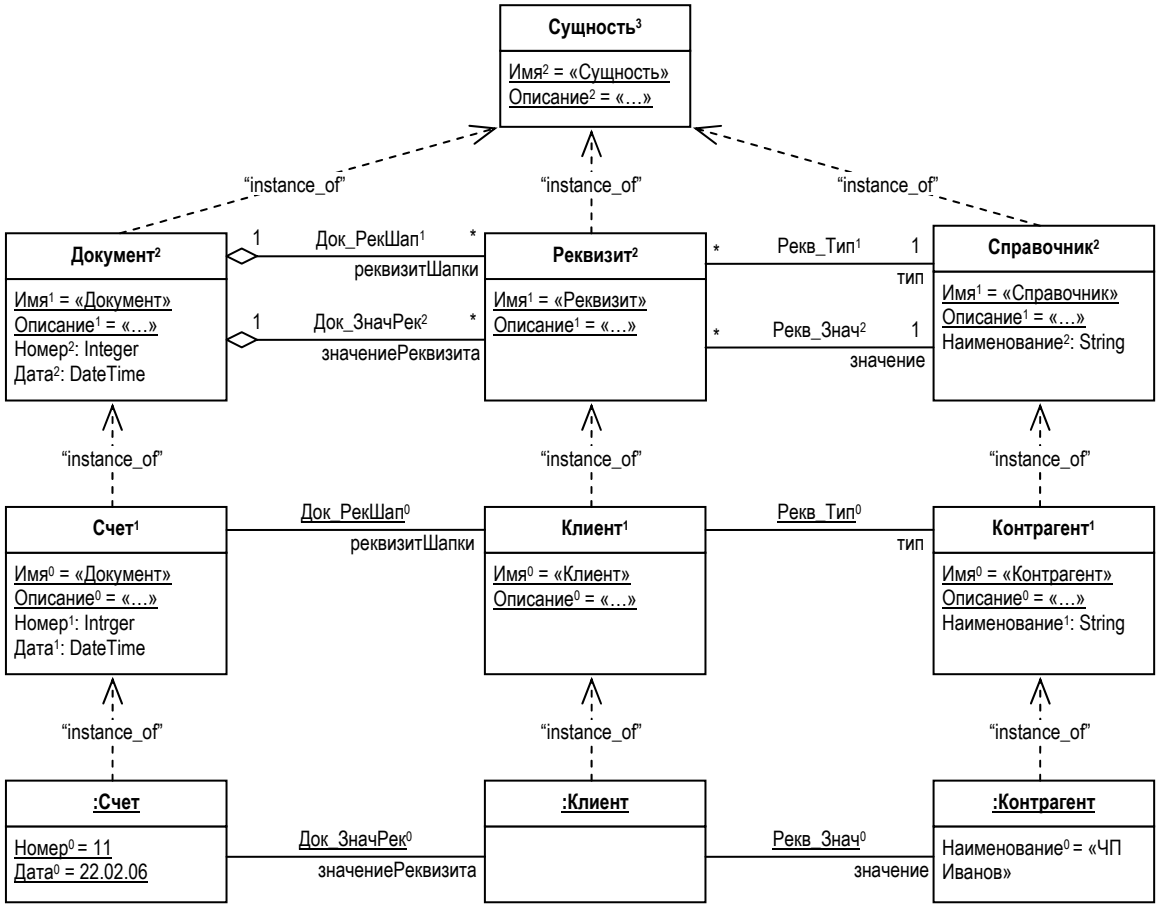


Рис. 4. Пример использования потенциалов

Для выхода из сложившейся ситуации Аткинсон и Кюхне предлагают обобщить мелкое порождение до глубокого (deep instantiation) [2]. В данном случае появляется возможность делать утверждения не только об экземплярах, но и об экземплярах экземпляров. Для этого предлагается с каждым элементом модели связывать его потенциал – число, определяющее возможное количество порождений элемента. Например, класс с потенциалом 2 (метакласс) при порождении превращается в класс с потенциалом 1 (традиционный класс) и далее в класс с потенциалом 0 (объект). Аналогично, атрибут с потенциалом 2 превращается в атрибут с потенциалом 1 (традиционный атрибут), который в свою очередь становится атрибутом с потенциалом 0 (слот). Пример использования потенциалов приведен на Рис. 4.

Кроме потенциалов, Аткинсон и Кюхне в своей работе [2] вводят понятие двойственного поля (DualField) – объект, обладающий семантикой и поля и слота. Другими словами, двойственное поле – это слот с ненулевым потенциалом. Примерами двойственных полей на Рис. 4 являются поля «Имя» и «Описание».

Не смотря на то, что введение потенциала позволяет избежать указанных выше проблем, возникающих при переходе к многоуровневому моделированию, этого явно не достаточно для решения практических задач. Предлагаемый в [2] язык, поддерживающий потенциалы, является слишком простым, его возможностей не хватает для решения реальных задач. Необходимы дополнительные инструменты метамоделирования, такие как метаатрибуты и метаассоциации.

Язык O_2ML

O_2ML – это рабочее название разрабатываемого автором языка многоуровневого онтологического моделирования, полученное из следующей цепочки: «Ontological Multi-Level Modeling Language → OMLML → O2ML → O_2ML».

В основе O_2ML лежит следующие три вещи:

- UML – богатые возможности «внутри-уровневого моделирования»;
- OMEGA – широкие возможности «межуровневого» моделирования;
- глубокое порождение – поддержка многоуровневого моделирования.

На Рис. 5, Рис. 6 и Рис. 7 приведен пример использования языка O_2ML. Из рисунков видно, что использование потенциалов позволяет сократить количество метаатрибутов, что приводит к более компактным и простым моделям. С формальной точки зрения атрибут с потенциалом $n > 1$ – это метаатрибут с потенциалом $n - 1$, удовлетворяющий следующим ограничениям:

- множество допустимых типов ограничено только одним типом;
- количество экземпляров в каждом классе-потомке класса-владельца равно одному;
- имена экземпляров совпадают с именем самого метаатрибута.

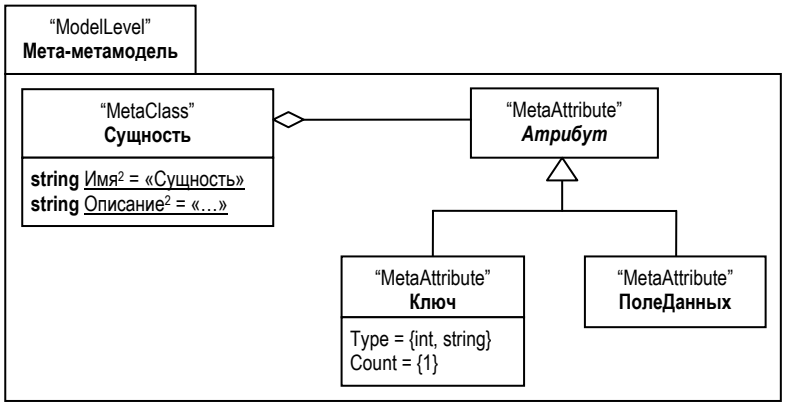


Рис. 5. Пример O_2ML-мета-метамодели

В графической нотации O_2ML используется отличное от UML описание атрибутов следующего вида: <Тип> <Имя> <Потенциал>: <Имя метаатрибута>. Такой подход согласуется с тем, что метаатрибут является классификатором соответствующих ему атрибутов.

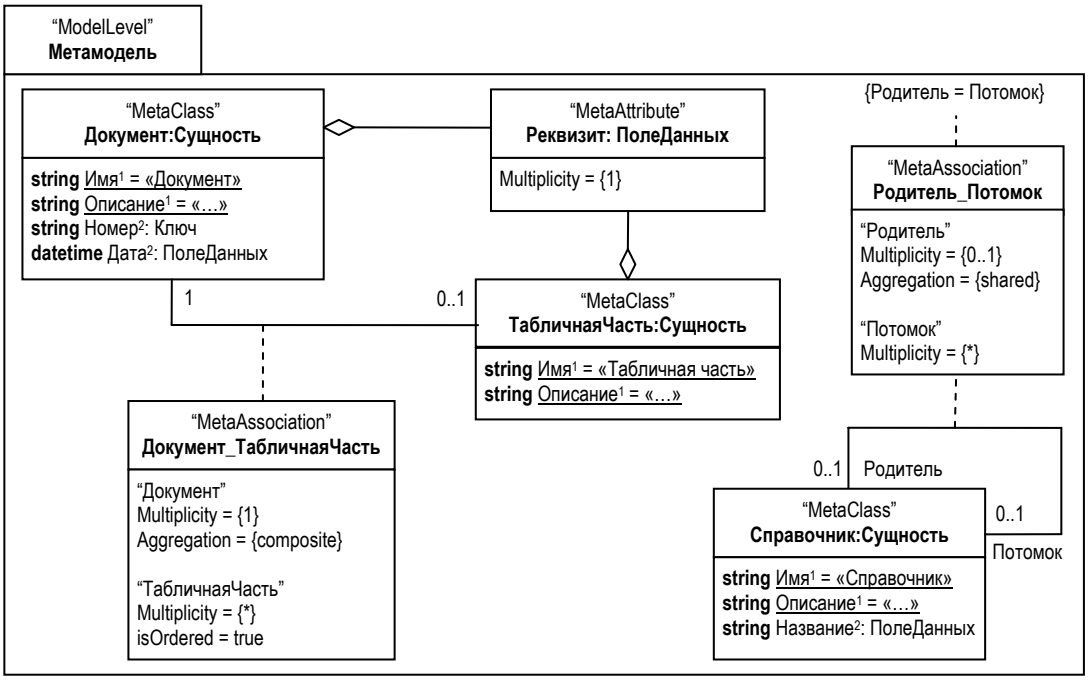


Рис. 6. Пример O_2ML-метамодели

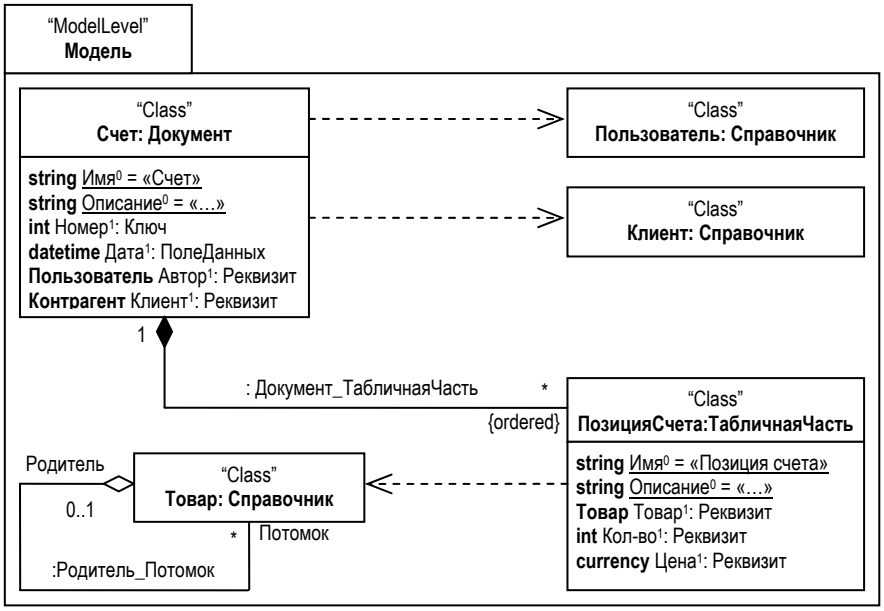


Рис. 7. Пример O_2ML-модели

Особенностью O2ML является то, что он обладает полностью формально определенной семантикой, что делает возможным непосредственное построение O_2ML-машины. Для описания семантики используется язык XOCL (eXecutable OCL), который расширяет возможности OCL и является частью XMF (eXecutable Metamodeling Framework) [3].

Библиографический список

[1] Atkinson C., Kühne T. Rearchitecting the UML Infrastructure. ACM Transactions on Modeling and Computer Simulation, Vol. 12, No. 4, October 2002.

[2] Atkinson C., Kühne T. The essence of multi-level metamodeling. In Proceedings of the Fourth International Conference on the Unified Modeling Language, M. Gogolla, C. Kobryn, Eds., Lecture Notes in Computer Science, vol. 2185, 19–33, 2001.

[3] Clark T., Evans E., Sammut P., Willans J. Applied Metamodelling: A Foundation for Language Driven Development http://albini.xactium.com/web/downloads/b1a35960appliedMetamodelling.pdf, 2004.

[4] Gitzel R., Ott I., Schader M. Ontological Metamodel Extension for Generative Architectures (OMEGA), Working Paper, University of Mannheim, Department of Information Systems III, http://www.bwl.uni-mannheim.de/Schader/_files/gitzel-omega.pdf, June, 2004.

[5] Object Management Group, Common Warehouse Metamodel, http://www.omg.org/technology/cwm, 2001.

[6] Object Management Group, Meta Object Facility Core v2.0, http://www.omg.org/cgi-bin/doc?formal/2006-01-01, January 2006.

[7] Object Management Group, UML Superstructure Specification v2.0, http://www.omg.org/cgi-bin/doc?formal/05-07-04, July 2005.

Информация об авторе

Сергей Шаврин *– Пермский государственный университет, старший преподаватель; e-mail: shavrin@gmail.com*

САМООРГАНИЗАЦИЯ ПРОЦЕССА ФУНКЦИОНИРОВАНИЯ ЭКСПЕРТНОЙ СИСТЕМЫ С ИСПОЛЬЗОВАНИЕМ ОНТОЛОГИИ ПРЕДМЕТНОЙ ОБЛАСТИ

Елена Нетавская

Аннотация*: Оптимизация процессов проектирования, создания, функционирования и сопровождения экспертных систем является важной задачей теории искусственного интеллекта и методов принятия решений. В статье предложен подход к ее решению с использованием технологии, базирующейся на методологии системного анализа, онтологии предметной области, принципах и методах самоорганизации. Изложены аспекты реализации такого подхода, базирующиеся на построении соответствия между иерархической структурой онтологии и последовательностью вопросов в автоматизированной систем контроля знаний.*

Ключевые слова*: Экспертная система, онтология, самоорганизация*

Введение

Экспертные системы (ЭС) относятся к основным и наиболее ранним направлениям искусственного интеллекта. Первые ЭС, базирующиеся на использовании вычислительной техники, известны с 60-х годов прошлого столетия. Сегодня сложно назвать отрасль науки или производства, где бы они не использовались. Вместе с тем, стремительное движение по этапам

<общество без границ>→<информационное общество>→<общество, базирующееся на знаниях>

и лавиноподобный рост количества информации значительно усложнили адекватное использование ЭС, что связано с информационной избыточностью, неполнотой и нечеткостью информации, ее субъективностью. Возникли проблемы представления знаний экспертов, оптимизации процессов получения заключений в ЭС, определения полноты информационных баз.

Типичным представителем ЭС являются автоматизированные системы контроля знаний (АСК3) по дисциплинам, связанным с информационными технологиями. Не останавливаясь на задачах верификации тестов и оценки результатов отметим, что в них в полной мере отражены все вышеуказанные проблемы. В дальнейшем изложении будем руководствоваться выводом [Feigenbaum, 1963] о том, что “основным принципом инженерии знаний является то, что возможности решателя задач интеллектуального агента в первую очередь определяются его информационной базой и лишь во вторую – используемым методом вывода”.

Анализ, характеристика и составные части процесса создания экспертных систем

Предпосылкой и побудительным мотивом получения новых знаний является идея или необходимость. В случае ЭС такая необходимость заключается в получении некоторого вывода, являющимся одним из определяющих факторов при принятии решений. С помощью АСКЗ оценивают уровень знаний. Будем иллюстрировать процесс создания ЭС при помощи трех методологических структурных единиц, которыми являются онтология (О), системный анализ (СА) и самоорганизация (СО).

Следуя этапам алгоритма СА, в соответствие с [Плотинский, 1992], [Згуровский, 1997], определим цель создания ЭС как средства, атрибутом которого является способность накапливать экспертные знания и впоследствии возможность заменить экспертов в процессах принятия решений. Задачи, решаемые для достижения цели, определяются предметной областью. В частности, для АСКЗ такими являются контроль знаний, определение их уровня и, частично, их приобретение. Функционирование ЭС определяется внешней средой, из которой поступает информация в виде тем, вопросов, возможных ответов, правил вывода и в которую передается результат – оценка уровня знаний.

Как сложную систему, ЭС можно представить тремя моделями [Тимченко, 1991]: строения, функционирования и развития. Модель строения является теоретико-множественной моделью:

$$M_b =< I_t, I_q, I_a, R_o >, \quad (1)$$

в которой отображен элементный базис системы: I_t – информационная таблица тем контроля; I_q – таблицы данных вопросов (в зависимости от типов вопросов их может быть несколько); I_a – таблицы возможных ответов с указанием градации их правильности; R_o – совокупность правил вывода, иногда представляемую некоторой процедурой или алгоритмом. Модель функционирования определяет процесс достижения цели системой, который осуществляется ее элементами, подсистемами, целостной ЭС:

$$M_f =< O_t, O_q, Q_a, P_1, P_2, ..., P_n, A >, \quad (2)$$

где O_t – динамические операции, сопровождающие процесс выбора темы; Q_q – динамические операции выбора множества вопросов; Q_a – динамические операции формирования множества возможных ответов и их оценки; $P_i, i = \overline{1,n}$ – совокупность операций, реализующих последовательность переходов при тестировании; A – алгоритм оценивания знаний. Модель развития отражает движение ЭС, обладающей атрибутами открытости, мобильности, системного и информационного единства, комплексности по этапам ее жизненного цикла:

$$M_d =< A_d \Theta A_m \Theta A_u, B_r >, \quad (3)$$

где Θ – логическая дизъюнкция или конъюнкция, A_d – процедуры адаптации к изменению внешних условий; A_m – процедуры модернизации и использования новых технологий; A_u – процедуры частичной или полной утилизации; B_r – механизмы обратной связи, позволяющие с учетом будущих процессов производить изменения в ЭС на всех этапах ее жизненного цикла.

Модели (1)-(3) являются дополнительным информационным фактором, позволяющим осуществлять структуризацию процесса создания и функционирования ЭС. Заметим, что эффективно функционировать будет та ЭС, как уже указано выше, которая имеет адекватно сформированный информационный базис. В его качестве рационально использовать онтологию – достаточно сложную организованную структуру знаний о предметной области с одной стороны, с другой – исходный материал для получения новых знаний [Гречко, 2005]. Построение и использование онтологии в ЭС базируется на том, что:

- онтология в таком случае совместно используется коллективами агентов;
- знания о предметной области используются неоднократно;
- знания о предметной области отделены от процесса и алгоритма экспертизы;
- она необходима для анализа знаний о предметной области.

Разработка и использование онтологий на сегодняшний день не формализованы, для их построения существуют только некоторые фундаментальные правила. Вместе с тем, области применения онтологий

и аспекты их развития достаточно разнообразны, о чем свидетельствует анализ работ 2005 года. Так, в [Гречко, 2005] построена онтология метода анализа иерархий Саати; в статье [Даревич, 2005] для повышения интеллектуального анализа текста предложено использовать взвешивание понятий в модели онтологии; авторы [Кучеренко, 2005] рассматривают вопросы, связанные с представлениями в онтологии нечетких понятий и отношений; статья [Шалфеева, 2005] посвящена исследованию оценивания в процессе создания онтологий. Алгоритм сравнения интересов продавцов и покупателей в Интернет-магазинах предложен в работе [Гладун, 2005]; в статье [Gribova, 2005] рассмотрена задача разработки пользовательского интерфейса с использованием разных типов диалога, базирующихся на использовании онтологий; процедура структурирования знаний в прикладной области изучается в [Palagin, 2005]. Анализ работ показывает, что их можно разделить на два типа: к одному относятся статьи, в которых изучаются аспекты создания и усовершенствования онтологий, к другому – решения прикладных задач с их использованием.

Поскольку, как отмечено выше, знания о предметной области существуют отдельно от методов анализа, неизбежно возникает проблема их композиции. Такое взаимодействие в условиях информационной избыточности должно быть достаточно эффективным по критерию минимизации времени. Однако решение этой задачи сталкивается с проблемой синергетического эффекта [Хакен, 1985], вследствие чего возникают структуры, объединяющие данные, вопросы, ответы и правила вывода, обладающие некоторой временной устойчивостью. Такие структуры во времени стремятся к положению с “минимальной энергией”. Но достижение указанного состояния происходит в условиях неполной информации. Таким образом, с одной стороны существует информационная избыточность исходных данных и недостаток информации об оптимизации процесса и результатов контроля знаний (в АСКЗ) – с другой. Разрешить такое противоречие предлагается с использованием самоорганизации процесса переходов по уровням ЭС (последовательности вопросов для контроля знаний в АСКЗ). Процедура самоорганизации функционирования ЭС позволит рационализировать и интеллектуализировать процесс обработки информации, необходимой для получения и контроля знаний.

Аспекты создания онтологий для экспертных систем

Онтологии допускают многократное использование в различных приложениях, они могут дополняться и модифицироваться. Для многих предметных областей, особенно в сфере товаров потребления, онтологии уже построены. Рассмотрим особенности их создания для контроля знаний. Предположим, что изучаемой дисциплиной является курс K по информационным технологиям. Исходя из проблемно-ориентированного изложения материала, дисциплина содержит лекции, практические и лабораторные занятия, что соответствует приобретению учащимися знаний Z, умений U и навыков N. Таким образом, изучение курса определяет реализацию отображения:

$$K \to < Z,U,N > . \qquad (4)$$

Известно, что онтология – это попытка всеобъемлющей и детальной формализации некоторой области знаний с помощью концептуальной схемы, которая состоит из иерархической структуры данных, содержащей все релевантные классы объектов, их связи и правила, принятые в этой области. Формально онтологию представляют как тройку $\{X,R,F\}$, где X – конечное множество концептов (элементов знаний), R – множество отношений между концептами, F – множество функций интерпретации концептов и / или отношений. Построение онтологии для ЭС, которой является АСКЗ, начинается с определения границ предметной области, базирующееся на представлении места и роли учебного курса в общей структурной схеме предметов и, как следствие, выяснении априорной информации, являющейся базисом изучения курса. Предметная область включает в себя основные понятия, а также перечень проблем, которые и очерчивают информационное поле курса (рис.1).

В соответствие с перечнем проблем необходимо определить концепты и построить иерархическую структуру. Так, в курсе по информационным технологиям концептами первого уровня могли бы быть: “Система”, ”Информация”, ”Интеллект”, ”Технология”, Модель”, ”Проектирование” и некоторые другие. Они определяются путем экспертных допущений как главные, базовые понятия. Концептами второго уровня будут определительные существительные или атрибуты концептов высшего уровня, например, “проектирование *интерфейса*”или “*математическая* модель”. Такое построение продолжается до

достижения базовых элементов – структурных единиц, обладающих самостоятельной ценностью для изучения курса.

Некоторую объективизацию в процесс построения онтологии может внести автоматизированный анализ текста электронного конспекта лекций. Для этого на первом этапе необходимо подсчитать количество наиболее часто встречающихся слов-концептов первого уровня. На втором этапе вычисляют количество наиболее часто встречающихся слов-атрибутов (прилагательных) или определительных понятий (существительных) концептов первого уровня. Третий этап (может инвертироваться со вторым) посвящается определению часто встречающихся отношений функционирования концептов первого уровня (иногда совместно с концептами второго уровня), выражающихся глаголами действия и т. д. Автоматизированная обработка позволяет определить концепты низших уровней, являющихся атрибутами, определяющими концептами для высших уровней. Результирующую онтологию получают, осуществляя композицию данных, полученных автоматизировано, и экспертных заключений.

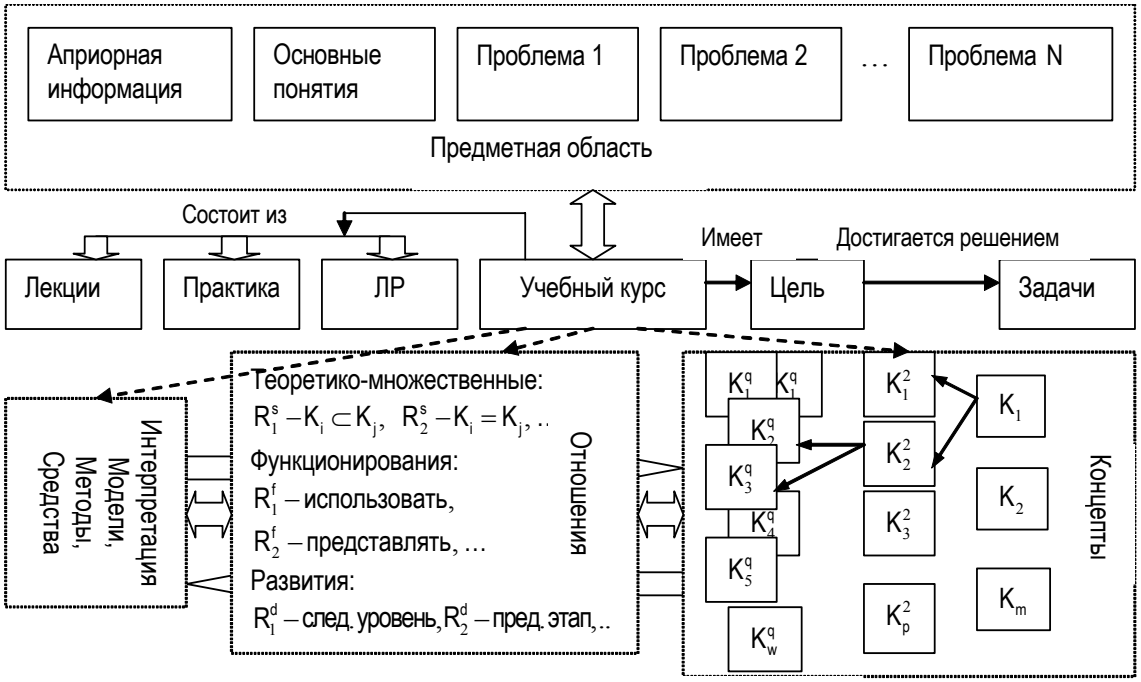


Рис. 1 – Структурная схема АСКЗ на основе онтологии

Технология суперпозиции онтологии и вопросов в экспертной системе

В основе ЭС лежит технология интерактивного общения, предполагающая наличие лица, принимающего решение, и пользователя (эксперта), отвечающего на вопросы системы. Рациональное проведение такого диалога, который на конечном этапе ведется между человеком и компьютером, возможно при выполнении таких условий: база данных вопросов и возможных ответов является достаточно полной для принятия решений; процедура экспертизы является верифицированной; алгоритм экспертизы должен быть оптимизированным по времени.

Традиционно, проведение процедура экспертизы полностью определялось лицом, принимающим решение, следствием чего было решение, базирующееся исключительно на субъективных предпочтениях. В системах АСКЗ преподаватель составлял список вопросов из разных тем курса, для контроля обучаемому случайным образом предлагалось их определенное количество. За ответ на вопрос он получал баллы и их суммарное количество составляло оценку, которая, в общем случае, не гарантировала знание предмета, поскольку случайный выбор вопросов не отражал полную картину изучаемых проблем и задач. Материал некоторых разделов оставался вне теста, а некоторые вопросы с незначительными модификациями повторялись.

Мы предлагаем при тестировании использовать принципы самоорганизации [Ивахненко, 1975], [Молчанов, 1988]: “множественности” моделей и свободы выбора. Их интерпретация по отношению к нашей задаче заключается в том, что существует множество упорядоченных последовательностей тестовых вопросов, приводящих к правильному результату, а также в том, что на любом этапе тестирования могут быть выбраны несколько вариантов его продолжения.

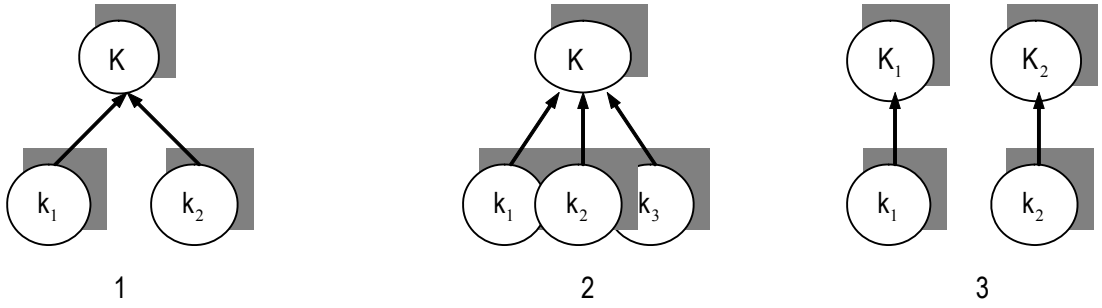


Рис.2 – Возможные составляющие конструкции онтологии

Составляя последовательность тестирования, будем руководствоваться структурой, определенной графом онтологии. При этом возможны варианты выбора соответствия, приведенные на рис. 2. Заметим, что граф представляющий онтологию, имеет нисходящую ориентацию. В то же время, процесс тестирования может иметь как нисходящий (дедуктивный – от общего к частному), так и восходящий (индуктивный – от частного к общему) характер. На рис. 2 показаны фрагменты, соответствующие восходящему подходу. Формирование последовательности вопросов в таком случае отвечает “И-ИЛИ” структуре. Концепты низшего уровня k_1 и k_2 (см. рис. 2.1.) составляют концепт K и между ними можно поставить логическое “ИЛИ”. Тогда в последовательности вопросов должны быть обязательно вопросы по сущности обоих концептов, а правильные ответы на них предполагают знание сущности K и избавляют от вопросов по его содержанию. Заметим, что концепты низшего уровня для k_1 и k_2 не имеют общих элементов, понятий, функций. Если это не так (см. рис. 2.2), то рационально задавать вопросы по сущности концепта, или его составляющей, являющимися общим для концептов одного уровня. Если концепты являются одноуровневыми (см. рис. 2.3) и не имеют пересечения на более низких уровнях, то необходимым условием прохождения теста являются правильные ответы на вопросы, инцидентные всем одноуровневым концептам.

Экспертная система может функционировать в “активном” и ”пассивном” режимах. В “пассивном”режиме последовательности вопросов системой определяются заранее и записываются в базу данных, в “активном” – последовательность вопросов формируется в процессе ответов обучаемого. В первом случае минимизируется время на генерацию вопроса, но отсутствует адекватная реакция на правильность ответов, во втором – если мощность онтологии достаточно большая, то время определения следующего вопроса может быть значительным. Преимущество “активного” режима заключается в том, что существует возможность гибкого реагирования и определения последовательности следующих вопросов в зависимости от предыдущих ответов.

Задачи и перспективы оптимизации процесса функционирования экспертной системы

Процесс извлечения знаний с использованием ЭС базируется на работе трех подсистем [Люгер, 2003]: редактора базы знаний, машины вывода и подсистемы объяснений. Оптимизация их функционирования требует решения таких задач:

- формального представления онтологии в элементном базисе базы знаний;
- обеспечения возможности определения соответствия между представлением онтологии и таблицей, содержащей тематические вопросы;

- разработки алгоритма проведения экспертизы (контроля знаний), предусматривающего возможность гибкой настройки в результате самоорганизации базы вопросов в режиме реального времени;
- разработки моделей и методов проведения экспертизы, начальным этапом которой является формализация вопросов в зависимости от типов ответов;
- учета возможности нечеткого представления субъективных суждений;
- разработка системы протоколирования и интерпретации результатов функционирования ЭС, предусматривающей объяснение логики проведения экспертизы.

В результате решения указанных задач открываются перспективы системного подхода к созданию ЭС в различных отраслях знания. Значительная степень унификации процесса их создания и проектирования оптимизирует процесс получения экспертных выводов. Необходимым условием этого является формирование онтологий соответствующих предметных областей, достаточное условие заключается в реализации технологии суперпозиции тестовых элементов и элементов онтологии.

Заключение

Предложенная технология создания экспертных систем, базирующаяся на методологии системного анализа, онтологии предметной области, а также принципах и методах самоорганизации является еще одним шагом в направлении создания эффективных экспертных систем. Эффективность заключается в минимизации времени проведения экспертиз и контроля знаний; в объективизации полученных решений, базирующихся на автоматизации процесса экспертного анализа, более полном охвате предметной области и уменьшении информационной избыточности тестовых вопросов и их последовательностей; непрямом формировании у экспертов и обучающихся представлений о структуре предметной области, ее базовых элементах и их функциональных взаимосвязях.

Стремительное развитие дистанционного обучения является еще одним аргументов в пользу создания и использования автоматизированных систем контроля знаний, базирующихся на использовании онтологий предметных областей, являющихся основой учебных курсов. Заметим, что разработка онтологии является достаточно сложным и трудоемким процессом, поэтому рационально этот процесс в границах учебного заведения, а в дальнейшем и в более широких масштабах унифицировать, для чего разработать программно-методическое обеспечение. Еще одним приложением для разработанных онтологий будет их использование в качестве базовых платформ для разработки дистанционных курсов, интегрирующих в себе подсистемы обучения, справочной информации, методических указаний, тестовых примеров, приемов отчетов и контрольных заданий.

Библиография

[Feigenbaum, 1963] E.A. Feigenbaum. The simulation of verbal learning behavior. In Feigenbaum and Feldman. – New York: McGraw-Hill, 1963.

[Плотинский, 1992] Ю.М. Плотинский. Математическое моделирование динамики социальных процессов. – Москва: Изд-во Московского ун-та, 1992. – 133 с.

[Згуровский, 1997] М.З. Згуровский, А.В. Доброногов, Померанцева Т.Н. Исследование социальных процессов на основе методологии системного анализа. – Киев: Наукова думка, 1997. – 222 с.

[Тимченко, 1991] А.А. Тимченко, А.А. Родионов. Основы информатики системного проектирования объектов новой техники. – Киев: Наукова думка, 1991. – 152 с.

[Гречко, 2005] А.В. Гречко. Онтология метода анализа иерархий Саати // Искусственный интеллект. – 2005. – № 3. – 746-757.

[Даревич, 2005] Р.Р. Даревич. Повышение эффективности интеллектуального анализа теста путем взвешивания понятий модели онтологии // Искусственный интеллект. – 2005. – № 3. – С. 571-577.

[Кучеренко, 2005] Е.И. Кучеренко, Д.А. Павлов. Некоторые аспекты анализа развития нечетких онтологий // Искусственный интеллект. – 2005. – № 3.– С. 162-169.

[Шалфеева, 2005] Е.А. Шалфеева. Классификация структурных свойств онтологии // Искусственный интеллект – 2005. – № 3. – С. 67-77.

[Гладун, 2005] А.Я. Гладун, Ю.В. Рогушина. Онтологии как перспективное направление интеллектализации поиска информации в мультиагентных системах Е-коммерции // In Proc. XI-th Int. Conf. “KDS-2005”. – Varna.– Vol. 1. – P. 158-165.

[Gribova, 2005] V. Gribova. Implementation of various dialog types using an ontology-based approach to user interface development // In Proc. XI-th Int. Conf. “KDS-2005”. – Varna.– Vol. 1. – P. 153-158.

[Palagin, 2005] A. Palagin, V. Peretyatko. Development of procedures of recognition of objects with usage multisensor ontology controlled instrumental complex // In Proc. XI-th Int. Conf. “KDS-2005”. – Varna.– Vol. 1. – P. 140-147.

[Хакен, 1985] Г. Хакен. Синергетика. – М.: Наука, 1985. – 320 с.

[Ивахненко, 1975] А.Г. Ивахненко. Долгосрочное прогнозирование и управление сложными системами. – Киев: Наукова думка, 1975. – 311 с.

[Молчанов, 1988] А.А. Молчанов. Моделирование и проектирование сложных систем. – Киев: Выща школа, 1988. – 359 с.

[Luger, 2002] G.F. Luger. Artificial intelligence. Structures and strategies for complex problem solving. – Addison Wesley: Boston, 2002. – 864 p.

Информация об авторе

Елена Нетавская – *Черкасский государственный технологический университет; бул. Шевченко 460/501, Черкассы, Украина; e-mail: neelena@list.ru*

АРХИТЕКТУРА ПРОГРАММНОГО КОМПЛЕКСА ОНТОЛИНЖ-KAON[1]

Владимир Горовой, Татьяна Гаврилова

Аннотация*: В работе описывается проект программного комплекса ОНТОЛИНЖ-KAON, предоставляющего технологическую поддержку всех стадий онтологического инжиниринга. Основной акцент сделан на оценке зрелости и качества онтологий, а также на использовании онтологий при помощи автоматизированной генерации порталов знаний, основанных на онтологиях. Возможность создания порталов знаний, построенных на онтологиях, может стать большим шагом вперед в области e-learning. В работе описываются преимущества, которые дают порталы знаний, построенные на базе онтологий.*

Ключевые слова*: онтологический инжиниринг, инженерия знаний.*

ACM Classification Keywords*: H.0 Information systems – General, I.2.6 Artificial intelligence - Learning*

Введение

Целью проекта ОНТОЛИНЖ-KAON является предоставление технологической поддержки полного цикла онтологического инжиниринга. В настоящее время существует огромное множество компонентов и программных продуктов, реализующих различные задачи в рамках работы с онтологиями. В связи с этим возникает актуальная задача интеграции этих компонентов в единую систему, поддерживающую стадии от создания онтологии, оценки ее зрелости и качества, до использования ее конечными пользователями.

К узким местам современных инструментов, обеспечивающих технологическую поддержку работы с онтологиями, относится оценка качества разрабатываемых онтологий и поддержка стадии использования онтологий. И это несмотря на то, что на сегодняшний день можно констатировать широкий интерес к онтологическому инжинирингу. Так имеется более 50 редакторов онтологий:

[1] Работа поддержана грантом **РФФИ 04-01-00466**

- Protégé,
- системы GALEN Case Environment (GCE),
- ICOM
- Integrated Ontology Development Environment
- IsaViz
- JOE
- KAON (including OIModeller)
- KBE -- Knowledge Base Editor (for Zeus AgentBuilding Toolkit)
- LegendBurster Ontology Editor
- LinKFactory Workbench
- Medius Visual Ontology Modeler
- NeoClassic
- OilEd
- OLR3 Schema Editor
- OntoBuilder
- и другие.

Однако все они не поддерживают разработку корпоративных порталов знаний. Первый шаг в этом направлении был сделан в рамках проекта KAON (http://kaon.semanticweb.org) [Motik et al., 2002]. Одна из разработанных подсистем (KAON Portal) позволяла генерировать портал, предоставляющий web-инерфейс для навигации по онтологии, созданной при помощи OI-Modeler'a. К недостаткам KAON Portal можно отнести то, что кроме самой онтологии, на нем нет никакой информации, и более того, нет возможности ее подключить. Кроме того, KAON Portal генерирует портал на основе онтологий, сохраненных во внутреннем формате (собственном расширении RDFS), не являющимся общепризнанным стандартом. Таким образом, онтологии, используемые для генерации портала, должны быть созданы только в KAON OI-Modeler, что, безусловно, является существенным ограничением.

Еще одним примером системы, ориентированной на создание портала является PORTO [Gavrilova et al., 2003]. Работа в системе PORTO состояла из следующих этапов:

1. Создание онтологии портала аналитиком при помощи визуального редактора
2. Создание дизайна портала и привязка концептов онтологии к представлению web-дизайнером
3. Online генерация страниц портала сервером PORTO в ответ на пользовательские запросы

К недостаткам PORTO можно отнести отсутствие интеграции с другими системами разработки онтологий. Таким образом, онтологию для генерации необходимо было разрабатывать в предлагаемом визуальном редакторе, сохраняющем данные в своем внутреннем формате.

Автоматизированная генерация порталов знаний на основе онтологий стала бы большим шагом вперед в области e-learning. Например, при разработке онтологии учебного курса или темы одной лекции автоматическое создание портала курса было бы неоценимо для студентов и освободило бы преподавателей от части работ по созданию портала.

То, что в основе портала лежит онтология, может сильно облегчить жизнь пользователям в процессе его использования. В этом случае можно дополнить стандартный текстовый поиск по порталу системой выполнения запросов по онтологии. В качестве интерфейса для продвинутых пользователей можно в этом случае использовать форму для ввода RQL-запросов [Karvounarakis et al., 2003]. Выполнение таких запросов избавит пользователей от фильтрации избыточной информации, которую он часто получает в результате стандартных текстовых запросов.

Компоненты системы и их взаимодействие

Компоненты системы ориентированы на решение следующих задач:

1. Формирование онтологий
2. Оценка онтологий
3. Использование онтологий

В качестве компонента формирования онтологий можно использовать любой редактор онтологий или другое средство для создания онтологий, позволяющее сохранять онтологии в формате OWL (http://www.w3.org/TR/2003/CR-owl-features-20030818). Из онтологических редакторов можно упомянуть Protege [Noy et al., 2001] или SWOOP [Kalyanpur et al., 2004]. Для создания OWL онтологий программным путем можно использовать Jena (HP labs - http://www.hpl.hp.com/semweb/downloads.htm), KAON2 (http://kaon2.semanticweb.org), IODT (IBM Integrated Ontology Development Toolkit - http://www.alphaworks.ibm.com/tech/semanticstk) или OWL-API (http://owl.man.ac.uk/api.shtml). Эти API значительно облегчают жизнь разработчикам программных средств, реализующих импорт своих внутренних онтологий в формат OWL. Таким образом, в результате работы первого компонента на выходе получается онтология в формате OWL.

Важным звеном системы ОНТОЛИНЖ-KAON является модуль оценки созданной онтологии и предоставления рекомендаций по ее улучшению. В качества такого компонента можно использовать существующие средства для оценки онтологий (OntoAnalyser, KAON2). К сожалению, эти средства оценки не достигли серьезного уровня зрелости и не очень широко используются. Это вызвано тем, что рекомендации инженера по знаниям в области формирования онтологий с трудом поддаются формализации и реализации их программным путем. В связи с этим актуальна задача построения новых средств оценки онтологий, отвечающих потребностям большинства пользователей. Некоторые идеи, касающиеся создания такого модуля, представлены ниже в разделе, посвященном компоненту оценки.

К компоненту оценки в системе ОНТОЛИНЖ-KAON предъявляются следующие требования:

1. Возможность работы с онтологиями в формате OWL

2. Вывод замечаний связанных с качеством онтологии и предоставление рекомендаций по улучшению качества.

Компонентом, реализующим задачу использования онтологий, являются корпоративные и образовательные порталы знаний, сгенерированные на основе онтологий, сформированных и оцененных с помощью остальных компонентов системы. Нами сформулированы следующие требования к порталу:

- генерация на основе онтологии в формате OWL
- доступ к созданной онтологии
- добавление релевантной информации к экземплярам (instance'ам) концептов для отображения на странице, сгенерированной порталом
- поиск по порталу (обычный)
- поиск по порталу, использующий то, что в основе портала лежит онтология.

Архитектура системы, решающей все поставленные задачи, представлена на рис. 1. Ее основные компоненты:

- OntologyProducer – компонент формирования онтологий
- OntologyEvaluator – компонент оценки качества и зрелости онтологий
- KAON Portal Extension - расширение KAON Portal'a. В задачу этого компонента входит генерация портала основанного на онтологиях
- Concept2Visualization - модуль, предоставляющий возможность связывания представления с экземплярами онтологии (например: концепту проект "OntoWeb" ставится в соответствие некоторая его визуализация на генерируемой странице, которая содержит необходимую информацию по проекту)
- Portal2ReasonerInterface - модуль, предоставляющий интерфейс для обращения с запросами поиска в рамках онтологии портала
- Reasoner - модуль, для осуществления рассуждений (например, Pellet OWL Reasoner).
- Servlet/JSP container - сервер, поддерживающий Servlet'ы и JSP. На таком сервере может работать KAON Portal Extension модуль (например, Apache Tomcat или JBoss).

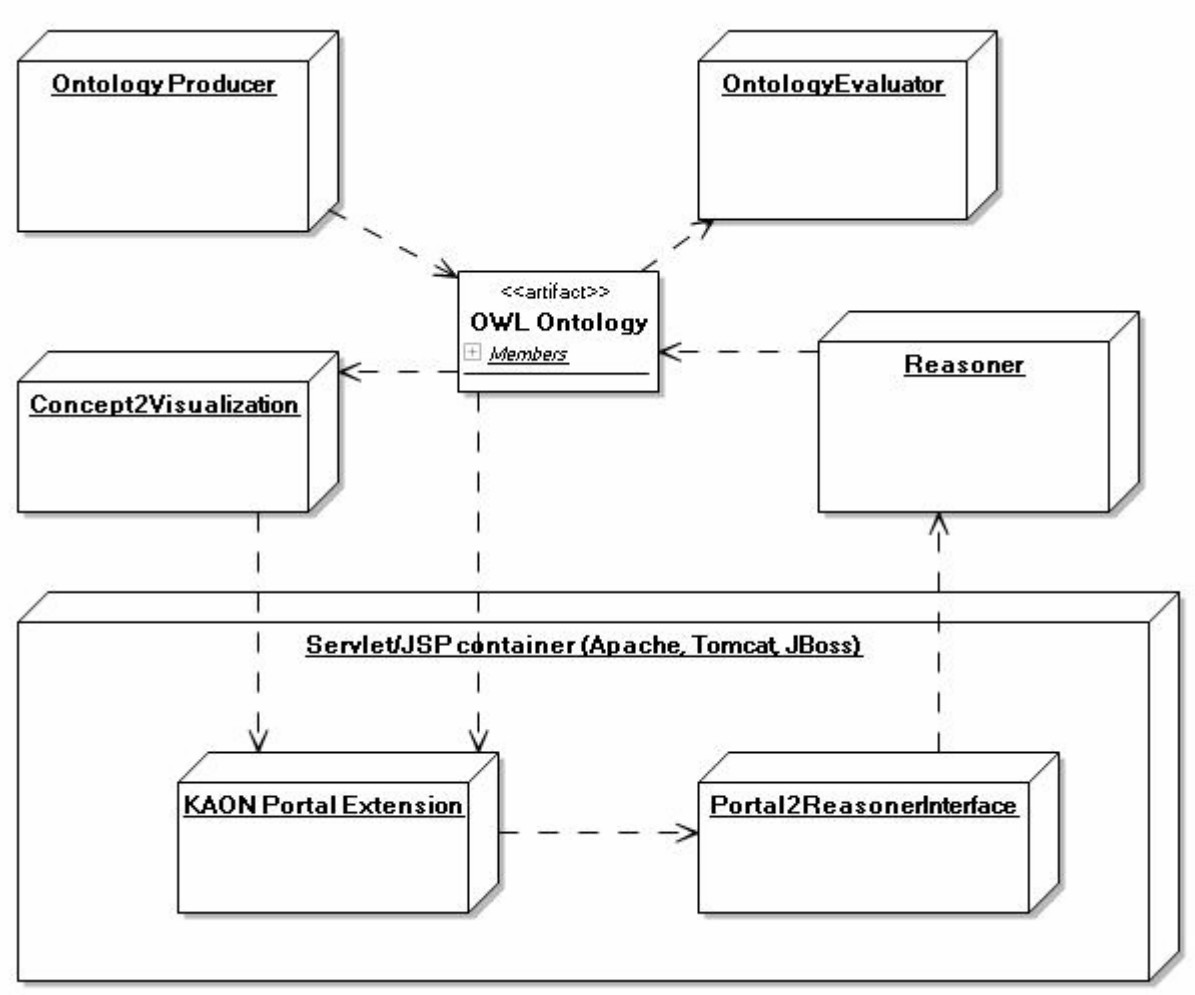


Рис. 1. Архитектура ОНТОЛИНЖ-KAON

Таким образом, в рамках разработанной архитектуры нами разрабатываются нижеследующие компоненты и проводится их интеграция друг с другом и другими компонетами системы ОНТОЛИНЖ-KAON:

- KAON Portal Extension
- Concept2Visualization
- Portal2ReasonerInterface
- OntologyEvaluator

Компонент оценки онтологий – OntologyEvaluator

В качества компонента оценки качества онтологий можно использовать существующие средства для оценки онтологий (OntoAnalyser, KAON2). К сожалению, существующие средства оценки не достигли серьезного уровня зрелости и не очень широко используются. Предлагаемое решение призвано устранить недостатки, присущие существующим средствам оценки.

В результате анализа многочисленных студенческих работ по созданию онтологий в простых и общеизвестных предметных областях были обнаружены основные факторы, которые отличали хорошие онтологии от плохих. Эти законы можно переформулировать и сделать применимыми для практического инженера по знаниям. Основная гипотеза может быть сформулирована как: «Гармония = концептуальный баланс + ясность».

При этом концептуальный баланс подразумевает, что

- Понятия одного уровня иерархии связываются с родительским концептом одним и тем же типом отношения (например, «класс-подкласс» или «часть-целое»).
- Глубина ветвей онтологического дерева должна быть примерно одинаковая (±2).
- Общая картинка должна быть довольно симметричной.
- Перекрестные ссылки должны быть по возможности исключены.

Ясность включает

- Минимизацию. Так максимальное число концептов одного уровня или глубина ветви не должна превышать знаменитое число Ингве-Миллера (7±2) [Miller, 1956].
- Прозрачность для чтения. Тип отношений должен быть по возможности очевиден, так чтобы не перегружать схему онтологии лишней информацией и опускать названия отношений.

Все эти законы согласуются с некоторыми результатами гештальт-психологии, сформулированные еще Максом Вертгеймером [Wertheimer, 1944]. Так основной принцип хорошего гештальта (хорошей формы) или закон прегнантности был сформулирован так:

«Организация любой структуры в природе или в сознании должна быть настолько хороша (регулярна, полна, сбалансирована или симметрична), насколько позволяют существующие условия».

Большую часть перечисленных факторов можно формализовать и реализовать их проверку в OntologyEvaluator'е. Таким образом, использование этого компонента может способствовать созданию гармонических онтологий.

Использование онтологий

На этой стадии предполагается генерация порталов знаний на основе построенных онтологий и последующее использование портала в качестве средства для навигации по онтологии, для поиска необходимой информации, для задания запросов к онтологии.

Следующие компоненты относятся к стадии использования онтологий:

- KAON Portal Extension
- Concept2Visualization
- Portal2ReasonerInterface
- OntologyEvaluator
- Reasoner
- Servlet/JSP container.

В качестве основы для KAON Portal Extension можно использовать разработанный в рамках KAON модуль KAON Portal. KAON Portal позволяет генерировать портал для навигации по онтологии, описываемой на частном расширении RDFS (получить онтологию в этом формате можно с помощью OI-Modeler). На рис. 2 представлен скриншот странички портала, посвященной концепту проект.

Для реального использования сгенерированного портала в качестве образовательного портала знаний необходим реализовать в KAON Portal Extension следующие возможности:

- Поддержка OWL, т.к. работа с общепринятым стандартом в области описания онтологий будет способствовать лучшей интероперабельности и интеграции с другими средствами онтологического инжиниринга.
- Возможность связывания представления с концептами онтологии. Без этой функциональности невозможно создать портал знаний, который можно использовать. Эту возможность реализует Concept2Visualization модуль.
- Интерфейс для обращения с запросами поиска в рамках онтологии портала. В самом простом варианте это может быть возможность задания RQL-запросов и предоставления результатов по ним. Эту функциональность реализует Portal2ReasonerInterface.

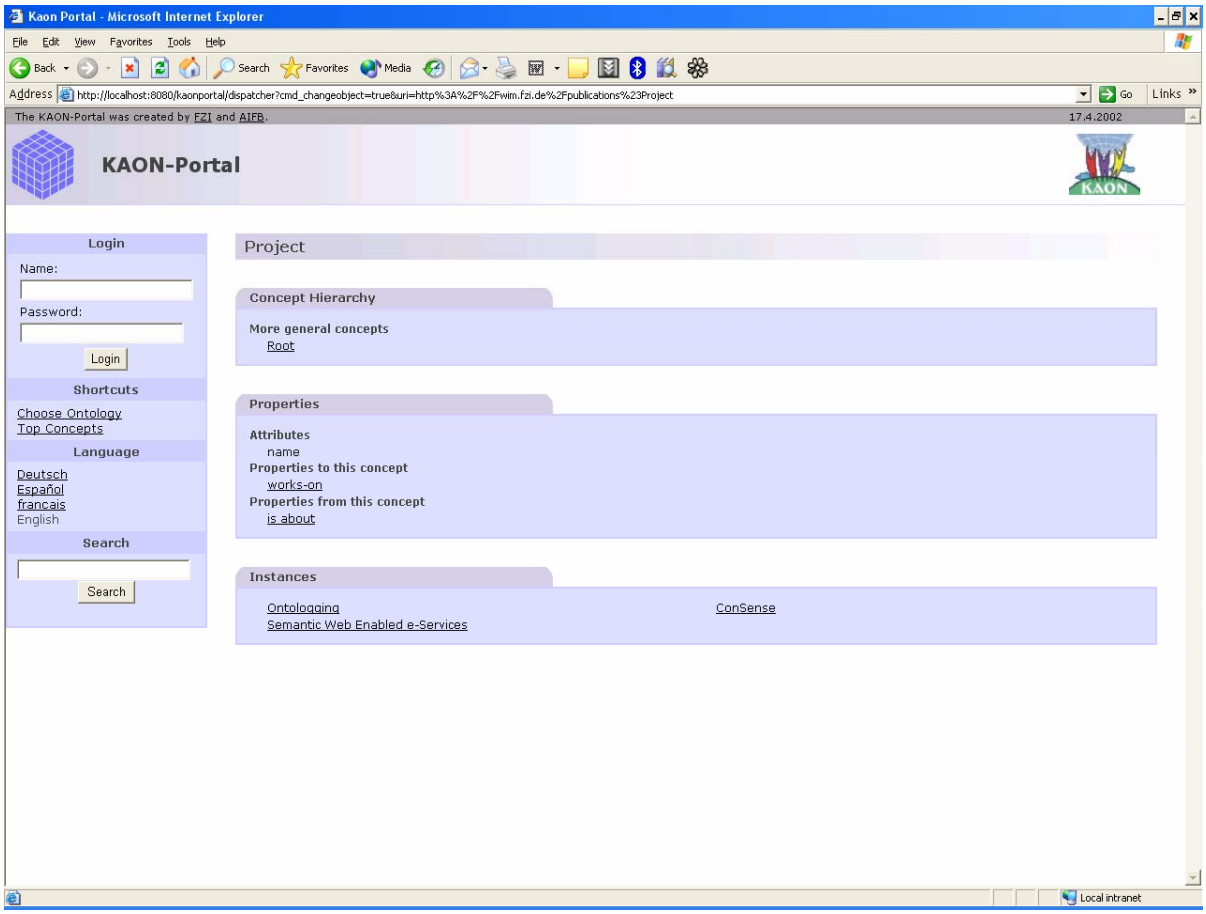


Рис. 2: Концепт проект

Заключение

При наличии всей вышеописанной функциональности ОНТОЛИНЖ-KAON может стать большим шагом вперед на пути к использованию технологий и методологий онтологического инжиниринга для создания образовательных порталов знаний. Интересным представляется подход, предлагаемый для оценки зрелости и качества онтологий. Он может содействовать созданию высококачественных гармонических онтологий. В сравнении с существующими системами онтологического инжиниринга новым является возможность динамической генерации портала, построенного на основе созданной онтологии, и содержащего как возможности для навигации по онтологии, так и информативную составляющую, относящуюся к экземплярам онтологии. Предлагаемое решение является гибким и позволяет автоматизировать отражение изменений, сделанных в онтологии, на страницах, генерируемых порталом. Новым также является возможность поиска по порталу, использующая то, что в основе портала лежит онтология. Эта функциональность коренным образом отличается от обычного поиска с помощью поисковых систем, т.к. предоставляет только семантически корректные результаты и избавляет пользователя от необходимости выбирать из множества вариантов, многие из которых имеют весьма отдаленное отношение к ожидаемым результатам.

Библиография

[Gavrilova et al., 2003] T.A. Gavrilova, V. A. Gorovoy. Ontological Engineering for Corporate Knowledge Portal Design // In "Processes and Foundations for Virtual Organizations", Eds. L. Camarinha -Matos and H. Afsarmanesh, Kluwer Academic Publishers, 2003. - p.289-296.

[Kalyanpur et al., 2004] A. Kalyanpur, E. Sirin, B. Parsia, J. Hendler. Hypermedia inspired ontology engineering environment: Swoop. // In Proceedings of 3rd International Semantic Web Conference (ISWC-2004), Japan (Poster).

[Karvounarakis et al., 2003] G. Karvounarakis, A. Magkanaraki, S. Alexaki, V. Christophides, D. Plexousakis, M. Scholl, K. Tolle. Querying the Semantic Web with RQL. // In Computer Networks and ISDN Systems Journal, Vol. 42(5), August 2003, pp. 617-640.

[Miller, 1956] G.A. Miller. The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information. // In The Psychological Review, 1956, vol. 63, pp. 81-97

[Motik et al., 2002] Boris Motik, Alexander Maedche, Raphael Volz. A Conceptual Modeling Approach for Semantics-Driven Enterprise Applications. // In Proceedings of the First International Conference on Ontologies, Databases and Application of Semantics (ODBASE-2002). Springer, 2002.

[Noy et al., 2001] N.F. Noy, M. Sintek, S. Decker, M. Crubezy, R.W. Fergerson, M.A. Musen. Creating Semantic Web Contents with Protege-2000. // IEEE Intelligent Systems 16(2), pp. 60-71, 2001

[Wertheimer, 1944] M. Wertheimer. Gestalt theory. // In Social Research, 11, 78-99.

Authors' Information

Vladimir Gorovoy *– PHD student, Saint-Petersburg State Polytechnical University, Intelligent Computer Technologies Dpt. 195251, Politechnicheskaya 29/9, St. Petersburg, Russia; e-mail: vgorovoy@mail.ru*

Tatiana Gavrilova *– Professor, Saint-Petersburg State Polytechnical University, Intelligent Computer Technologies Dpt. 195251, Politechnicheskaya 29/9, St. Petersburg, Russia; e-mail: gavr_csa@rambler.ru*

СИСТЕМА ИНТЕЛЛЕКТУАЛЬНОГО ПОИСКА И АВТОМАТИЧЕСКОЙ КАТАЛОГИЗАЦИИ ДОКУМЕНТОВ НА ОСНОВЕ ОНТОЛОГИЙ

Вячеслав Ланин, Людмила Лядова, Светлана Чуприна

Аннотация*: Статья посвящена описанию подхода к реализации системы интеллектуального поиска и автоматической классификации и каталогизации документов в CASE-системе, управляемой метаданными. Представленная система использует преимущества подхода, основанного на онтологиях, в совокупности с традиционным подходом, основанным на концепции ключевых слов. Разработанный метод характеризуется универсальностью применения, возможностью интеграции с существующими средствами поиска документов, а также мощными интеллектуальными возможностями.*

Keywords*: электронный документ, автоматическая каталогизация и классификация, онтологический подход, проектирование и разработка информационных систем.*

ACM Classification Keywords*: I.2 Artificial Intelligence: I.2.7 Natural Language Processing – Text analysis; D.2 Software Engineering: D.2.2 Design Tools and Techniques – Computer-aided software engineering (CASE).*

Введение

При разработке крупных распределенных информационных систем, отдельные подсистемы которых должны устанавливаться в территориально удаленных учреждениях, имеющих различные технические возможности, отличающихся многообразием организационных форм и форм деятельности, инструментальные средства, используемые для их создания, должны удовлетворять требованиям, обеспечивающим возможность их настройки на различные условия эксплуатации и потребности пользователей как при установке, так и динамически, в ходе эксплуатации. Реализация этих требований обеспечивает эффективность затрат на создание системы, высокую степень ее адаптируемости и масштабируемости, живучести.

В CASE-системе METAS (METAdata System), основанной на интерпретации многоуровневых метаданных, описывающих информационную систему, созданную с помощью этой технологии, с различных точек зрения и с различной степенью детализации, уникальные возможности динамической настройки системы

обеспечиваются средствами реструктуризации базы данных системы, генерации и настройки пользовательского интерфейса, генерации запросов и формирования отчетов (документов) [Лядова, 2003].

Этап анализа предметной области при использовании для разработки информационной системы CASE-средств становится наиболее трудоемким и ответственным этапом. Любые изменения условий деятельности организации, для которой создается информационная система, требуют выполнения повторного анализа и внесения изменений в модель информационной системы. Изменения условий эксплуатации системы, потребностей пользователей чаще всего связаны с какими-либо нормативными документами, правовыми актами, регламентирующими деятельность в данной предметной области или на уровне конкретного учреждения. Анализ предметной области, таким образом, во многом опирается на анализ этих документов, образующих сложную систему. Внесение изменений в модель должно основываться именно на изменениях, закрепляемых в нормативных документах.

Снижение трудоемкости работы аналитика может быть обеспечено максимальной автоматизацией процесса анализа документов. Для решения этой задачи необходимо иметь средства поиска и хранения всего множества документов, получаемых из различных источников, изданных в исследуемой области деятельности на различных уровнях, а также средства их классификации, каталогизации и анализа.

В данной работе рассматриваются проблемы, возникающие при работе с информацией в неоднородной программной и организационной среде, связанные с поиском документов, их электронной каталогизацией. Примерами таких документов могут служить различные внутренние документы организации (приказы, договоры, акты и пр.), нормативно-правовые акты и т.п. Все эти документы поступают в информационную систему децентрализовано, из разнородных источников, являются обычно слабоструктурированными, что осложняет работу с ними. Крайне важными для реализации задачами в этой области являются автоматизация процессов обмена данными с различными информационно-правовыми системами, обеспечение возможности импорта текстов и документов из файлов и баз данных разнообразных форматов и систем управления документами.

К основным проблемам, препятствующим быстрой и качественной работе с документами в электронных системах управления документами, можно отнести недостаточную структурированность информации, ее избыточность, наличие большего объема малополезной с точки зрения конкретной задачи, решаемой пользователем, информации. На результативность процесса поиска необходимых документов оказывает большое влияние и человеческий фактор: зачастую пользователь не готов к долгому ожиданию результатов поиска и просмотру и анализу большого объема результирующей выборки. Кроме того, большинство пользователей неэффективно используют поисковое программное обеспечение и, как правило, они игнорируют расширенные поисковые возможности и ограничиваются короткими типовыми запросами.

Эффективным решением части перечисленных выше проблем может служить создание специализированного инструментария в информационных средах и системах электронного документооборота, основанного на методах искусственного интеллекта, позволяющего избавиться от ряда нежелательных перечисленных выше свойств.

Задача поиска документов

Рассмотрим ситуацию поиска человеком какой-либо информации в книге, «бумажном» документе. Самый очевидный вариант – прочитать всю книгу (документ), но этот процесс может потребовать значительных временных затрат. Однако, если пользователь уже имеет некоторые знания в соответствующей предметной области, то он может воспользоваться оглавлением книги (документа), чтобы отобрать разделы, посвященные интересующим его вопросам, или воспользоваться предметным указателем, чтобы определить номера страниц, на которых упоминаются искомые термины.

В данном случае оглавления и указатели являются инструментами, упрощающими поиск. При работе с информационными системами, включающими средства управления документами (не только их формирования и хранения в самой системе, но и поиска во внешних источниках, импорта, анализа, классификации и каталогизации) роль «искомой информации» играют документы, а в качестве «оглавлений» и «предметных указателей» выступают службы, именуемые тематическими каталогами.

Допустим, что пользователь работает с электронной системой и ему необходимо собрать сведения о городе «Пермь». Можно выделить несколько этапов поиска информации. У пользователя появляется необходимость найти сведения по какому-либо вопросу, т.е. *возникает информационная потребность*. Затем пользователю необходимо некоторым образом *формализовать* свою информационную потребность. Процесс формализации в традиционных поисковых системах сводится к выявлению набора понятий и терминов (ключевых слов), характеризующих информационную потребность, и определению отношений между ними. Выделенное множество ключевых слов с зафиксированными отношениями между ними называется *запросом*. На следующем этапе пользователь через интерфейс поисковой системы *вводит запрос*. Система на множестве документов, являющемся информационно-поисковым пространством, осуществляет *выборку документов*, которые по внесенным в систему критериям соответствуют запросу пользователя, и *формирует результат* (*отклик*). Найденные документы по своему содержанию (рис. 1) делятся на две группы: документы, соответствующие информационной потребности пользователя, и документы, не соответствующие его информационной потребности, но соответствующие запросу пользователя с точки зрения информационно-поисковой системы (информационный шум). В рассмотренном выше примере к шуму могут относиться документы, в которых «Пермь» не будет являться названием города.

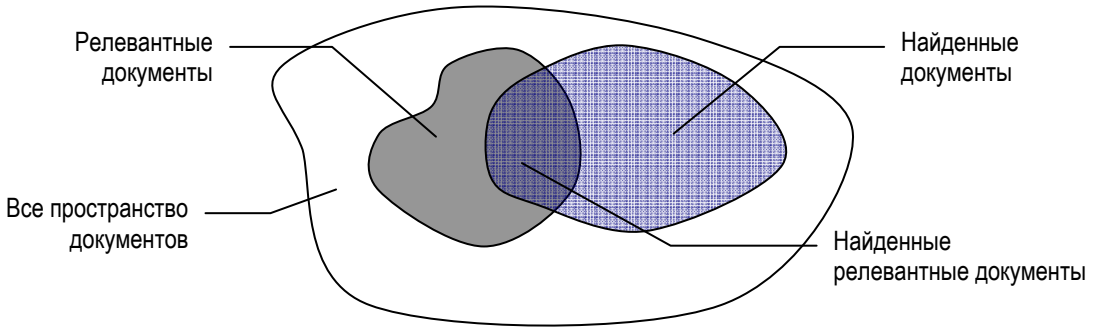


Рис. 1. Пространство поиска документов

Меру соответствия полученного отклика и информационной потребности пользователя называют *смысловой релевантностью*, а меру соответствия отклика запросу – *формальной релевантностью*. Как правило, признаком, по которому информационно-поисковая система определяет формальную релевантность документа, является присутствие ключевых слов запроса в тексте данного документа. При поиске, основанном на ключевых словах, за пределами множества найденных документов обычно остается часть документов, которые соответствуют информационной потребности пользователя. В примере с ключевым словом «Пермь» не найденными по запросу могут остаться документы, в которых вместо фраз «город Пермь», «Пермь» употребляются выражения «столица Пермского края», «крупнейший город западного Урала» и т.п.

Основная проблема поиска информации заключается в том, что большинство поисковых систем основываются на поиске ключевых слов, а для таких систем «слово» *не имеет четкого смысла*, или, другими словами, семантического содержания.

Большинство технологий работы с документами ориентированы на организацию удобной работы с информацией для человека. Но зачастую методы работы с электронной информацией просто копировали методы работы с «бумажной» информацией. В текстовом редакторе присутствуют широкие возможности форматирования текста (представления в удобном для человека виде), но практически отсутствуют возможности для передачи смыслового содержания текста. Компьютеру в большинстве случаев отводится роль «пишущей машинки» или вычислительного устройства, нацеленного на механический перебор вариантов ответов. Для эффективного решения задачи поиска необходимо расширить понятие традиционного документа: *с документом необходимо связать знания, позволяющие интерпретировать и обрабатывать хранящиеся в этом документе данные*.

Методы искусственного интеллекта, как правило, используются для решения трудно формализуемых задач, постановка которых проста и понятна для человека, но при разработке алгоритмов их решения

возникают трудности. Одна из таких задач – работа с документами в информационных системах: их поиск и каталогизация, анализ и извлечение информации.

В настоящее время существуют различные подходы, модели и языки, ориентированные на интегрированное описание данных и знаний. Наиболее перспективным и универсальным на данный момент представляется онтологический подход.

Понятие онтологии

В настоящее время понятие «онтология» является одним из наиболее часто используемых понятий. Термин «онтология» применяется в различных контекстах, в которых ему приписывается различный смысл. Учитывая специфику решаемых в данной работе задач, будем считать, что *онтология – это точная спецификация некоторой области, которая включает в себя словарь терминов (понятий) предметной области и множество связей между ними* (типа «элемент-класс», «часть-целое»), которые описывают, как эти термины соотносятся между собой в конкретной предметной области. Фактически в данном случае онтология – это *иерархическая понятийная основа рассматриваемой предметной области*, для которой разработана информационная система.

Поиск подходящих онтологий сложен, занимает много времени. Данные обстоятельства приводят к тому, что под конкретные задачи подчас невозможно найти подходящую онтологию из числа разработанных ранее, поэтому создание новой отологии, учитывающей специфику конкретной задачи, является оправданным. Кроме того, использование готовых онтологий обладает еще рядом недостатков. В частности, знания разных людей могут укладываться в разные онтологии, при этом нельзя утверждать, что одна из них лучше другой. Во многих случаях для одной и той же организации, предприятия, в котором установлена информационная система, или для какой-либо трудно формализуемой предметной области можно построить несколько различных онтологий, отражающих различные точки зрения на предметную область и решаемые в ней задачи.

Для описания онтологий и работы с ними существуют различные языки и системы, однако, наиболее перспективным представляется визуальный подход, позволяющий специалистам непосредственно «рисовать» онтологии, что помогает наглядно сформулировать и объяснить природу и структуру явлений. Визуальные (графовые) модели обладают особенной познавательной силой.

Онтологический поиск документов

В соответствии с предлагаемым подходом [Chuprina, 2004] поиск информации осуществляется с помощью онтологии предметной области информационной системы или с помощью специально разработанной пользователем онтологии. В общем случае интерпретация данных, информации, содержащейся в документах, является чрезвычайно сложной задачей, но для решения задачи поиска документов необходим лишь механизм сопоставления документа и онтологии.

Процесс поиска документа на основе онтологий можно описать следующим образом.

Процесс начинаем с *поиска в документе основных понятий онтологии*. Если все понятия найдены в документе, то считаем, что онтология описывает данный документ.

Допустим, системе не удается найти какое-либо понятие онтологии, тогда начинается *просмотр и поиск синонимов данного понятия*. Успешный поиск свидетельствует о том, что онтология описывает данный документ.

Если не найдены и синонимы или они просто не представлены в онтологии, то система пытается *«собрать» понятие «по частям», учитывая связи «часть-целое»*. Если учет данных типов связей также не дал результатов, то система может *перейти по связям «класс-подкласс» для искомого понятия к рассмотрению более конкретного или более обобщенного понятия*.

Таким образом, мы получаем *рекурсивный механизм сопоставления онтологии и документа*. Система пытается «привязать» документ к онтологии или ее фрагменту. В отличие от традиционной схемы сопоставления по ключевым словам изложенный механизм обладает значительно большей семантической мощностью, позволяя *найти в документе понятия, не представленные в явном виде*.

Основными преимуществами использования онтологического подхода в решении задач поиска являются:

– *системность* (онтология представляет целостный взгляд на предметную область);
– *единообразие* (знания, представлены в единой форме);
– *целостность* (построение онтологии позволяет восстановить недостающие логические связи предметной области во всей их полноте).

Найденные во внешних источниках документы могут быть импортированы в информационную систему для последующей их классификации и каталогизации, анализа и извлечения необходимой как пользователям системы, так и ее разработчикам, информации.

Онтологическая классификация и каталогизация документов

Для организации процесса каталогизации документов пользователю необходимо с каждой категорией документов сопоставить онтологию (рис. 2).

При поступлении в систему нового документа он последовательно сопоставляется с онтологией каждой категории. При успешном процессе сопоставления документ попадает в данную категорию. Один и тот же документ может соответствовать нескольким онтологиям, и, следовательно, может быть отнесен к нескольким категориям.

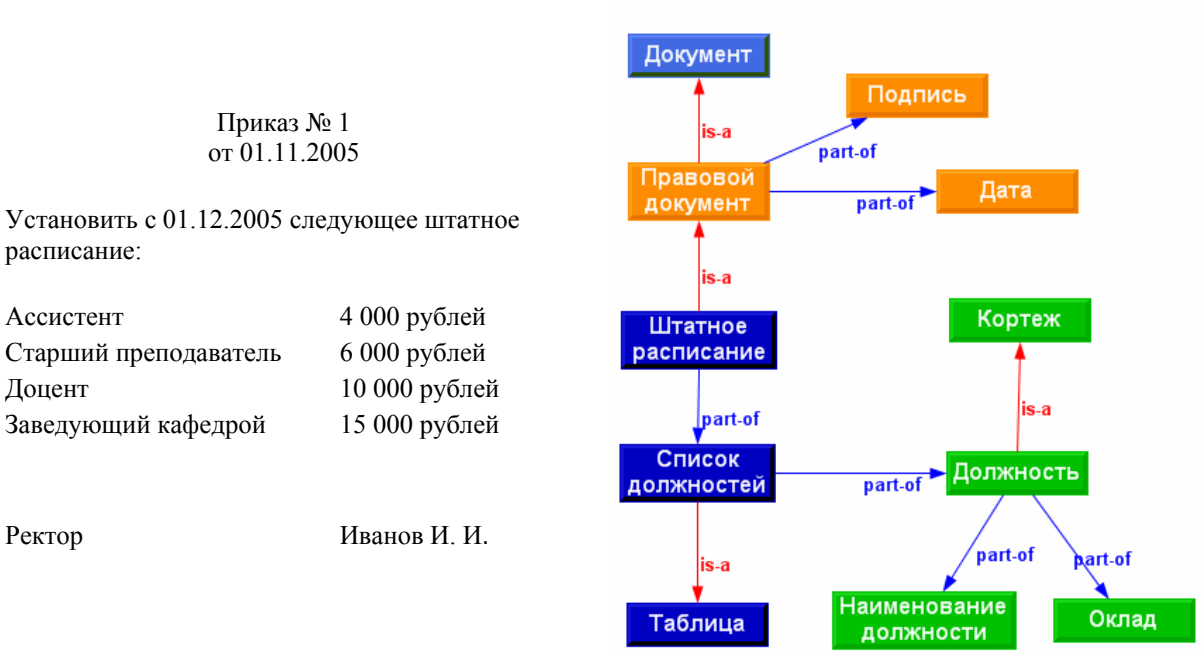


Рис. 2. Документ и его онтология

Систему категорий удобно представить в виде дерева. Следовательно, и соответствующие онтологии образуют иерархию. При таком подходе к описанию онтологии дочерние вершины будут уточнять онтологии родительских вершин. Например, с вершинами верхнего уровня можно связать небольшие онтологии, описывающие в системе управления документами распорядительные документы или договоры и т.п., а вершины следующих уровней будут соответствовать онтологиям, которые будут конкретизировать эти типы документов.

Неотъемлемой частью подхода является возможность работы с различными форматами документов.

Для организации механизма сопоставления онтологий и слабоструктурированных документов необходима интерпретация таких понятий, как «таблица», «кортеж», «дата», «число» и т.д., позволяющих учитывать формат конкретного документа, его структуру. Поэтому в систему необходимо включить компоненты, обеспечивающие унифицированный доступ к документам в различных форматах. Такую функциональность можно реализовать за счет «устанавливаемых драйверов форматов» – компонентов, реализующих заранее определенный интерфейс, обеспечивающих доступ к документам определенного формата. Реализация таких «драйверов» может быть основана на использовании шаблонов, образцов, позволяющих распознать структуру документа.

Заключение

Разработанный подход характеризуется универсальностью применения, возможностью интеграции с существующими средствами поиска документов, а также мощными интеллектуальными возможностями. Использование предложенного механизма поиска позволяет получить результирующие выборки с высоким показателем смысловой релевантности.

Механизм онтологической каталогизации в совокупности с возможностями конкретной информационной системы позволяет обеспечить эффективное управление документами, генерируемыми в системе и получаемыми из внешних неоднородных источников.

CASE-технология METAS, разработанная сотрудниками АНО «Институт компьютинга», обеспечивает возможность эффективного использования предложенных в данной статье средств как при создании информационной системы, так и в ходе ее эксплуатации пользователями, а также при выполнении анализа документов для осуществления динамической настройки системы.

Технология предоставляет в распоряжение пользователя не только средства настройки, но и удобные инструменты для навигации по информационным объектам, представляющим сущности предметной области, в соответствии с существующими между ними связями. Проводник объектов работает с деревом объектов, которое может настраиваться пользователями в соответствии с их информационными потребностями. Дерево объектов дает возможность не только просмотра объектов, но и выполнения над ними всех определенных для них в информационной системе операций. Каждый документ, сформированный в системе, представляется в ее базе данных соответствующей сущностью. Таким образом, он «попадает» в вершину дерева объектов, представляющую эту сущность, и становится доступным для просмотра пользователями. Один и тот же документ может быть получен при обходе дерева по различным ветвям в зависимости от задач, решаемых пользователем.

Включение в дерево объектов вершин, предназначенных для каталогизации документов, с каждой из которых пользователь может связать свою онтологию, позволяет использовать тот же механизм навигации и для работы с документами, импортированными из внешних источников и каталогизированными в системе.

Расширение CASE-системы средствами анализа слабоструктурированных документов, классифицированных и каталогизированных на основе созданных разработчиками информационной системы и ее пользователями онтологий, может существенно снизить трудоемкость сопровождения системы, ее настройки на меняющиеся в ходе эксплуатации условия и информационные потребности пользователей.

Библиографический список

[Лядова, 2003] Л.Н. Лядова, С.А. Рыжков. CASE-технология METAS. В кн.: Математика программных систем. Пермский государственный университет, Пермь, 2003. С. 4-18.

[Chuprina, 2004] S. Chuprina, V. Lanin, D. Borisova, S. Khaeva. Internet Intelligent Search System SmartFinder. In: Proc. of the European Workshop on the Integration of Knowledge, Semantics and Digital Media Technology. Knowledge-Based Media Analysis for Self-Adaptive and Agile Multimedia Technology / The Royal Statistical Society, November 25-26, 2004, London, U.K. P. 151-156.

Сведения об авторах

Вячеслав Ланин *– Пермский государственный университет, студент магистратуры кафедры математического обеспечения вычислительных систем; Россия, г. Пермь, 614990, ул. Букирева, д. 15; e-mail: lanin@perm.ru*

Людмила Лядова *– АНО «Институт компьютинга», заместитель директора; Россия, г. Пермь, 614097, ул. Подлесная, д. 19/2, к. 38; e-mail: lnlyadova@mail.ru*

Светлана Чуприна *– Пермский государственный университет, доцент кафедры математического обеспечения вычислительных систем; Россия, г. Пермь, 614990, ул. Букирева, д. 15; e-mail: chuprina@psu.ru*

AN APPROACH TO AUTOMATED DETECTION OF USABILITY DEFECTS IN USER INTERFACES

Valeriya Gribova

Abstract. *The article presents a new approach to automated detection of usability defects in user interfaces. The principal features of the approach are: creation of an expandable system for detection of usability defects, detection defects within the design phase, and information to the developer not only about existence of defects but also advice on their elimination.*

Keywords*: Ontology, defects, interface model, user interface development*

ACM Classification Keywords*: I.2.2 Artificial intelligence: automatic programming*

Introduction

Quality and speed of software development are traditionally considered as a compromise where one of them is paid more attention than the other. However, to remain a competitive company developing software should not only increase speed but also improve quality of its software. To achieve this aim, a lot of efforts of developers are required. According to the Cnews channel in 2001 defects in software cost the world business 175 billion US dollars.

A user interface is an integral part of most software so quality of its development is of critical importance. In addition to general criteria of software quality the user interface has an additional one, namely, usability. The user estimates the whole application program based on its user interface.

Estimating usability is an expensive task in terms of time and labor. This problem is usually solved by increasing the number of testers or by automation of the process.

In this article an additional component of automated detection of usability defects to a tool for user interface development is proposed. The main task of this component is to detect defects of usability in a user interface and to give advice on their elimination. The paper demonstrates urgency of the problem, the basic idea of the approach, and an ontology of defects.

Urgency of the Problem

Usability is the measure of the quality of a user's experience when interacting with an application program. It is also a combination of factors that affect the user's experience with the application program, including easiness of learning, efficiency of using, memorability, error frequency and severity, and subjective satisfaction [http://www.usability.gov].

Every year the number of interface elements and their properties is increasing. There are criteria for design of each interface element, their groups and individual characteristics depending on the user's profile (age, experience, specific requirements, etc.), the structure of a domain, a field of using an application program, a type of an application program, and so on. However, all criteria of usability are described in articles, textbooks and manuals informally, as sets of recommendations. The developer must know all these criteria. This fact requires high qualification of developers, their expertise in usability principles, and more evaluators. As a result, cost and time of development increase. To make an application program reliable and to improve its quality, it is suggested to provide the process of user interface development with a system of automated detection of usability defects.

Automation of this process has several potential advantages over non-automated methods, such as [1]:

-Reducing the cost of usability evaluation;
-Increasing consistency of the errors uncovered;
-Predicting time and error costs across an entire design;
-Reducing the need for evaluation expertise among individual evaluators.
-Increasing the coverage of evaluated features.
-Enabling comparisons between alternative designs.
-Incorporating evaluation within the design phase of user interface development.

At present only a few model-based tools for user interface development have facilities for evaluation of a user interface. However, all of them are built into a tool for development and cannot be expanded. These tools quickly become out of day because interface elements are modified, requirements to their design are changed, and new standards are established. So an expandable system of automated detection of usability defects is a problem of urgency.

The Basic Idea of the Approach

The principal requirements to a system of automated detection of usability defects are expandability of the system, informing the developer about defects, and giving advice on its elimination.

The author has described a conception of user interface development based on ontologies in [2]. The main idea of this conception is to form an interface model using universal ontology models which describe features of every component of the model and then, based on this high-level specification, generate a code of the user interface. Components of the interface model are a domain model, a presentation model, a model of linking to an application program and a model of a dialog scenario. Every component of the interface model is formed by a structural or graphical editor managed by a domain-independent ontology model.

Similarly, a presentation model is formed by a graphical editor managed by a graphical user interface (GUI) ontology model. The GUI ontology model describes knowledge required for designing WIMP (windows, icons, menus, and pointing devices) interfaces. It consists of two basic groups of elements (windows and widgets) and three additional groups (control panels, menus and extra elements). Windows are main elements in a user interface since they make up its structure. Other elements are constituents of windows. Widgets (push and radio buttons, checkboxes, lists, etc.) manage an application program and specify properties of objects. Control panels are used to get quick access to commands.

Thus, the GUI ontology model describes interface elements, their properties and interconnections. It is platform-independent and expandable.

Example 1 shows a fragment of the GUI ontology for a text element of a menu.

Example 1. A fragment of the GUI ontology

The example shows the hierarchy of menu elements (see Fig. 1) and description of a text menu element.

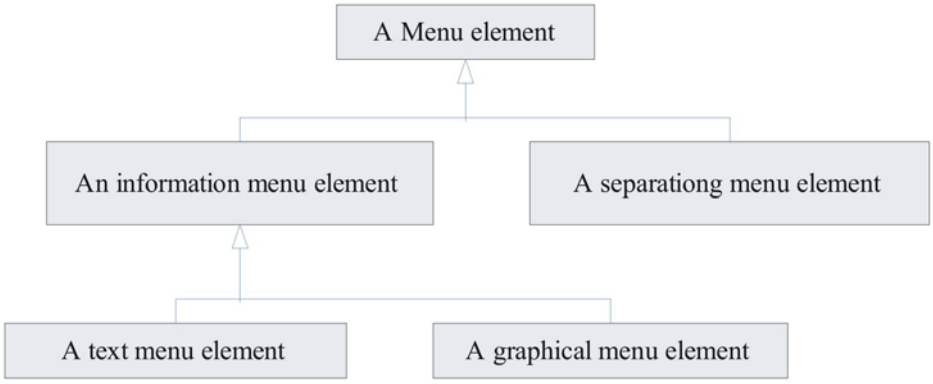


Fig. 1 The hierarchy of menu elements

A text menu element

Description: a class for presenting menu elements with verbal information.

Superclass: an information element of a menu.

Parameters:

Text: describes name to a menu element [type: String]

Prefix: describes prefix of the element [type: String]

Postfix: describes postfix of the element [type: String]

Font: describes font of the element [type: Font parameters]

Background: describes background color of the element [type: color]

A particular presentation component of a user interface model is a subset of the GUI ontology model. It means that to form a presentation component of the user interface model the developer is to determine values of properties of the GUI ontology model. This process requires that the developer should have expertise in usability principles; otherwise a presentation component a user interface model would have defects.

To detect these defects a knowledge base of interface defects has been made. Every element of this knowledge base is linked to elements of the GUI ontology model. It should be noted that since the GUI ontology model is expandable, when a new element is added to this ontology model, description of a defect in the knowledge base could be modified or a new description of a defect could be added.

There can be two ways to detect defects in the interface model. The first one is detecting a defect in designing an interface element, e.g., when the developer forms a string (a component of an interface element). If the length of this string exceeds a maximal length, the developer can be informed immediately. The second one is checking a set of properties in different interface elements. It is possible only after a fragment of an interface has been designed. For example, to detect a defect of an interface element arrangement in a window it is necessary to design this window first and then to check it. Therefore, the system of automated detection of usability defects is to work in two modes. Fig. 2 shows the basic architecture of the system.

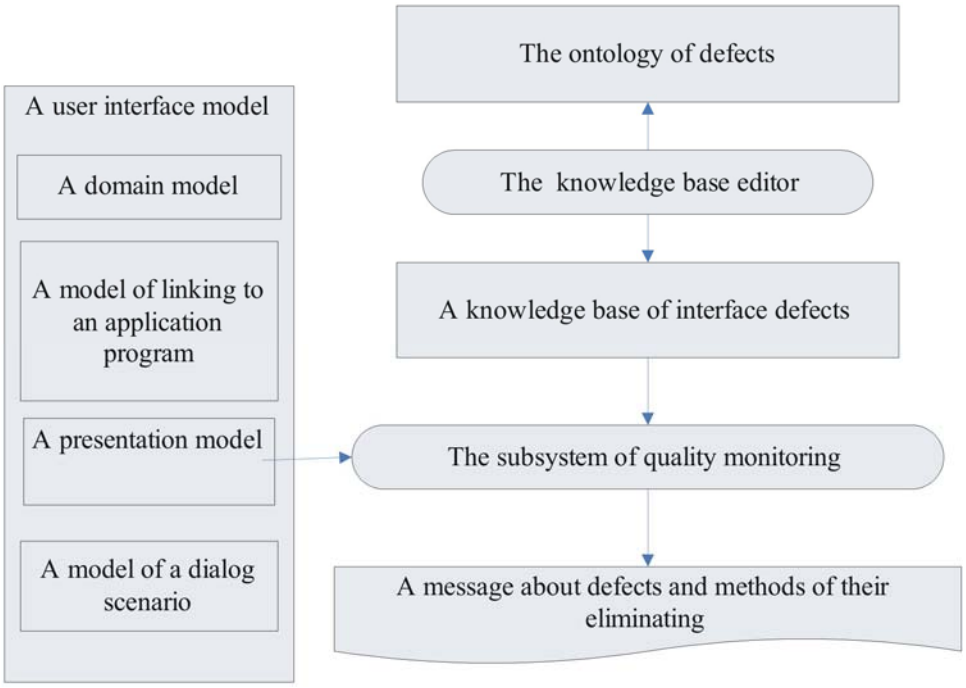


Fig. 2 The basic architecture of the system of automated detection of usability defects.

Ontology of Defects

A defect (fault) is detected in software when the developer makes a mistake due to a typo, poor understanding of some processes, principles, and so on. A defect is a coded mistake of the developer. To detect defects in software it is necessary to accurately classify them. The following ontology of defects is proposed.

1. ***Name of a defect***.

2 ***Type of a defect***. There may be two defect types, namely, presentation element defects or composition defects. The former occurs in designing an interface element; the latter is found after designing a set of different interface elements.

3. ***Name of a class***. It is a metaterm of the GUI ontology model. It indicates a class name of the interface element whose defect is described. This item can contain some classes. On the one hand, an interface element can consist of some classes; on the other hand, when a composition defect is described we must include all classes involved in detecting a defect.

4. ***Superclass***. It is also a metaterm of the GUI ontology model indicating a name of a parent class.

5. ***Parameters***. There are the parameters that are used for detecting a defect. They correspond to parameters of a class from the GUI ontology model.

6. ***Method of detecting***. It is an algorithm of detecting a defect.

7. ***Advice***. It is a message to the developer on eliminating a defect.

To illustrate the above, let's consider the following descriptions of defects from the knowledge base based on the ontology of defects.

Name of the defect: too many menu elements.

Type of the defect: a presentation element defect.

Name of the class: a top-level menu.

Superclass: menus.

Parameters: the number of menu elements.

Method of detecting: the number of menu elements>9

Advice. This menu consists of more than 9 elements. It will be difficult for the user to perceive. The number of menu elements should be decreased.

Name of the defect: the window has no name.

Type of the defect: a presentation element defect.

Name of the class: a window.

Superclass: an element of the GUI ontology model.

Parameters: the name (type: Boolean).

Method of detecting: the name=0

Advice. The window has no name.

Summary

In this article an approach to automated detection of usability defects is proposed. The basic idea of the approach is to add a system of automated detection of usability defects to the tool for user interface development operated by a knowledge base of interface defects. The main task of the system is to detect defects in a user interface model within the design phase and to give advice to the developer on their elimination.

At present a prototype of the system has been developed at the Intellectual Systems Department of the Institute for Automation and Control Processes, the Far Eastern Branch, the Russian Academy of Science.

The GUI ontology model and a knowledge base of interface defects corresponding with this ontology are used at the Mathematics and Computer Science Institute of the Far Eastern National University within the course "User Interfaces".

Acknowledgements

The research was supported by the Far Eastern Branch of Russian Academy of Science, the grant «An Expandable System for Quality Monitoring».

Bibliography

1. Ivory, M.Y., Hearst, M.A.: State of the Art in Automating Usability Evaluation of User Interfaces. ACM Computing Surveys, 33 (December 2001) 1–47. Accessible at http://webtango.berkeley.edu/papers/ue-survey/ue-survey.pdf.
2. Gribova V., Kleshchev A. From an Ontology-oriented Approach Conception to User Interface Development. International Journal "Information Theories & Applications". 2003. vol. 10, num.1, p. 87-94

Author's Information

Gribova Valeriya *– Ph.D. Senior Researcher of the Intellectual System Department, Institute for Automation & Control Processes, Far Eastern Branch of the Russian Academy of the Sciences: Vladivostok, +7 (4323) 314001; e-mail: gribova@iacp.dvo.ru; http://www.iacp.dvo.ru/es*

СИСТЕМЫ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ КАК ПЕРСОНАЛЬНЫЙ ИНТЕЛЛЕКТУАЛЬНЫЙ ИНСТРУМЕНТАРИЙ ЛИЦА, ПРИНИМАЮЩЕГО РЕШЕНИЕ

Алексей Ф. Волошин

Аннотация: *Тезисно обсуждаются проблемы развития познания «познающего» субъекта – от изучения объекта и способа действия до единой системы «субъект – способ действия – объект». Анализируются проблемы повышения адекватности моделей «живой» природы. Обсуждается концепция развития систем поддержки принятия решений как экспертных систем до систем поддержки принятия решений как персонального инструментария лица, принимающего решение. Анализируется опыт развития систем качественного прогнозирования на основе многопараметрических зависимостей, представляемых деревом решений, реализующих концепцию «множественного субъективного детерминизма». Приводятся примеры прикладных систем прогнозирования эколого-экономических и социальных процессов и обсуждаются пути их развития.*

Ключевые слова: *Практическое знание; способ действия; искусственный интеллект; системы поддержки принятия решений; дерево решений; экспертные системы; лицо, принимающее решение.*

Введение во введение

Автор солидарен с мыслью Нильса Бора, что «трудно что-либо предвидеть, особенно будущее». Почему? Потому, отвечают представители большинства теорий, учений и верований, что все зависит от неподконтрольных человеку сил внешнего мира. И не играет особой роли, как эти силы «проявляются» человеку.

Представители «стохастической неопределенности» считают, что влияние причины на следствие определяется «объективной вероятностью», которая является человеку в частоте появления событий, имеющей смысл при бесконечном числе испытаний. Им возражает А.Эйнштейн (и мы вместе с ним): «Я никогда не поверю, что Господь Бог играет в кости».

Детерминисты, наоборот, утверждают, что следствие однозначно определяется причиной, «написанной на небесах», однако – «пути Господни неисповедимы».

Христианское мировоззрение допускает «локальную» свободу сознательного выбора - «делай, что должен».

Атеистическая доктрина предполагает полную зависимость будущего от действий человека в прошлом и настоящем, главное – действовать в соответствии с «объективными законами», которые «открываются» человеку в процессе практической деятельности.

Многие восточные изотерические учения утверждают, что будущее формируется желаниями и волей человека, которые, в свою очередь, определяют принятие или не принятие тех или иных решений, на основе которых формируются те ситуации, в которых человек оказывается. Однако, в отличие от предыдущего, «практическое знание» не есть знание «управления деятельностью», а «знание управления личной волей».

Введение

Какова же роль «ученых» в структуре современного человечества, познающего мир и себя в этом мире? В частности, тех, кто занимается проблемами «искусственного интеллекта»? Для ответа на этот вопрос прежде всего необходимо уточнить понятия. Согласно определению Дж.Люгера [Люгер, 2003, с.781], искусственный интеллект (ИИ) в меньшей степени представляет собой теорию закономерностей, лежащих в основе разумного поведения («интеллекта»), и в большей – эмпирическую методологию создания и исследования моделей, на которые эта теория опирается. «Исследователи постепенно поняли, что интеллектуальные программы необходимо помещать прямо в предметную область, а не лелеять в лаборатории» [Люгер, 2003, с.802]. Задача ученых – создавать инструменты и «практическое знание», причем это «знание не объекта и не субъекта», а «способа действия» [Люгер, 2003, с.806].

Сужая понятия «объект», «субъект» и «способ действия» до их узкого, естественнонаучного понимания, можно утверждать, что развитие науки «нового времени» (от Декарта, Ньютона, Лейбница) как раз и шло от изолированного изучения объекта (в первую очередь, на основе моделей механики) и субъекта (психология, социология) и в какой-то мере их взаимозависимости (математическая биология, математическая экономика) до их взаимодействия (способа действия субъекта при изучении объекта, в частности, самого себя). По мнению автора, человечеству (во всяком случае, его «интеллектуальным» представителям) необходимо сделать еще один шаг, который заключается в переходе к созданию единой системы «субъект-способ действия-объект».

Особенности моделей «живой природы»

При моделировании процесса «неживой природы» принципы, как детерминизма, так и стохастической неопределенности, себя вполне оправдали. Так, можно «с высокой степенью точности» (с «большой вероятностью») на основе законов механики и учета случайных возмущений определять координаты ракеты через «часы – дни – годы». Степень же адекватности моделей социально-экономических процессов, в которых субъект является «активной компонентой», несравнимо более низкая. Они имеют очень высокую степень неопределенности результата, более того, этот результат, зачастую, является бессмысленным (т.е. в принципе непроверяемым). Причем это относится, как к нормативным моделям (которые отвечают на вопрос «что нужно делать, чтобы достичь желаемого?»), так и позитивным («что будет?»). В [Волошин, 2005] приводится анализ проблем принятия решений и обсуждаются пути их преодоления. В частности, при построении математических и информационных моделей социально-экономических процессов выделяются две основные проблемы – «субъективизация объективности» (непосредственный учет влияния субъекта на процесс принятия решения) и «объективизация субъективности» (компенсации влияния индивидуальных характеристик познающего субъекта – учет его «объективных» особенностей познания действительности).

В данной работе акцент делается именно на «способе действия» при познании объекта субъектом. Вместе с тем, автор попытается проиллюстрировать все подходы «борьбы с неопределенностью», упоминаемые во «Введении во введение».

Система поддержки принятия решений как экспертная система

В работах [Волошин, Пихотник, 1999], [Voloshin, Panchenko, 2001], [Волошин, Панченко, 2002], [Voloshin, Panchenko, 2003], [Волошин, 2005], [Волошин, Головня, 2005], большинство которых представлялось на конференциях KDS, развивается концепция «качественного прогнозирования на основе многопараметрических зависимостей, представляемых деревом решений» [Волошин, Панченко, 2002]. Идейная основа этой концепции – «множественный субъективный детерминизм». Считается, что следствие определяется множеством взаимозависимых причин, степень влияния которых на следствие определяется «субъективно» (экспертным измерением). Чем больше параметров, которые «формируют» причину, тем лучше (для адекватности модели), однако это приводит к сложностям в анализе модели (возникает «проклятие размерности», с которым необходимо бороться [Волошин, Панченко, 2002], в частности, и методами искусственного интеллекта).

В работах [Voloshin, Panchenko, 2003] и [Волошин, Головня, 2005] описывается инструментальная система («задача ученых – разрабатывать инструменты», см. выше) создания прикладных систем поддержки

принятия решений (СППР) в различных областях. Построение прикладной СППР сводится к выделению экспертами проблем и подпроблем (вершин дерева) и связей между ними (дуг дерева). Экспертами далее определяются веса (вероятности) переходов между вершинами. Допускаются нечеткие оценки экспертов с помощью логических переменных, описываемых значениями функции принадлежности (векторами действительных чисел от 0 до 1). Каждый эксперт задает три оценки – оптимистическую, реалистическую и пессимистическую, скаляризация которых осуществляется с учетом психологического типа эксперта. Тип определяется на основании психологических тестов, заложенных в систему. На основе психологических тестов определяются также коэффициенты «правдивости», «независимости», «осторожности» и т.д.

Дерево строится на основе коллективных оценок экспертов с применением метода попарных сравнений. Для построения результирующего дерева применяются алгебраические методы обработки экспертной информации, в качестве расстояния между ранжировками применяется метрика Хемминга и мера несовпадений рангов объектов. Результирующее дерево определяется как медиана Кемени-Снелла:

$$Arg\min_{A}\sum_{i=1}^{n} d(A,A^i) \text{ или как компромисс: } Arg\min_{A}\max_{i} d(A,A^i),$$

где A^i - матрица, задаваемая i-м экспертом, в которой элемент a_{ij} =1 тогда и только тогда, когда i-я вершина предпочтительнее для эксперта j-ой, a_{ji} = -1, для равноценных объектов a_{ij} =0, a_{ii} =0.

В случае задания преимущества в нечеткой форме элементы матрицы задаются через функции принадлежности.

Для определения оптимальных путей в дереве предлагаются алгоритмы последовательного анализа вариантов [Волошин, Панченко, 2002], позволяющие обрабатывать деревья с сотнями вершин.

Дерево решений задается таблицами. Каждая таблица – это отдельный уровень дерева, каждая строка таблицы – отдельная вершина на этом уровне. Каждый элемент строки – это вероятность, с которой возможен переход из данной вершины в вершину нижнего уровня. Эти вероятности задаются функциями принадлежности, представляющие собой вектора действительных чисел от 0 до 1 любой длины. Таблица заполняется путем опроса экспертов. Существующие функции позволяют добавлять столбцы, строки, задавать словарь (который позволяет вербальным оценкам эксперта ставить в соответствие вероятности, путем задания определенных уровней), сохранять таблицы в файле, считывать таблицы из файла.

Экспертным путем задаются матрицы – результат сравнения вариантов вершин, которые могут быть включены в дерево. На основе анализа матриц определяются вершины, которые включаются в дерево и вероятности, с которыми возможен переход в них из вершин верхнего уровня. Если дерево решения декомпозируется на несколько поддеревьев, которые имеют одинаковые листья, вначале вычисляются вероятности этих листьев в каждом из них, а затем находятся вероятности для всего дерева в целом.

Создан ряд прикладных систем – прогнозирование курса валют, опосредованного расчета валового национального продукта, диагностики сердечно-сосудистых заболеваний, прогнозирование индекса инфляции и др. [Волошин, Пихотник, 1999], [Voloshin, Panchenko, 2001], [Волошин, Панченко, 2002], [Voloshin, Panchenko, 2003], [Волошин, 2005], [Волошин, Головня, 2005].

На конференции KDS-1999 приводился результат расчета курса национальной валюты (гривны) на 01.01.2001, который анализировался на KDS-2001. Точность прогноза оказалась равной ±2% в то время, как прогноз абсолютного большинства зарубежных и отечественных государственных и частных организаций выходил за 50% (в обе стороны). Прогноз индекса инфляции в Украине на 2005 год, выполненный в июне 2005 года в дипломной работе Сатир В.В., равнялся 12,8%. В бюджете Украины было заложено 9,8%, официальная статистика по результатам 2005 года дает 10,5%, международные организации – 12,5-13,0%. Высокая точность прогноза, на наш взгляд, обуславливается «объективной» причиной – учетом большого числа разнородных взаимосвязанных причин, влияющих на результат. При построении дерева решений, прогнозирующего курс валют, учитывались экономические, финансовые, политические (изменения в законодательстве, возможности отставки правительства и т.п.) и т.д. параметры, характеризующие социально-экономическое «состояние» как Украины, так и стран ближнего зарубежья и всей мировой системы. Вторая причина, по мнению автора, «субъективная» - это «узкоспециальная» экспертная оценка, эксперт зачастую и не догадывается, что он в конечном счете прогнозирует.

Разрабатываемые модели качественного прогнозирования на основе дерева решений имеют еще одну интересную особенность, которую отметил еще Поппер в 1959 г. [Люгер, 2003] – «научные теории должны ошибаться». То есть, должны существовать обстоятельства, при которых модель не может успешно аппроксимировать явление. Это связано с тем, что для подтверждения правильности модели не достаточно никакого конечного числа подтверждающих экспериментов. Ошибки в существующих теориях должны стимулировать дальнейшие исследования. Этому в полной мере соответствует рассматриваемая модель – если прогнозное значение не соответствует действительности, то это всего лишь означает, что не учтены некоторые факторы (которые возможно появились на интервале прогнозирования) или неверно оценены степени влияния параметров (что не мешает сделать это в дальнейшем). Или – отказаться от принципа «влияния причины на следствие»!

СППР как персональный интеллектуальный инструментарий ЛПР

При использовании разработанного инструментария, описанного в предыдущем разделе, для разработки прикладных систем диагностики заболеваний [Волошин, Головня, 2005] явно определились ограничения в применимости используемых СППР как экспертных систем, усредняющих знание и опыт экспертов. Так, при диагностике наиболее сложнодиагностируемых психических заболеваний «эксперты» из Московской и Санкт-Петербургской школ (обе признаны в мире) зачастую оценивают степень взаимовлияния параметров взаимоисключающим образом. Таким образом, получаем «среднюю температуру по госпиталю».

В результате мы пришли к следующей концепции разработки СППР – СППР должна быть не экспертной системой, а «интеллектуальным усилителем» лица, принимающего решение, другими словами, «персональным интеллектуальным инструментарием» ЛПР. Общение с врачами-диагностами (в частности, с нашим соавтором в [Волошин, Головня, 2005]) убедило нас в том, что все они сознательно или подсознательно следуют «теории внедренного и осуществленного действия» [Люгер, 2003], т.е. «опыта действия» (см. выше). Здесь уместно привести два следующих соображения. Первое – по данным Всемирной организации здоровья смертность от неправильного диагноза стоит в мире на пятом месте. Второе – со времен Авиценны, который утверждал, что «врач – это человек, который лечит от болезни, о которой знает очень мало, лекарством, о котором знает еще меньше, человека, которого вообще не знает», мало что изменилось. Какой же выход? «Промоделировать» способ действия ЛПР (в данном случае, врача), а не строить модель на основе его знания. Так, опытный врач учитывает десятки и сотни параметров, взаимосвязь и взаимовлияние которых он не может оценить в принципе. Поэтому ему не остается ничего другого, как выделить «основные» параметры, отбрасывая «второстепенные». А это может привести к непредсказуемому результату. Общение с опытным врачом, наблюдающим за сотнями больных, подтвердило наше представление о принципе «внедренного действия» при установлении диагноза – диагноз устанавливается именно на основании построения (сознательного или подсознательного) дерева решений. Поэтому задача заключается в предоставлении ЛПР инструментария для представления и обработки «его» дерева решений. Здесь важно отметить, что информация о самом дереве установления диагноза может иметь конфиденциальный характер. Если полученный «компьютерный» диагноз не совпал с интуитивным представлением ЛПР (или с «истинным» диагнозом, установленным паталогоанатоном), необходимо обеспечить средствами для осуществления обратной связи для коррекции дерева, в частности, разработать эффективные алгоритмические процедуры анализа дерева на «чувствительность», что является одной из первоочередных наших задач при развитии описанного подхода.

Выводы

Не отрицая необходимости создания систем поддержки принятия решений как экспертных систем, автор уверен в расширении сферы использования СППР как «персональных ИИ - систем», «настраивающихся» на конкретного пользователя. В первую очередь, это касается «творческих» сфер деятельности человека (примером может служить медицинская диагностика). Здесь уместна аналогия с историей появления и развития персональных компьютеров.

Список литературы

[Люгер, 2003] Люгер Ф.Дж. Искусственный интеллект. Стратегии и методы решения сложных проблем. Москва: «Вильямс», 2003. 264с.

[Волошин, Пихотник, 1999] Волошин А., Пихотник Е. Экспертная система прогнозирования курса гривны. "Искусственный интеллект", 1999, №2. С.354-359 (укр.).

[Voloshin, Panchenko, 2001] Voloshin O.F., Panchenko M.V. The Forecasting of Stable Processes by a Tree Solution Method using a Pairwise Comparison Method for Analysis of Expert Information. Труды международной конференции «KDS-2001», Том 1, Санкт-Петербург, 2001. С.50-53 (англ.).

[Волошин, Панченко, 2002] Волошин А.Ф., Панченко М.В. Экспертная система качественного оценивания на основе многопараметрических зависимостей. "Проблемы математических машин и систем", 2002, №2. С.83-89 (укр.).

[Voloshin, Panchenko, 2003] Voloshin O.F., Panchenko M.V. The System of Quality Prediction on the Basis of a Fuzzy Data and Psychografphy of the Experts. International Journal “Information Theories & Applications”, 2003, Vol.10, №3. P.261-265.

[Волошин, 2005] Волошин А.Ф. О проблемах принятия решений в социально-экономических системах. Труды конференции «KDS-2005», Том 1, София, 2005. С.205-212.

[Волошин, Головня, 2005] Волошин А.Ф., Головня В.М. Система качественного прогнозирования на основе нечетких данных и психографии экспертов. Труды конференции «KDS-2005», Том 1, София, 2005. С.237-243.

Сведения об авторе

Волошин Алексей Федорович *– Киевский национальный университет имени Тараса Шевченко, факультет кибернетики, профессор. Киев, Украина. E-mail: ovoloshin@unicyb.kiev.ua*

ПРОБЛЕМЫ ПРИНЯТИЯ КОЛЛЕКТИВНЫХ СОЦИАЛЬНО-ЗНАЧИМЫХ РЕШЕНИЙ

Алексей Ф. Волошин, Павел П. Антосяк

Аннотация: *К задаче построения коллективного ранжирования по индивидуальным ранжированиям сводятся многочисленные практические задачи, в частности, социально-значимые. Основная проблема использования соответствующих математических моделей и вычислительных методов – «проклятие размерности». Предлагаются математические модели, адекватно представляющие «идеологию» лица, принимающего решение. Описывается алгоритм построения коллективного ранжирования, базирующийся на методе анализа, конструирования и отсеивания решений для аддитивной свертки. Приводятся результаты экспериментальных исследований алгоритма.*

Ключевые слова: *Задача коллективного принятия решений; ранжирование, матрица парных сравнений, ацикличность; метод идеальной точки; последовательный анализ вариантов; медиана Кемени-Снелла.*

Введение

Одной из наиболее распространенных задач линейного упорядочения (ранжирования) объектов является задача нахождения результирующего (коллективного, группового, согласованного) ранжирования по индивидуальным ранжированиям, заданных отдельными экспертами (субъектами). К этой задаче сводятся многочисленные задачи, возникающие в проектировании технических устройств, диагностике заболеваний, усреднении индивидуальных мнений в социологии, выработке коллективных решений в политике и т.д. Практика требует рассмотрения подобных задач с десятками и сотнями объектов и сотнями и тысячами экспертов. Так, за индивидуальными списками кандидатов в депутаты, представляемыми делегатами съезда партии (блока партий), необходимо в Центральную избирательную комиссию представить список от партии (блока). На последних выборах в Верховный Совет Украины в

марте 2006 года 45 партий и блоков представляли свои списки кандидатов в депутаты, содержащие десятки и сотни лиц (от десятков до 450), принятые съездами, руководящими органами партий и блоков на основе «интегрирования» десятков, сотен и тысяч списков «экспертов». Процедуры (методы, алгоритмы) создания коллективных списков широкой общественности не известны, понятно только, что они имели эвристический характер (не исключено даже, что «диктаторский», «волюнтаристический»). Не менее социально-значимой проблемой является установление очередности рассматриваемых вопросов в законодательных и исполнительных учреждениях, поскольку известно, что от этого зависит результат принимаемых решений [Мулен, 1991]. По мере развития демократических ценностей желательно, во-первых, упоминаемые процедуры осуществлять «демократически» (то есть, в каком-то понимании учитывать мнение каждого члена группы, партии, общества), во-вторых, эти процедуры должны быть «научно» обоснованными. Первое требование приводит к необходимости определить критерий «близости» коллективного порядка к индивидуальному. В свою очередь, группа, партия, блок должны отдавать себе отчет в том, что ее «идеология» реализуется (может быть реализована) в терминах определенных «математических моделей», используемых на практике и доказавших свою эффективность. Так, «коммунистическая» идеология предполагает полное равенство индивидуальных предпочтений, что приводит к построению коллективного предпочтения, равноудаленного от индивидуальных. «Социал-демократическая» идеология обращает внимание прежде всего на наиболее «обездоленного», что приводит к максиминному критерию (максимизируется минимальный доход или минимизируются максимальные потери). «Либеральная» идеология, не обращая внимания на распределение благ среди членов сообщества, стремится максимизировать «коллективное благо».

Задача коллективного принятия решений

Формально сформулированную выше задачу можно описать следующей математической моделью:

$$\max \left\{ u_i(x) \middle| x \in X \subseteq E^n \right\},\ i \in I = \{1,...,n\},$$

где u_i – функция полезности i-го эксперта, значение которой зависит от «ситуации» x, которая определяется набором стратегий всех экспертов.

Тогда три выше сформулированные критерия близости ситуации x к коллективному решению x^* приводит к следующим задачам:

1. $u_1(x) = u_2(x) = ... = u_n(x) \to \max\limits_{x \in X}$,
2. $\min\limits_{i=\overline{1,n}} \{u_i\} \to \max\limits_{x \in X}$,
3. $\sum\limits_{i \in N} u_i(x) \to \max\limits_{x \in X}$.

В постановках 1-3 можно ввести весовые коэффициенты для учета «важности» («компетентности») экспертов.

Постановка задачи построения коллективного ранжирования

Пусть на фиксированном множестве объектов $O = \{o_1,...,o_n\}$ экспертами, нормированные коэффициенты компетентности α_l, $l \in L = \{1,...,m\}$ которых известны, заданы матрицы парных сравнений P^l, $l \in L$. Элементы $p_{ij}^l \in \{0,1\}$ матриц P^l представляют собой результат сравнения l-ым экспертом объектов o_i и o_j, $i, j \in I = \{1,...,n\}$, $i \neq j$:

$$p_{ij}^l = \begin{cases} 1, \textit{если объект } o_i \textit{ лучше чем } o_j, \\ 0, \textit{если } o_i \textit{ хуже чем } o_j. \end{cases} \quad (1)$$

Одним из методов нахождения результирующего ранжирования является определение медианы Кемени-Снелла (соответствующей третьей задаче из предыдущего раздела):

$$R^* \in Arg\min_{R\in\Re}\sum_{l\in L}\alpha_l d(R,P^l), \tag{2}$$

где $\Re$ – множество всех матриц, которые соответствуют строгим ранжированиям n объектов (ранжирование и матрицы, которые им отвечают, будем обозначать одинаковыми символами), $d(R,P^l)=\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\left|r_{ij}-p_{ij}^l\right|$ – расстояние Хемминга между R и P^l.

Свойства ацикличности бинарных отношений

Согласно [Мулен,1991], ацикличным отношением на O называется полное бинарное отношение R, у которого строгая компонента $R^{(s)}$ не имеет циклов: не существует последовательности объектов $o_1,o_2,...,o_T$, такой, что $o_tR^{(s)}o_{t+1}$, $t=\overline{1,T-1}$ и $o_TR^{(s)}o_1$. Обозначение $o_tR^{(s)}o_{t+1}$ означает, что выполняется отношение o_tRo_{t+1} и не выполняется $o_{t+1}Ro_t$.

З а м е ч а н и е 1 [Макаров,1982]. Для рассматриваемого случая, в силу свойств матриц с элементами вида (1), отсутствие (присутствие) циклов эквивалентно отсутствию (присутствию) циклов длины 3.

Следовательно, присутствие в отношении R циклов означает, что существуют такие индексы объектов $i_1,i_2,i_3\in I$, для которых выполняется:

$$o_{i_1}Ro_{i_2}, o_{i_2}Ro_{i_3}, o_{i_3}Ro_{i_1}. \tag{3}$$

Для применения последовательного алгоритма [Волкович,1978] построения ацикличного решения задачи (2), приведем некоторые определения и утверждения.

Определение1. Цикличностью элемента r_{ij}, $i,j\in I$, матрицы, которая соответствует отношению R, назовем количество циклов вида (3), в которых объекты o_i и o_j находятся в отношении o_iRo_j.

Определение2. Элемент с ненулевой цикличностью назовем цикличным элементом отношения R.

Для фиксированного, но каждого, объекта $o_\phi\in O$, $\phi\in I$, определим множество пар индексов:

$$C_\phi=\{(i,j)\,|\,r_{ij}=1, i\in I_\phi^1, j\in I_\phi^0\}, \tag{4}$$

где

$$I_\phi^0=\{i\,|\,r_{\phi i}=0, i\in I\setminus\{\phi\}\},\ I_\phi^1=\{i\,|\,r_{\phi i}=1, i\in I\setminus\{\phi\}\}.$$

Очевидно, что множество C_ϕ описывает все пары объектов с индексами i_1 и i_2, которые находятся в отношении R, и для которых выполняется $o_\phi Ro_{i_1}$ и $o_{i_2}Ro_\phi$. То есть, для любого фиксированного $\phi\in I$ множество C_ϕ описывает все циклы вида $o_\phi Ro_{i_1}$, $o_{i_1}Ro_{i_2}$, $o_{i_2}Ro_\phi$. Тогда справедливо

Утверждение1. Множество $C=\bigcup_{i\in I}C_i$ описывает множество всех цикличных элементов отношения R.

Справедливость утверждения 1 вытекает из правила построения множеств C_i, $i\in I$, и замечания 1.

Утверждение 2. Отношение R ациклично тогда и только тогда, когда $C=\varnothing$.

Доказательство утверждения 2 следует непосредственно из замечания 1 и предыдущего утверждения.

Алгоритм нахождения строгого коллективного ранжирования

Пусть R - бинарное отношение, на котором целевая функция задачи (2) достигает своего минимального значения, но которое не является ацикличным. $R=(r_{ij})$, $i,j\in I$, – матрица данного бинарного отношения. Поскольку $R\notin\Re$, то R не может выступать в качестве решения. Поэтому такое бинарное отношение можно назвать «идеальной точкой» задачи (2).

Рассмотрим вопрос построения ацикличного отношения, которое в каком-то понимании является ближайшим к «идеальной точке».

Ацикличности бинарного отношения R можно достичь путем последовательного исключения циклических элементов. Пусть r_{ij}^{u} – некоторый цикличный элемент бинарного отношения R. Удалением цикличного элемента назовем правило:

$$r_{ij}^{u} := 0,\ r_{ji}^{u} := 1. \qquad (5)$$

З а м е ч а н и е 2. Удаление цикличного элемента приводит к:

1) уменьшению цикличности (в частности, возможно удаление цикличных элементов);
2) увеличение цикличности (не исключается появление новых цикличных элементов).

Учитывая замечание 2, введем следующие обозначения. Для любого элемента r_{ij}, $i, j \in I$, бинарного отношения R через $\tilde{c}_{ij}$ обозначим его цикличность, c_{ij}^{-} – количество цикличных элементов, цикличность которых уменьшается после удаления элемента r_{ij}, c_{ij}^{+} – количество элементов, цикличность которых увеличивается после удаления элемента r_{ij}.

Таким образом, каждому бинарному отношению поставим в соответствие матрицу цикличностей $C = (c_{ij})$, $i, j \in I$, где $c_{ij} = \tilde{c}_{ij} + c_{ij}^{-} - c_{ij}^{+}$.

С учетом приведенных результатов можно построить алгоритм нахождения решения задачи (2), каждый итерационный шаг которого формально описывается следующим образом.

Шаг1. Для каждого $i \in I$ по правилу(4) строим множество C_i и определяем множество всех цикличных элементов $C := \bigcup_{i \in I} C_i$. Переход к шагу 2.

Шаг2. Если $C = \varnothing$, то стоп. Иначе переход к шагу 3.

Шаг3. Для всех $i, j \in I$ принимаем $\tilde{c}_{ij} := 0$, $c_{ij}^{-} := 0$, $c_{ij}^{+} := 0$. Переход к шагу 4.

Шаг4. Для всех $i, j \in I$ и для всех $k \in I$ $\tilde{c}_{ij} := \tilde{c}_{ij} + 1$, если $(i, j) \in C_k$; $c_{ij}^{-} := c_{ij}^{-} + 1$, если $(j, k) \in C$; $c_{ij}^{+} := c_{ij}^{+} + 1$, если $k \in I_i^1 \bigcap I_j^0$; $c_{ij} := \tilde{c}_{ij} + c_{ij}^{-} - c_{ij}^{+}$. Переход к шагу 5.

Шаг5. «Удаление» элемента $r_{i^* j^*}$, где $i^*, j^* \in Arg \max_{i,j \in I} c_{ij}$. Если таких элементов несколько, то «удаляем» тот элемент $r_{\bar{i}\bar{j}}$, для которого $\bar{i}, \bar{j} \in Arg \min_{i,j \in I} \sum_{l \in L} \alpha_l p_{ij}^{l}$. Если, в свою очередь, такой элемент не единственный, то «удаляем» один из элементов $r_{\tilde{i}\tilde{j}}$, где

$$\tilde{i} \in Arg \min_{i \in I} \sum_{j \in I} \sum_{l \in L} \alpha_l p_{ij}^{l}. \qquad (6)$$

«Удаление» цикличного элемента по правилу (5), т.е. принимаем $r_{\tilde{i}\tilde{j}} := 0$, $r_{\tilde{j}\tilde{i}} := 1$. Переход к шагу 1.

Неоднозначность выбора элементов из множества (6), а также тот факт, что анализируется «локальный отрезок» пути, не позволяют, в общем случае, получить глобально-оптимальное решение задачи (2).

Улучшение локально-оптимального решения возможно путем увеличения длины анализируемого отрезка с использованием «параллельных процедур», разработкой параллельно-декомпозиционных алгоритмов, а именно параллельного анализа отрезков пути с последующим конструированием решения из оптимизированных отрезков [Волошин,1987].

Вычислительный эксперимент

С целью экспериментального исследования предложенного алгоритма, разработан программный комплекс, который позволяет формировать и решать тестовые задачи. В вычислительном эксперименте решалась серия задач со случайными данными (использовался датчик псевдослучайных чисел с равномерным распределением). Решались задачи размерностью (количество объектов) от 3 до 8 (такая размерность задач позволяет сравнить найденное решение с глобально-оптимальным, которое

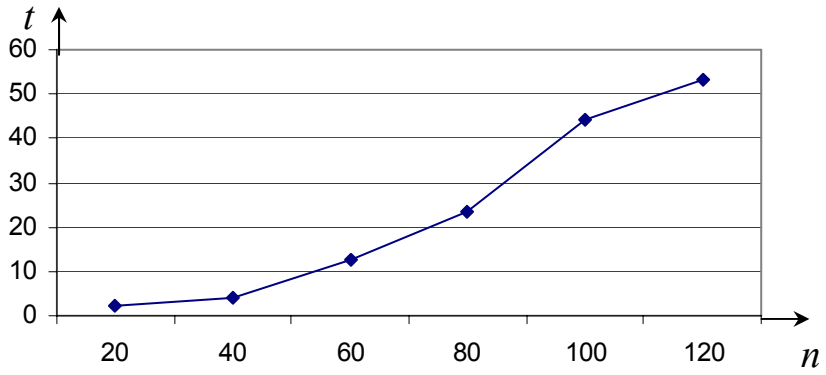
находилось прямым перебором). Генерировалось по 10 тестовых задач для каждого случая. Количество экспертов менялось от 1 до 10 (все эксперты считались равнокомпетентными). Изучалось отклонение $\Delta = f - f^*$ значения f, найденного по алгоритму, от оптимального значения f^*. В табл.1 для каждой размерности приведены минимальное ($\Delta_{\min}$), максимальное ($\Delta_{\max}$), среднее (Δ_c) отклонения, а также совпадение в процентах (%) найденного решения с глобальным оптимумом.

Таблица 1.

n	$\Delta_{\min}$	$\Delta_{\max}$	Δ_c	%
3	0	0	0	100
4	0	0	0	100
5	0	2	0.2	90
6	0	8	1.4	70
7	0	12	3.8	40
8	0	22	4.2	40

На рис.1 представлено значение среднего времени решения одной задачи данной размерности (в минутах, использовался персональный компьютер на базе процессора AMD Athlon(tm)XP 1800+ с тактовой частотой 1.54 GHz и оперативной памятью 256 Mb).

Рисунок 1.



Список литературы

[Мулен, 1991] Мулен Э. Кооперативное принятие решений: Аксиомы и модели. Москва: Мир,1991. 464с.

[Макаров,1982] Макаров И.М, Виноградская Т.М. и др. Теория выбора и принятия решений. Москва: Наука, 1982. 328с.

[Волкович,1978] Волкович В.Л., Волошин А.Ф. Об одной схеме метода последовательного анализа и отсеивания вариантов. Кибернетика. 1978. №4. с. 98-105.

[Волошин, 1987] Волошин А.Ф. Метод локализации области оптимума в задачах математического программирования. Доклады АН СССР. 1987. Т. 293, № 3. с. 549–553.

Сведения об авторах

Волошин Алексей Федорович *– Киевский национальный университет имени Тараса Шевченко, факультет кибернетики, профессор. Киев, Украина, e-mail: ovoloshin@unicyb.kiev.ua*

Антосяк Павел Павлович *– Киевский национальный университет имени Тараса Шевченко, факультет кибернетики, аспирант. Киев, Украина, e-mail: AntosP@ukr.net*

РОЛЬ ФАКТОРА КУЛЬТУРНОЙ ИДЕНТИЧНОСТИ В ДВУЯЗЫЧНОМ ОБЩЕСТВЕ

Ирина Горицына, Александр Глущенко

Аннотация: *Рассматривается проблема, которая возникает в двуязычном обществе. Культурная идентичность населения может помочь сохранению языков меньшинств в районах их компактного проживания. С другой стороны, знание языка большинства в этой группе населения расширяет возможности в полной мере пользоваться своими правами и свободами.*

Ключевые слова*: функция полезности, теория игр, культурная идентичность. Отсутствие культурной идентичности у населения страны может привести к переходу всего населения на язык большинства.*

Введение

Рассмотрим общество, разговаривающее на двух языках ***{a, b}***. Для простоты предположим, что семьи состоят из одного родителя и одного ребенка. Жизнь людей разделяется на два периода: жизнь с родителем (детство), когда они получают образование, и самостоятельная жизнь.

Функция полезности родителя состоит из двух частей. Во-первых, владение языком приносит определенную пользу, во-вторых, немаловажную роль имеет и любовь (забота) родителя к своему ребенку.

Польза (полезность) от владения языком выбрана так, чтобы учесть преимущества, получаемые при возможности на бытовом уровне облегчить общение с людьми, говорящих на двух языках, что немаловажно, осознавая себя частью другой культуры или хотя бы понимая ее. Это учтено в модели отдельно от материального потребления, поскольку нет достаточных доказательств того, что знание определенного языка существенно влияет на потребление товаров. Различия, которые мы видим в моделях потребления, главным образом определяется культурными факторами (религия, раса, этническая принадлежность, и т.д.).

Задача максимизации родительской функции полезности включает затраты на образование ребенка. Родитель считает своим долгом передать ребенку свою этническую (или культурную) идентичность. Культурная идентичность однозначно связана со знанием одного из двух языков. Это представлено в функции полезности родителя. Она включает функции полезности потребления однородного товара своим ребенком и учитывает знание (наличие способностей) родного языка и косвенно учитывает знание второго (не родного) языка. Однако, знание второго языка приводит к увеличению родительской функции полезности через увеличение заработной платы a, следовательно, способствует возрастанию потребностей ребенка, владеющего двумя языками.

Языковая функция полезности (v) - одинакова для родителя и для ребенка. Однако, для того, чтобы промоделировать желание передать культурную идентичность, предположим, что родитель и ребенок получает одинаковую пользу от владения одним и тем же языком при условии, что и все общество разговаривает на этом же языке. Другими словами, он получает такую же пользу, как будто ребенок живет в одноязыковой, монокультурной стране. В противном случае, он не получает никакой пользы от передачи культурной идентичности. Функция максимизации также включает отрицательную компоненту, соответствующую затратам на обучение ребенка одному или двум языкам.

Наличие языковых навыков представлено булевой переменной s, которая показывает владение языком.

Функция социализация или функция изучения языка ***S*** зависит от знания родителем языка, вектора денежных "доходов" и удельного веса населения, которое разговаривает на этом языке. Функция зависит от булевых (двоичных) переменных и имеет ограниченные значения по каждому факторов. Предполагается, что обучение языку меньшинства обходится дороже в обществе, которое имеет другой язык большинства. В то время, как одинаковые потребности требуют одинаковых затрат, чем ниже удельный вес населения, разговаривающего на языке меньшинства, тем больше значение функции изучения языка ***S*.**

Бюджетные ограничения родителя и ребенка одинаковы. В доходной части бюджетного уравнения заработная плата родителя зависит от его языковых способностей. Предполагается, что языковые способности родителя известны и все люди вначале имеют некоторые знания другого языка. Заработная плата имеет постоянную компоненту ***W***, не связанную со знанием языков, и компоненту, обусловленную знанием языка и зависящую от доли населения, с которой человек может общаться, основываясь на своих знаниях языков.

Постановка задачи и решение

Задача Родителя состоит в максимизации функции полезности от потребления некоторых материальных благ в совокупности с некоторыми благами и преимуществами, которые дают культурная идентичность и знание языков, на которых разговаривают члены общества

$$U = u(x) + s^i_p \bullet v(q^i_p + q^{ij}_p) + s^j_p \bullet v(1 - q^i_p) + u(x_c(s^i, s^j)) + s^i \bullet v(1) - C(\tau^i + \tau^j) \to max$$

определить x, x_c, τ^i, τ^j в области допустимых решений (бюджетных ограничений для себя и своего ребенка)

$$px = W + w(s^i_p \bullet q^i_p + q^{ij}_p + s^j_p \bullet q^i_p)$$

$$px_c = W + w(s^i \bullet q^i + q^{ij} + s^j \bullet q^j),$$

где $s^i = S(s^i_p, \tau^i, q^i_p + q^{ij}_p)$, $s^i \in \{0,1\}$, $s^i_p \in \{0,1,\}$ $q^i + q^j + q^{ij} = 1$.

Условные обозначения:

$u(x)$	монотонно возрастающая, строго вогнутая функция полезности,
x	множество выборов родителя (наборы благ, на которые могут быть потрачены деньги);
x_c	множество выборов ребенка (наборы благ, на которые могут быть потрачены деньги);
$i \in \{a,b\}$	языки, на которых разговаривают, **a** - язык большинства, **b** - язык меньшинства;
s^i	языковая переменная ребенка, s^i =1, если ребенок имеет способности к языку **i**, s^i =0 в противном случае;
s^i_p	языковая переменная родителя, s^i_p =1, если родитель владеет языком **i**, s^i_p =0, в противном случае;
$q^i \in [0;1]$	удельный вес детей, говорящих только на языке **i**; $q^i = N(s^i=1 \,\&\, s^j=0)/N, \quad q^j = N(s^i=0 \,\&\, s^j=1)/N;$
$q^{ij} \in [0;1]$	удельный вес детей, говорящих на двух языках; $q^{ij} = N(s^i=1 \,\&\, s^j=1)/N, \quad q^i + q^j + q^{ij} = 1;$
$q^i_p \in [0;1]$	удельный вес родителей, говорящих только на языке **i**;
$q^{ij}_p \in [0;1]$	удельный вес родителей, говорящих на двух языках;
$v(q^i)$	дополнительная польза от владения языком **i** (монотонно возрастающая, строго вогнутая функция);
$v(1)$	функция полезности **i** –язычного родителя при наличии у ребенка способностей к родному языку;
$w(q^i)$	монотонно возрастающая, строго вогнутая функция заработной платы, связанная со знанием **i**-го языка;
$C(\tau^i)$	строго возрастающая функция стоимости, $C(0)=0, \; dC(0)/d\tau^i=0$;
τ^i	N-мерный вектор затрат;
S	механизм социализации, подготовки к жизни в обществе $S(0,0,0)=0, \quad dS/ds^i_p=0, \quad dS/d\tau^i=0, \quad dS/dq^i=0$;
N	численность населения.

Функция социализации устанавливает однозначное соответствие между s^i и τ^i при условии, что знания языков родителей и распределения населения относительно знаний языков известны заранее. Поэтому, введем новую переменную для обозначения затрат, необходимых для изучения языка. Обозначим минимальные затраты, необходимые для обучения ребенка языку при условии, что и родитель говорит на этом же языке:

$$\tau^{I} \rightarrow \min$$

при условии $S(1, \tau^{i}, q^{i}_{p} + q^{ij}_{p})=1$.

Пусть τ^{i}_{11} – значение целевой функции этой задачи. Естественно, что родитель заинтересован в минимизации затрат на обучение языку. Аналогично обозначим через τ^{i}_{01} минимальные затраты (значение целевой функции), необходимые для обучения ребенка языку, на котором родитель не разговаривает:

$$\tau^{i} \rightarrow \min$$

при условии $S(0, \tau^{i}, q^{i}_{p} + q^{ij}_{p})=1.$

Так как затраты определены в задаче максимизации функции полезности, то существует только две альтернативы: обучать ребенка языку, или нет. Следовательно, максимальное значение $\tau^{i} \in \{0, \tau^{i}_{11}\}$, если родитель говорит на языке ***i*** или $\tau^{i} \in \{0, \tau^{i}_{01}\}$, если он не говорит на этом языке.

Стабильность распределения языковых групп

В дальнейшем мы абстрагируемся от факта, что минимальные затраты, необходимые для обучения определенному языку τ^{a}_{11}, τ^{a}_{01} и т.д. зависят от времени, так как изменяется во времени и доля населения, которое говорит на языке $a - (q^{a}_{p} + q^{ab}_{p})$. Т.е., соответствующее значение должно быть равным $\tau^{i}_{11}(q^{i}_{p} + q^{ij}_{p})$. Однако для задачи оптимизации в рамках одного временного периода примем эти величины заданными.

Для нахождения решения с учетом возможного распределения населения в зависимости от знания языка (языков), будем считать величины q^{a}, q^{b} известными.

Определение расходов на образование ребенка можно рассматривать как кооперативную игру. Игроками является все население страны (региона). В зависимости от знания языков и от своей культурной идентичности все игроки образуют четыре коалиции, внутри которой все игроки соблюдают одну стратегию, т.е. действуют как один игрок. Стратегиями игроков является желание обучить своего ребенка одному или двум языкам.

Для исследования этих игр будем использовать равновесие Нэша в чистых стратегиях. Следовательно, игра может быть представлена как игра четырех игроков с тремя стратегиями у каждого игрока. Игроками являются:

- игрок, владеющий только языком a (**A**),
- игрок, владеющий двумя языками, отождествляющий себя с культурой a (**2A**),
- игрок, владеющий двумя языками, отождествляющий себя с культурой b (**2B**),
- игрок, владеющий только языком b (**B**).

В качестве трех стратегий выбраны:

- обучение своего ребенка языку ***a***,
- обучение ребенка двум языкам,
- обучение ребенка языку ***b***.

Представим схематически стратегии и выигрыши, принимая во внимание, что зависят q^{a}, q^{b}, $q^{2}=q^{ab}$ от выборов других игроков (табл. 1).

Таблица 1. Таблица выигрышей игроков

	Стратегии		
Игроки	Обучение языку ***a***	Обучение двум языкам	Обучение языку ***b***
Игрок **2A**	$u\{W + w(q^{a} + q^{2})\} + v(1) - C(\tau^{a}_{11})$	$u\{W + w(1)\} + v(1) - C(\tau^{a}_{11} + \tau^{b}_{11})$	$u\{W + w(1 - q^{a})\} - C(\tau^{b}_{11})$
Игрок **2B**	$u\{W + w(q^{a} + q^{2})\} - C(\tau^{a}_{11})$	$u\{W + w(1)\} + v(1) - C(\tau^{a}_{11} + \tau^{b}_{11})$	$u\{W + w(1 - q^{a})\} + v(1) - C(\tau^{b}_{11})$
Игрок **A**	$u\{W + w(q^{a} + q^{2})\} + v(1) - C(\tau^{a}_{11})$	$u\{W + w(1)\} + v(1) - C(\tau^{a}_{11} + \tau^{b}_{01})$	$u\{W + w(1 - q^{a})\} - C(\tau^{b}_{01})$
Игрок **B**	$u\{W + w(q^{a} + q^{2})\} - C(\tau^{a}_{01})$	$u\{W + w(1)\} + v(1) - C(\tau^{a}_{01} + \tau^{b}_{11})$	$u\{W + w(1 - q^{a})\} + v(1) - C(\tau^{b}_{11})$

Решение устойчиво для группы игроков **2A**, если

$$u\{W + w(1)\} + v(1) - C(\tau^a_{11} + \tau^b_{11}) \geq u\{W + w(q^a + q^2)\} + v(1) - C(\tau^a_{11}), \quad (1)$$

$$u\{W + w(1)\} + v(1) - C(r^a_{11} + r^b_{11}) \geq u\{W + w(1 - q^a)\} - C(\tau^b_{11}). \quad (2)$$

Решение устойчиво для группы игроков **2B**, если

$$u\{W + w(1)\} + v(1) - C(\tau^a_{11} + \tau^b_{11}) \geq u\{W + w(qa + q2)\} - C(\tau^a_{11}) \quad (3)$$

$$u\{W + w(1)\} + v(1) - C(\tau^a_{11} + \tau^b_{11}) \geq u\{W + w(1 - q^a)\} + v(1) - C(\tau^b_{11}) \quad (4)$$

Решение устойчиво для группы игроков **A**$(s^a = 1, s^b = 0)$, если

$$u\{W + w(q^a + q^2)\} + v(1) - C(\tau^a_{11}) \geq u\{W + w(1 - q^a)\} - C(\tau^b_{01}) \quad (5)$$

$$u\{W + w(q^a + q^2)\} + v(1) - C(\tau^a_{11}) \geq u\{W + w(1)\} + v(1) - C(\tau^a_{11} + \tau^b_{01}) \quad (6)$$

Решение устойчиво для группы игроков **B**$(s^a = 0, s^b = 1)$, если

$$u\{W + w(1-q^a)\} + v(1) - C(\tau^b_{11}) \geq u\{W + w(q^a + q^2)\} - C(\tau^a_{01}) \quad (7)$$

$$u\{W + w(1-q^a)\} + v(1) - C(\tau^b_{11}) \geq u\{W + w(1)\} + v(1) - C(\tau^a_{01} + \tau^b_1 1) \quad (8)$$

Модель без учета фактора передачи культурной идентичности

Чтобы понять, как фактор передачи культурной идентичности меняет задачу, посмотрим на равновесие Нэша в этой же игре, но при отсутствии этого фактора (т.е. **v(1)=0**).

Тогда условия устойчивости решения изменятся.

Для группы игроков **2A**

$$u\{W + w(1)\} - C(r^a_{11} + r^b_{11}) \geq u\{W + w(q^a + q^2)\} - C(r^a_{11}), \quad (9)$$

$$u\{W + w(1)\} - C(r^a_{11} + r^b_{11}) \geq u\{W + w(1 - q^a)\} - C(r^b_{11}). \quad (10)$$

Для группы игроков **2B**

$$u\{W + w(1)\} - C(r^a_{11} + r^b_{11}) \geq u\{W + w(q^a + q^2)\} - C(r^a_{11}), \quad (11)$$

$$u\{W + w(1)\} - C(r^a_{11} + r^b_{11}) \geq u\{W + w(1 - q^a)\} - C(r^b_{11}). \quad (12)$$

Первое, что можно заметить, пара условий (9-10) идентична условиям (11-12), т.е. эти игроки отличались только тем, что имели различную культурную идентичность. Отсутствие это фактора сделало эти группы идентичными, т. е. людьми, разговаривающими просто на двух языках. Язык используется исключительно как средство общения, и не существует различий в функциях полезности, и двуязычными являются те, кто говорит на двух языках.

В многоходовой игре (ход отождествляется с обучением поколения: дети становятся родителями, обучают своих детей, те, в свою очередь, становятся родителями и т.д.) проанализируем ситуацию с языками: во-первых будет доминировать стратегия изучения второго языка, т.е. наступит полное двуязычие.

Если группа игроков **B** начинает изучать второй язык, то q^b станет равным 0 и $w(q^a + q^2) = w(1)$ в уравнении (7) и получим уравнение (15)

Если группа игроков **A** начинает изучать второй язык, то q^a станет равным 0 и $w(1 - q^a) = w(1)$ в (5) и получим уравнение (13)

Таким образом, для игроков **A** ($s^a = 1, s^b = 0$) и получили такие неравенства устойчивости

$$u\{W + w(q^a + q^2)\} - C(r^a_{11}) \geq u\{W + w(1)\} - C(r^b_{01}), \quad (13)$$

$$u\{W + w(q^a + q^2)\} - C(r^a_{11}) \geq u\{W + w(1)\} - C(r^a_{11} + r^b_{01}). \quad (14)$$

Аналогично для группы игроков. B ($s^a = 0, s^b = 1$) –

$$u\{W + w(1-q^a)\} - C(r^b_{11}) \geq u\{W + w(1)\} - C(r^a_{01}), \quad (15)$$

$$u\{W + w(1-q^a)\} - C(r^b_{11}) \geq u\{W + w(1)\} - C(r^a_{01} + r^b_{11}). \quad (16)$$

Сравним неравенства (12) и (15): правая часть (12) совпадает с левой частью (15)

Изучать языки ***ab*** предпочтительнее изучения языка ***b*** (следует из (12)), а изучать язык ***b*** предпочтительнее ***a*** (следует из (15)). Следовательно, изучать языки ***ab*** предпочтительнее ***a***.

Аналогично сравнивая уравнения (11) и (13) получим утверждение: изучать языки ***ab*** предпочтительнее ***b***.

Такие же утверждения вытекают из условий стабильности групп игроков **2A** и **2B.**

Аналогично будет происходить переход от двуязычия к ***a***-одноязычию, поскольку первоначально мы предположили, что язык ***a*** – это язык большей части населения. Это можно увидеть, изучив уравнение (11), которое в предположении, что $\mathbf{w(q^a + q^2)=w(1)}$ приобретает вид

$$\mathbf{u\{W + w(1)\} - C(r^a_{11} + r^b_{11})>= u\{W + w(1)\} - C(r^a_{11}).}$$

Следовательно, $\mathbf{C(r^a_{11}+r^b_{11})\leq C(r^a_{11})}$. Получили противоречие, так затраты на изучение двух языков не могут быть меньше затрат на изучение одного языка. Значит, двуязычие в результате многоходовой кооперативной игры должно смениться ***a***-одноязычием.

Таким образом, если общество игнорирует культурный фактор, то двуязычие постепенно сменится полным доминированием языка большей части населения. Только экономическими выгодами нельзя объяснить изучение двух языков в любой группе игроков. Все люди, говорящие на двух языках, перейдут на язык большинства населения. Только появление компоненты полезности от культурной идентичности (осознание своей культурной идентичности) может оправдать усилия по поддержке языка меньшинства как культурной ценности, при возможном изучении языка большинства, как средства связи с большинством и всеми выгодами в заработной плате, которые это сопровождают.

Библиография

[Волошин, 2006] Волошин А.Ф., Мащенко С.О., Теория принятия решений. Учебное пособие.–Киев: ИПЦ “Киевский угиверситет”, 2006.–304 с. (на укр.яз).

[Хартия, 1992] Европейская хартия региональных языков или языков меньшинств.– Страсбург 5 ноября 1992 г.

[Василенко, 2006] Василенко В. Какие языки в Украине нуждаются в особой защите. Коллизия между национальным законом и обязательствами государства // «Зеркало недели», № 10 (589) Суббота, 18 - 24 Марта 2006 года

Информация об авторах

Ирина Горицына *– Киевский национальный университет им. Т. Шевченко, старший научный сотрудник, факультет кибернетики; пр. Акад. Глушкова 2, стр. 6, Киев, Украина; e-mail: goritsyna@unicyb.kiev.ua*

Александр Глущенко – *Киевский национальный университет им. Т. Шевченко, заведующий лабораторией, факультет кибернетики; пр. Акад. Глушкова 2, стр. 6, Киев, Украина; e-mail: mmeed@unicyb.kiev.ua*

УСТОЙЧИВОСТЬ ПО ОГРАНИЧЕНИЯМ ВЕКТОРНЫХ ЗАДАЧ ЦЕЛОЧИСЛЕННОЙ ОПТИМИЗАЦИИ С ДИЗЪЮНКТИВНЫМИ ЛИНЕЙНЫМИ ФУНКЦИЯМИ ОГРАНИЧЕНИЙ

Наталия В. Семенова

Резюме*: Рассматриваются векторные задачи целочисленной оптимизации на допустимом множестве, являющемся объединением (возможно бесконечного числа) выпуклых множеств, каждое из которых описывается конечным или бесконечным числом линейных неравенств. Исследовано влияние возмущений коэффициентов дизъюнктивных функций ограничений на поведение эффективных, неэффективных и различных множеств оптимальных решений (Парето-оптимальных, строго и слабо эффективных). Проведен сравнительный анализ различных типов устойчивости по ограничениям указанных задач*

Ключевые слова*: векторная оптимизация, целочисленное программирование, дизъюнктивные функции, устойчивость.*

Введение

Современный интерес к исследованию проблем корректности многокритериальных моделей обусловлен в значительной степени их широким применением для решения важных задач экологии, экономики, управления, проектирования различных сложных систем, принятия решений в условиях неопределенности и многих других. Он также связан с неточностью входных данных, неадекватностью моделей, которые используются, ошибками численных методов и другими факторами. Существует достаточно много оптимизационных, в том числе дискретных задач, для которых как угодно малые ошибки в исходных данных порождают значительные искажения истинного искомого решения. В связи с этим представляется особенно важным выделять такие классы задач, в которых малым изменениям исходных данных соответствуют малые изменения конечных результатов.

Постановка задачи. Необходимые и достаточные условия различных типов устойчивости

Рассмотрена задача векторной целочисленной оптимизации следующего вида:

$$D_P(F,X)\colon \max\{F(x) \mid x\in X\},$$

где $F(x)=(f_1(x),\ldots,f_\ell(x))$ – векторный критерий, определенный на множестве $X=D\cap Z^n$; $X\neq\varnothing$; D – замкнутое ограниченное множество из R^n; $D=\{x\in R^n \mid \min\{A_i\, x\le b_i \mid A_i\in H_i\, b_i\in B_i\}\}$, A_i и H_i– соответственно матрица и замкнутое множество в $R^{m\times n}$; Z^n– множество всех целочисленных векторов из R^n; $f_1,\ldots,f_\ell$ – действительные функции. Допустимое множество D *задачи* D_P (F,X) можно представить в следующем виде: $D_i=\bigcup_{i=1}^{m} D_i,\ D_i=\left\{x\in R^n : A_i x\le b_i\right\}$ Обозначим $P(F,X)$ множество всех эффективных (Парето-оптимальных) решений. Очевидно $\forall x\in X$: $x\in P(F,X)\Leftrightarrow\pi(x)=\varnothing$,) $\pi(x)=\{y\in X \mid F(y)\ge F(x), F(y)\neq F(x)\}$.

Продолжая исследования, отраженные в работах [1-3], здесь представляются результаты сравнительного изучения различных подходов к определению понятия устойчивости для векторной задачи целочисленной оптимизации с дизъюнктивныvи линейными функциями ограничений и ограниченным множеством допустимих решений. Рассмотрены пять известных типов устойчивости оптимизационных задач, по-разному описывающих такую ситуацию, при которой малым изменениям входных параметров задачи соответствуют малые изменения выходных результатов. Характеристика каждого типа устойчивости может быть представлена в терминах существования такой окрестности исходных данных задачи в пространстве всех ее возможных исходных данных, что множество оптимальных решений любой возмущенной задачи с исходными данными из этой окрестности обладает некоторым заданным свойством устойчивости по отношению к множеству оптимальных решений первоначальной задачи. Первый тип (T_1) устойчивости описывает ситуацию, при которой множество оптимальных решений задачи имеет непустое пересечение с аналогичным множеством любой из указанных возмущенных задач. Отметим, что этот тип устойчивости присущ всем рассматриваемым здесь задачам поиска оптимальных по Парето и по Слейтеру решений на конечном допустимом множестве. Второй тип (T_2) устойчивости связан с наличием хотя бы одной допустимой точки, принадлежащей множествам оптимальных решений всех задач с исходными данными из указанной выше окрестности. Третий тип (T_3) устойчивости характеризует тот случай, когда достаточно малые возмущения исходных данных задачи не приводят к появлению новых оптимальных решений. Четвертый тип (T_4) устойчивости присущ задачам, для которых все их оптимальные решения не теряют свойства оптимальности при малых изменениях исходных данных. И, наконец, пятый тип (T_5) устойчивости означает, что достаточно малые изменения исходных данных не приводят ни к каким изменениям множества оптимальных решений. Проблема устойчивости многокритериальных задач к возмущениям коэффициентов функций ограничений исследована с точки зрения непосредственно связанных с ней вопросов устойчивости точек некоторых специальных подмножеств допустимой области.

Обозначим $O_\delta(U)$ δ-окрестность множества $U\subset R^n$: $O_\delta(U)=\{x\in R^n \mid \rho(x,U)<\delta\}$.

Для любых $\delta>0$ и $A_i(\delta)\in O_\delta(H_i)$, $b_i(\delta)\in O_\delta(B_i)$ рассмотрим возмущенное допустимое множество $X(\delta)=\{x\in Z^n \mid \min\{A_{ii}(\delta)x\le b_i(\delta) \mid A_i(\delta)\in O_\delta(H_i),\ b_i(\delta)\in O_\delta(B_i)\}\}$. Задачу $D_P(F,X)$ назовем:

T_1-устойчивой по ограничениям, если $\exists\delta>0$: $\forall A_i(\delta)\in O_\delta(H_i)$, $\forall b_i(\delta)\in O_\delta(B_i)$ $P(F,X)\cap P(F,X(\delta))\neq\varnothing$;

T_2-устойчивой по ограничениям, если $\exists\delta>0$ и $\exists x\in P(F,X)$: $\forall A_i(\delta)\in O_\delta(H_i)$, $\forall b_i(\delta)\in O_\delta(B_i)$ $x\in P(F,X(\delta))$;

T_3-устойчивой по ограничениям, если $\exists\delta>0$: $\forall A_i(\delta)\in O_\delta(H_i)$, $\forall b_i(\delta)\in O_\delta(B_i)$ $P(F,X(\delta))\subseteq P(F,X)$;

T_4-устойчивой по ограничениям, если $\exists\delta>0$: $\forall A_i(\delta)\in O_\delta(H_i)$, $\forall b_i(\delta)\in O_\delta(B_i)$ $P(F,X)\subseteq P(F,X(\delta))$;

T_5-устойчивой по ограничениям, если $\exists\delta>0$: $\forall A_i(\delta)\in O_\delta(H_i)$, $\forall b_i(\delta)\in O_\delta(B_i)$ $P(F,X)=P(F,X(\delta))$.

Отметим, что свойство T_3-устойчивости (T_4-,T_5 устойчивости) по ограничениям задачи $D_P(F,X)$ является дискретным аналогом свойства полунепрерывности сверху (соответственно полунепрерывности снизу, непрерывности) по Хаусдорфу точечно-множественного отображения $P(A,b)$: $R^{m\times n}\times R^m\to 2^X$; $(A,b)\to P(F,X(A,b))$.

Рассмотрены необходимые и достаточные условия для всех перечисленных типов устойчивости задачи $D_P(F,X)$ по ограничениям. Справедливы следующие теоремы:

Теорема 1. $P(F,X)\cap intD\neq\varnothing\Leftrightarrow$ задача $D_P(F,X)$ как T_1-, так и T_2-устойчива по ограничениям.

Теорема 2. Если$P(F,X)\subseteq intD$, то задача $D_P(F, X)$ T_3- устойчива по ограничениям.

Теорема 3. Если $P(F,X)\neq X$, $intD\cap Z^n\neq\varnothing$ и все частичные критерии $f_1,\ldots,f_\ell$ – псевдовогнутые функции, то задача $D_P(F,X)$ T_3- устойчива по ограничениям $\Leftrightarrow$ $\pi(x)\cap intD\neq\varnothing$ $\forall x\in X\backslash P(F,X)$.

Теорема 4. Если $intD\cap Z^n=\varnothing$ и все частичные критерии $f_1,\ldots,f_\ell$ – псевдовогнутые функции, то задача $D_P(F,X)$ T_3- устойчива по ограничениям $\Leftrightarrow$ $P(C,X)=X$.

Теорема 5. $P(F,X)\subseteq intD$ $\Leftrightarrow$задача $D_P(F,X)$ как T_4–, так и T_5– устойчива по ограничениям.

Проанализировав полученные результаты, приходим к выводу, что из T_4– устойчивости по ограничениям задачи $D_P(F,X)$ следует ее T_3-устойчивость по ограничениям, а понятия T_1–,T_2–, а такжеT_4– и T_5– устойчивости по ограничениям попарно эквивалентны.

Выводы

Полученные необходимые и достаточные условия разных типов устойчивости по ограничениям свидетельствуют о том, что как бы ни была мала точность приближения исходных данных, множество $P(F,X(\delta))$ в общем случае нельзя считать приближением множества $P(F,X)$. эффективных решений исходной задачи. Это естественно приводит к необходимости создания регуляризующего оператора, представляющего собой определенный вид возмущений исходных данных задачи, для того, чтобы заменить возможно неустойчивую задачу серией заведомо устойчивых эквивалентных задач. Таким образом эффективный процесс постановки и решения рассмотренной векторной задачи дискретной оптимизации предполагает наличие этапа исследования устойчивости этой задачи относительно изменения исходных данных с последующей, если это необходимо, ее регуляризацией, которая позволяет перейти от возможно неустойчивой исходной задачи к заведомо устойчивой к возмущениям входных параметров задаче.

Библиография

[1] Сергиенко И.В., Козерацкая Л.Н., Лебедева Т.Т. Исследование устойчивости и параметрический анализ дискретных оптимизационных задач.– К.: Наук. думка, 1995.

[2] Лебедева Т.Т., Семенова Н.В., Сергиенко Т.И. Устойчивость векторных задач целочисленной оптимизации: взаимосвязь с устойчивостью множеств оптимальных и неоптимальных решений // Кибернетика и сист. анализ. – 2005.– №4.– С.90-100.

[3] Лебедева Т.Т., Сергиенко Т. И. Сравнительный анализ различных типов устойчивости по ограничениям векторной задачи целочисленной оптимизации // Кибернетика и сист. анализ. – 2004. – № 1. – С.63-70.

Информация об авторе

Наталия Владимировна Семенова – *Институт кибернетики им. В.М.Глушкова НАНУкраины 03680 МСП Киев 187, проспект академика Глушкова, 40, Украина; e-mail: nvsemenova@meta.ua*

ПРИМЕНЕНИЕ МОДЕЛИ КОМПЕТЕНЦИЙ ПРИ РЕШЕНИИ ЗАДАЧ УПРАВЛЕНИЯ ПЕРСОНАЛОМ

Юрий В. Бондарчук, Григорий Н. Гнатиенко

***Аннотация**: Предложены решения задач управления персоналом на основе модели компетенций, созданной в компании. Описанные схемы позволяют осуществлять целенаправленный процесс управления трудовыми ресурсами компании на всех этапах ее деятельности.*

***Ключевые слова**: квалиметрия персонала, база компетенций персонала, профиль должности, ранжирование.*

***ACM Classification Keywords**: K.3.2 Computer and Information Science Education.*

Введение

Эффективная деятельность организации традиционно ассоциируется с эффективным исполнением функций управления персоналом. На сегодня нет достоверных методов оценки стоимости персонала, его “веса” в бизнесе. Но в экспертных оценках известных бизнесменов, стоимость персонала в структуре бизнеса успешных компаний равно, а иногда превышает стоимость всех других активов компании. В ряде оценок 20% коммерческого успеха современных компаний зависит от производства, 80% успеха зависит от персонала.

Компетенции персонала есть ключевым понятием концепции управления персоналом. Компетенция есть комплексом профессиональных знаний, навыков, установок и ориентаций. В каждой отдельной профессиональной сфере конкретные компетенции и их комбинации принимают конкретные специфики.

В управленческой практике компаний развитых стран модели профессиональной компетенции рассматриваются как многосторонний инструмент работы с персоналом, ориентированный на достижение стратегии бизнеса. Модель профессиональной компетенции связывает систему управления людскими ресурсами с бизнес-целями организации, а также способствует единству и согласованности в работе всех структурных подразделений.

Сегодня практически все наибольшие зарубежные компании имеют модели компетенций. Причем информация для модели есть строго конфиденциальной. Ибо конкуренты на основе модели компетенций могут получить представление об особенностях жизнедеятельности компании.

Постановка задачи

Пусть в компании создано базу компетенций $A = \{a_1,...,a_n\}$, n – базовое число компетенций, $I = \{1,...,n\}$ – множество индексов компетенций.

При рассмотрении задач, которые возникают на модели компетенций, будем исходить из таких очевидных утверждений:

- менеджерские компетенции имеют прямую связь с бизнес-успехом менеджера;
- компетенции можно измерять в некоторых шкалах;
- каждая компетенция имеет некоторую стоимость, которую можно определить.

Обзор публикаций

Численные масштабные исследования в сфере компетенций [Албастова, 1998; Иванцевич, 1993] показали, что существуют обобщенные блоки бизнес-поведения, которые объединяют в себе группы тесно взаимосвязанных характеристик, что в совокупности формируют корпоративные компетенции.

Выделяются общие для каждого работника базовые компетенции, которые не присущи отдельной профессии или области деятельности и равною мерою способны привести к успешным результатам

независимо от вида деятельности и типа организации. Например, управленческие компетенции, навыки, необходимые для управления проектами, коммуникационные навыки, умение работать в команде, надежность, ответственность, системное мышление, интерес к учебе, самостоятельность, организаторские способности, предвидение, гибкость, ориентация к достижениям, принятие решений и т.п. Базовые компетенции есть решающим инструментом профессионального успеха. Они воспринимаются как набор умений, квалификаций, обладанием запасом знаний и умений, необходимых для успешного взаимодействия в современном мире. Всего исследователями установлено более 20 общих корпоративных компетенций.

Как правило, используется пять степеней выражения компетенций: негативный (дефицитный) уровень, уровень понимания, базовый уровень, сильный (высокий) уровень, лидерский уровень. Особенности выделения названных уровней связаны с ранжированием соответствующих уровней усвоения знаний: <понимание> $\prec$ <осмысление> $\prec$ <воспроизведение> $\prec$ <применение> $\prec$ <трансформация>.

Планирование потребности в персонале

Управление персоналом, как правило, имеет три основных фактора: создание условий деятельности, создание и развитие межличностных и профессиональных взаимодействий, развитие творческого потенциала работников организации с учетом возможных стимулов. Понятие компетенций эффективно применяется на всех этапах управления персоналом.

При определении потребности в персонале анализируются планы деятельности компании, существующие тарифы и нормы, прогнозы уровня механизации трудовой деятельности и т.п. При планировании количества работников компании перспективным есть использование аппарат нечетких множеств.

Пусть X – универсальное множество k-мерных альтернатив, которое представляет все возможные варианты k плановых показателей компании. Y – универсальное множество m-мерных альтернатив, которое представляет все возможные варианты уровней механизации (автоматизации) m структурных подразделений компании. Z – универсальное множество m-мерных альтернатив, которое представляет все возможные варианты количества работников в m подразделений компании.

A – нечеткое множество планов производства, т.е. совокупность пар вида $(x, \mu(x))$, где $x \in X$, а μ – функция принадлежности нечеткого множества A: $X \to [0,1]$. B – нечеткое множество уровней механизации (автоматизации) на разных участках производства, т.е. совокупность пар вида $(y, \mu(y))$, где $y \in Y$, а μ – функция принадлежности нечеткого множества B: $Y \to [0,1]$. C – нечеткое множество количества работников в каждом из m структурных подразделений компании, т.е. совокупность пар вида $(z, \mu(z))$, де $z \in Z$, а μ – функция принадлежности нечеткого множества C: $Z \to [0,1]$.

С учетом анализа проектов, которые предусмотрено реализовать, можно определить “интегральные компетенции проектов”, и оптимизировать количество персонала, который сможет реализовать эти проекты.

В ходе планирования потребности в персонале логично определять не только количество персонала, но и анализировать наборы компетенций, которыми должны владеть разные категории работников. Расчет потребности в управленском персонале по всем функциям, которые реализуются на производстве специалистами разных категорий, ведется на основе моделей и формул, которые разрабатываются при планировании численности разных категорий персонала компании с помощью экономико-математических зависимостей.

Для расчета потребности в персонале применяется такие показатели:

- расчетное количество управленского персонала по категориям
- среднегодовое количество производственного персонала
- количество основных работников
- количество работников с подрядной формой оплаты труда
- стоимость активной части основных производственных фондов и т.д.

Разработка требований к персоналу

Для каждой позиции имеется своя иерархия компетенций. Одной из основных задач управления персоналом есть формирование “портрета” специалиста. Для отбора и группирования качеств, которыми должен обладать специалист на конкретной должности, применяется метод векторного анализа.

Профиль должности, полученный в результате исследования компетенций, необходимый для успешного выполнения функций, будем обозначать $\varphi_i^{\Pi} = (a_{i_1}^{\Pi},...,a_{i_{si}}^{\Pi}),\ i = 1,...,\lambda,$ где λ – количество должностей в компании, для которых определяются профили, s_i – количество компетенций для i –й должности.

Профиль конкретной должности $\varphi_t^{\Pi},\ \ t \in \{1,...,\lambda\},$ формируется в результате анализа мнения руководителя соответствующего подразделения, сравнения профилей должностей, с которыми взаимодействует специалист, который будет занимать эту должность, и на основе сравнения соответствующих должностных инструкций.

Подбор и отбор кандидатов на вакантные должности

Данные о кандидате на должность сопоставляются с профилем той позиции, на которую он претендует. После полученных общих данных о кандидате, психодиагностике черт личности, изучения его структуры интеллекта и способностей осуществляется оценка качеств кандидата. Эта информация формализуется в виде профиля кандидата на должность: $\varphi_i^{K} = (a_{i_1}^{K},...,a_{i_{si}}^{K}).$

Следующей задачей есть определение меры (процента или степени) соответствия кандидата вакантной должности. Для этого вводится мера схожести профилей $\xi = \xi(\varphi^{\Pi},\varphi^{K}).$ Считается, что профили кандидата и должности абсолютно совпадают, когда значение $\xi = 1.$ Меры схожести профилей можно измерять с использованием подходов, предложенных в [Гнатиенко, 1997].

Разработка системы подбора персонала состоит в формировании резерва кадров на замещение вакантных рабочих мест. Система подбора персонала должна содержать процедуры расчета потребности в персонале, нормативное описание профессиональных требований к работникам, моделировании рабочих мест, способы профессионального подбора кадров, а также принципы формирования кадрового резерва на вакантные должности.

Определение веса компетенции в структуре заработной платы работника

Для решения этой задачи могут быть применены различные методы определения весовых коефициентов, которые описаны, например, в работах [5-11].

Отношение предпочтения между профилями должностей позволяет решить задачу определения стоимости отдельной компетенции. Для этого перспективным есть применения функций принадлежности нечеткого множества.

В результате применения указанных методов определяются нормированные весовые коэффициенты $\rho_s, s \in I,$ влиянии компетенции на важность должности в компании, $\rho_s > 0, s \in I,\ \sum_{s \in I} \rho_s = 1.$

Адаптация персонала на начальных этапах работы

Формализация процедуры адаптации есть важной частью повышения качества компании в целом. Совместной целью нового работника, его непосредственного руководителя, службы управления людскими ресурсами и компании в целом, есть рост меры схожести $\xi = \xi(\varphi^{\Pi},\varphi^{K})$ профиля нового работника и профиля должности, на которую его принято. В случае, когда выделены компетенции, необходимые для успешного выполнения обязанностей на каждой должности, служба управления персоналом осуществляет целенаправленное повышение величины коэфициента $\xi = \xi(\varphi^{\Pi},\varphi^{K})$ вместе с руководством структурного подразделения.

На основании комплексного анализа процедур адаптации персонала служба управления людскими ресурсами осуществляет планирование индивидуальных и корпоративных тренингов, курсов повышения квалификации, организацию циклов лекций и т.п.

Ранжирование структурных подразделений

Понятно, что жизнедеятельность компании реализуется всеми структурными подразделениями. Но можно определить важность выполняемых функций и таким образом их относительный условный вес подразделений, которые эти функции реализуют. Ранжирование, как структурных подразделений, так и должностей в структурных подразделах можно осуществить на основании анализа компетенций, необходимых для каждой должности, а также путем анализа экспертных ранжирований, полученных от топ-менеджеров. Последняя задача может быть формализована таким образом.

На множестве n структурных подразделений a^ν, $a^\nu \in A$, $\nu \in L=\{1,...,n\}$, каждым экспертом из группы k топ-менеджеров задано строгое ранжирование подразделений. Т.е. руководители линейно упорядочивают множество подразделений, приписывая каждому структурному подразделению место в последовательности подразделений компании.

Наиболее обоснованным методом поиска результирующего ранжирования объектов считается расчет медианы заданных ранжирований:

$$R^* \in \operatorname{Arg}\min_{R\in\mathfrak{R}} \sum_{i\in I} d(R,R^i), \qquad (1)$$

где $d(R,R^i)$ - "расстояние" между ранжированиями объектов $R\in\mathfrak{R}$ и $R^i\in\mathfrak{R}$, $i\in I$; $\mathfrak{R}$ - множество всех возможных строгих ранжирований объектов. Ранжирование объектов R^*, найденное в виде (1), называется медианой Кемени-Снелла. Для класса ранжирований задача (1), в которой используется метрика Хеминга, впервые сформулированная Кемени и Снеллом в работе [Кемени, 1972].

Если используется метрика разности рангов объектов, то результирующее ранжирование называется медианой Кука-Сейфорда [Cook. 1982].

В [Макаров, 1982] приводится также алгоритм поиска результирующего ранжирования в виде среднего

$$R^* \in \operatorname{Arg}\min_{R\in\mathfrak{R}} \sum_{i\in I} d^2(R,R^i).$$

В работе [Волошин, 1990] введено и обосновано постановку задачи поиска компромиссного ранжирования объектов в виде

$$R^* \in \operatorname{Arg}\min_{R\in\mathfrak{R}} \max_{i\in I} d(R,R^i). \qquad (2)$$

Решение задачи (2) для случая, когда расстояние между ранжированиями измеряются метрикой Хеминга, получило название ВГ-медиана. При использовании метрики разности рангов решение задачи вида (2) называется ГВ-медианой [Гнатиенко, 2001].

Анализ и определение стоимости должностей

Создание единой системы материальной мотивации играет важную роль на всех этапах роботы с персоналом и во всех сферах деятельности компании. Системность в определении стоимости должностей способствует экономии всех ресурсов компании: финансовых, производственных, дистрибьюторских, людских. Определение зависимости между весом должности и вознаграждением есть базисом для определения внутренней ценности и конкурентности компании.

Аналитическая коллективная оценка (калибрование) должностей, как правило, осуществляется по методике известных разработок Едварда Хея (Hay Group) или Ватсон Вейт (Watson Wyatt), которые обеспечивают сопоставимость должностных позиций по единым группам факторов.

Существенным этапом есть также анализ вознаграждений и дополнительных компенсаций для каждой конкретной должности. Оценка веса должностей позволяет сравнивать их на функциональном уровне, секторе рынка, регионе. Традиционными секторами рынка есть: товары ширпотреба, телекоммуникации, розничная торговля, страховой сектор, производство, энергетический сектор, банковский сектор, фармацевтический сектор, транспортные средства, издательский сектор, сети автозаправочных станций.

Определение необходимости в повышении квалификации работника

В компании, что внедрила модель компетенций, осуществляется анализ качеств работников, которые развиты достаточной мерою, и тех, которые требуют развития. Т.е. выявляются те компетенции $a_i^K \in \varphi_i^K$, уровень которых есть недостаточным для полноценного выполнения функций $i-м$ работником. Профиль кандидата на должность $\varphi_i^K, i \in I$, такой, что $\varphi_i^П \succ \varphi_i^K$, требует от службы управления персоналом действий, которые содействуют повышению уровня компетенции работника.

Регулярное измерение уровня компетенций персонала дает возможность службе управления персоналом реализовывать взвешенную, целенаправленную политику обучения и развития персонала.

Планирование карьеры

Основная идея модели профессиональной компетенции: положение про изменчивость корпоративных компетенций. Причем деструктивные последствия для организации имеют не только недостаточная компетенция, т.е. дефицит профессиональности (случай $\varphi^K \prec \varphi^П$), но и завышенная компетенция (случай $\varphi^K \succ \varphi^П$). На основании выявленного недостаточного уровня отдельных компетенций осуществляется формирование групп для проведения тренингов и обучения. Поэтому по результатам анализа компетенций работника можно вместе с развитием его компетенций $a_i^K \in \varphi_i^K$, $i \in I$, планировать и его карьеру.

Определение результативности обучения, тренингов, семинаров

Для каждой должности в компании можно расписать алгоритм достижения заданного уровня развития компетенций. Например, компетенция “влияние” складывается из таких качеств: внимательности, умения быть настойчивым, умения говорить весомо, умения обосновывать идеи, умения активно слушать, умения заручиться поддержкой, искусства переговоров и т.д.

Результативность обучения, тренингов, семинаров можно измерять разными методами: анализом результатов деятельности подразделений и отдельных работников до и после тренингов, оценкой аудиторией качества тренинга, измерением изменения уровня компетенций $a_i^K \in \varphi_i^K$, $i \in I$, которые планировалось увеличить в процессе проведения обучения. При измерении изменений уровня конкретной компетенции можно определить цену градации компетенции как стоимость обучения и экспертную оценку труда работника.

Реализация оценки результатов работы

Анализ эффективности деятельности персонала можно измерять в двух направлениях: оценивание управленческих качеств менеджеров и реализовывать оценку выполнения работниками должностных обязанностей. Причем оценка управленческих качеств менеджеров дает возможность оценивать стратегический потенциал компании, увидеть положение компании в перспективе, политику обучения и развития персонала. Для оценки результатов работы следует построить квалиметрическую модель оценки сложности труда специалистов и оценку трудовых затрат работника. Стоится также математическая модель оценки результатов работы на основе учета полезности затрат рабочего времени. После этого осуществляется оперативная оценка эффективности деятельности отдельных работников, подразделений и компании в целом с точки зрения достижения ими бизнес-целей за некоторый период. Осуществляется также построение единых рейтингов подразделений и работников с точки зрения эффективности их работы. Таким образом, создается объективная основа для универсальной системы премирования.

Плановые аттестации персонала

Одной из важнейших функций службы управления персоналом есть организация адекватной оценки соответствия работника требованиям занимаемой им должности. Эта оценка осуществляется на всех этапах деятельности компании.

Квалиметрическая оценка индивидуальных качеств работников (S) имеет место во всех системах аттестации. Как правило, выделяется две группы индивидуальных качеств, которые характеризуют моральные (μ) и физические (ψ) качества работников. Оценка индивидуальных качеств может быть представлена в виде функции $S = S(\mu, \psi)$.

На сегодня психологами выявлено более 150 качеств, которые отображают моральность. Выделяют пять интегральных факторов моральности: f_1 - доброта, f_2 - надежность, f_3 - воспитанность, f_4 - отношение к работе, f_5 - сознание. Квалиметрический подход к формированию модели оценки моральных качеств предусматривает обнаружение весомости каждого фактора $f_i, i = 1,...,5.$ Квалиметрическая факторно-критериальная модель позволяет определить уровень морального развития работника на основании анализа близости значений $f_i, i = 1,...,5,$ к идеальным.

Аттестационная политика компании может быть выражена в виде алгоритмов измерения деятельности структурных подразделений и каждого работника отдельно. Во время аттестации выявляются все компетенции персонала и их уровень, так как осуществляется выборочное измерение важнейших для выполнения функций компетенций. Следующим этапом есть оценка трудового потенциала персонала и определение индивидуального размера оплаты труда по результатам аттестации.

Формирования кадрового резерва

Практика показывает, что основной причиной ошибок менеджеров есть не недостаток знаний, а отсутствие навыков. Большинство менеджеров знает как правильно работать, но необходимым есть не увеличение знаний, а обретение навыков на практике.

При решении задач работы с кадровым резервом осуществляется формирование множества работников, которые есть потенциально перспективными с точки зрения их карьерного продвижения.

Поэтому процедура формирования кадрового резерва может быть выражена такой последовательности шагов: <сопоставление профилей должностей и профилей кандидатов> $\rightarrow$ <определение отличий> $\rightarrow$ <мониторинг стоимости обучения> $\rightarrow$ <подготовка комплексного плана обучения>.

Мониторинг рынка труда и заработной платы

На основании статистических данных, полученных с разных источников, осуществляется анализ стоимости должностей. Результаты анализа могут быть представлены, например, в виде функций принадлежности. На основании этих данных определяется заработная плата работников с использованием аппарата нечетких множеств.

Мотивация персонала

Технология внутренней и внешней гармонизации окладов компании, как правило, базируется на классическом методе бальной оценки. Для этого на первом этапе следует построить внутрифирменную шкалу окладов (разрядов, грейдов (grades)). Дальше осуществляется выбор размеров вилок окладов или строятся функции принадлежности материальных вознаграждений для каждой должности. Следующим этапом есть приведения оплаты в компании в соответствие к рыночному уровню оплаты в зависимости от стратегического позициирования компании на рынке труда. На основании анализа внутренних и внешних влияний определяется объективный размер оплаты для каждой новой должности в компании.

Ротация персонала

Регулярный мониторинг уровня компетенций работников компании дает возможность целенаправленно осуществлять служебно-профессиональное перемещение работников. Например, может получаться, что $\xi(\varphi_i^{\Pi}, \varphi^{K}) > \xi(\varphi_j^{\Pi}, \varphi^{K})$, т.е. $i-й$ работник есть потенциально перспективнее для работы на должности с профилем φ^{K} от $j-го$ работника.

Основными факторами для разработки проектов расстановки кадров и их дальнейшая ротация есть: модели служебной карьеры, философия организации, кодекс законов об труде, материалы аттестационной комиссии, трудовой договор с работником, штатное расписание, служебная инструкция, личное дело работника, положение об оплате труда, положение о расстановке кадров. В результате реализации этого проекта все вакантные рабочие места в компании могут быть заняты с учетом всех важных факторов, в том числе и с учетом индивидуальных предпочтений работников и их плановой карьеры.

Прогноз потенциальных конфликтов

Психодиагностика межличностных отношений, которая регулярно измеряется службой управления людскими ресурсами, дает возможность измерять социально-психологический климат в коллективе и взаимоотношения между разными группами интересов работников. При этом важную роль играет модель компетенций.

Выводы

Рассмотрены основные задачи, которые возникают в компании на всех этапах управления персоналом. Предложены схемы решения этих задач на основе модели компетенций, созданной в компании. Приведенные способы управления персоналом могут быть алгоритмизованы и внедрены в виде программного обеспечения в любой крупной компании.

Библиография

1. [Албастова. 1998] Албастова Л.Н. Технологии эффективного менеджмента.–М.:ПРИОР, 1998.– 288с.
2. [Иванцевич, 1993] Иванцевич Дж., Лобанів А.А. Человеческие ресурсы управления. М.: Дело, 1993. – 304с.
3. [Гнатієнко, 1997] Гнатієнко Г.М., Єпик Н.Б. Про визначення міри схожості вподобань експертів //Вісн.Київ.ун-ту. Фіз.-мат. науки.-Київ, 1997, №3, С.159-165.
4. [Гнатієнко, 2001а] Гнатієнко Г.М. Визначення міри схожості експертних розподілів об'єктів за належністю до кластерів //Вісн. Київ.ун-ту. Фіз.-мат. науки.-Київ, 2001, №3, С.220-223
5. [Гнатієнко, 1990] Гнатиенко Г.Н. Задание предпочтений на множестве критериальных функций в задачах многокритериальной оптимизации //Вестн. Киев. ун-та. Моделирование и оптимизация слож. систем. 1990.-Вып.9-С.87-92.
6. [Гнатієнко, 2000а] Гнатієнко Г.М. Деякі математичні аспекти соціальної експертизи //Соціальна експертиза в Україні: методологія, методика, досвід впровадження/За ред. Ю.І.Саєнка.-К.: Ін-т соціології НАНУ, 2000. – 194с.
7. [Гнатієнко, Дробот, 2000] Гнатієнко Г.М., Дробот О.В. Використання вагових коефіцієнтів у моделюванні процесів спільного інвестування //Наукові праці Кірово-градського державного університету. Економічні науки.-Вип.8.-Кіровоград: КДТУ, 2000.-С.125-132.
8. [Михалевич, 1982] Михалевич В.С., Волкович В.Л. Вычислительные методы исследования и проектирования сложных систем. – М.: Наука, 1982. - 286 с.
9. [Гнатієнко, 2000б] Гнатієнко Г.М. Метод адаптивного визначення інтервалів відносної важливості параметрів багатовимірних об'єктів //Вісн.Київ.ун-ту. Фіз.-мат. науки.-Київ, 2000, Вип.№1, С.215-221.
10. [Михалевичб 1981] Михалевич В.С., Волкович В.Л., Волошин А.Ф. Метод последовательного анализа в задачах линейного програм-мирования большого размера //Кибернетика.–1981.–№4.–С.114-120.
11. [Волошин, 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. – С. 34-42.
12. [Кемени, 1972] Кемени Дж.Г., Снелл Дж.Л. Кибернетическое моделирование. М.: Советское радио. 1972. 192с.

13. [Кук, 1982] Cook W.D., Seiford L.M. On the Borda-Kendall Consensus Method for Priority Ranking Problems //Management Science. 1982. Vol.28, №6. P.621-637.
14. [Макаров, 1982] Макаров И.М., Виноградская Т.М., Рубчинский А.А. и др. Теория выбора и принятия решений: Учебное пособие. М.:Наука, 1982.-328с.
15. [Волошин, 1990] Волошин А.Ф., Гнатиенко Г.Н. Построение компромиссной ранжировки в задаче группового выбора //Проблемы теоретической кибернетики: Тез. 8-й Всесоюз. конф., ч.2.-Волгоград, 1990, С.44-46.
16. [Гнатієнко, 2001б] Гнатієнко Г.М. Послідовні алгоритми знаходження строгого компромісного упорядкування об'єктів у задачах ранжування //Вісн.Київ.ун-ту. Фіз.-мат. науки.-Київ, 2001, №4, С.218-225.

Информация об авторах

Юрий В. Бондарчук *– Националный университет им. Т.Шевченка, факультет кибернетики, доцент, Киев, Украина, e-mail: byv@univ.kiev.ua*

Григорий Н. Гнатиенко *– Компания «Верес», директор по персоналу, Киев, Украина, e-mail: g.gnatienko@veres.com.ua*

НЕЧЕТКИЕ МНОЖЕСТВА: КЛАССИФИКАЦИЯ СИТУАЦИЙ

Владимир Донченко

Аннотация*: Для модифицированного варианта определения нечёткого множества рассматривается задача классификации ситуации, определяемой набором нечётких множеств. Предлагается алгоритм классификации, являющийся аналогом ММП (метода максимального правдоподобия). Рассматривается модель реализации набора нечётких множеств на основе «зондирующих множеств». Исследуются свойства предложенной модели.*

Ключевые слова*: нечёткое подмножество, функция принадлежности, классификация, кластеризация.*

Введение

Концепция нечётких множеств (в дальнейшем – НМ), предложенная Лотфи Заде [Zadeh L. 1965] (см. систематическое изложение в монографии [Кофман А. 1982]), была попыткой удовлетворить потребности практики в расширении возможностей описания неопределённости, возникающей в реальных задачах. Такая потребность характеризовала как описание условий, так и представление результатов в форме, которая бы учитывала наличие «пограничных», «переходных» областей для возможных "crisp" – альтернатив («чётких» альтернатив). Предложенная теория с самого начала позиционировалась с одной стороны как теория альтернативная или обобщающая классическую теорию множеств, с другой – как альтернативная, не имеющая ничего общего со статистическим (теоретико вероятностным) подходом к описанию неопределённости. И та и другая претензии являются, по-видимому, симптомами становления. Действительно, в части теоретико множественной НМ – теория находится скорее в фазе «наивности»: ни о какой аксиоматике речь не идёт, в [Донченко В.С. 2005] делается попытка реализации представлений об аксиоме абстракции, но, скорее, – в «наивной» постановке), отсутствуют какие либо конструкции, которые можно было бы назвать «нечёткой логикой» в смысле действительных логических исчислений, а не того что принято называть (см., например [Кофман А. 1982]) таковой. Действительно, то, что называют нечёткой логикой сейчас, является аналогом "crisp"- исчисления высказываний, хотя и в этом варианте отсутствует полнота основных операций, характерная для классического "crisp"- исчисления, т.е исчисления в булевском варианте..

Претензия НМ теории на альтернативность статистическому методу, под которым, собственно, понимают исследование объектов на основе частот (в том числе и предельных – вероятностей) результатов, которыми эти объекты представлены в наблюдениях, то, как представляется, такое «самоограничение» является неплодотворным для самой НМ – теории, поскольку таким образом отсекается мощная

методология исследования и интерпретации результатов. Тем более, что, как показано в работе автора [Донченко В.С. 2005], статистическая интерпретация функции принадлежности достаточно естественна. Рассмотрение статистической интерпретации в уже упомянутой работе автора с одной стороны – указало на отсутствие в определении объекта нечёткости в классической НМ – теории, а с другой – указало на способ разрешения указанной коллизии. Такое разрешение формально нашло своё воплощение в концепции модифицированного нечёткого множества (МНМ) и состоит (цитированная выше работа автора), в явном введении объекта нечёткого описания: «чёткого» – crisp – объекта в классическое определение нечёткого множества. Таким crisp – объекта нечёткости является crisp – предикат или,соответственно, – crisp – множество, которые и будут характеризоваться нечётко с помощью функции принадлежности. Реализация Формально введение объекта нечёткого описания: объекта нечёткости – осуществляется введением дополнительного аргумента: в дальнейшем для удобства crisp - предиката P – в функцию принадлежности. Функция принадлежности становится зависимой от двух аргументов: элемента $e \in E$ носителя и crisp - предиката P как параметра.

Определение. (МНМ – Модифицированное Нечёткое Множество). Нечетким подмножеством множества E в модифицированном варианте(МНМ), которое нечетко описывает crisp - свойство P из $\wp$ – или соответствующее ему множество U_P, – называется пара (E, $\mu^{(P)}(e)$) или пара (E, $\mu^{(U_P)}(e)$), где:

- E - абстрактное множество, которые будем называть носителем нечеткого подмножества;
- P – предикат из множества $\wp$ на некотором универсальном для предикатов множестве а U_P – подмножество множества E, которое отвечает предикату P;
- $\mu^{(P)}(e)$ – функция двух аргументов: e∈E и $P \in \wp$; эту функцию, как и в классической теории нечетких подмножеств, будем называть функцией принадлежности, прибавляя, что она нечетко реализует или характеризует свойство P или соответствующее множество U_P.

В заключение отметим, что очевидным примером вероятностной интерпретация нечетких подмножеств и примером МНМ являются обобщенные варианты логит- и пробит- регрессий, в которых, как известно, вероятность появления определённого события зависит от количественных характеристик, наблюдаемых вместе с появлением или не появлением события в эксперименте. Ниже появляется другой естественный пример МНМ, построенных на основы «зондирования» распределений вероятностей стандартными множествами, например: вычисления их значений на шарах фиксированного радиуса в R^m.

Нечёткие множества в модифицированном варианте (МНМ -множества)

Модифицированное определение НМ - множества (в дальнейшем МНМ) как пары $(E, \mu^{(P)}(\cdot)), \mu : E \to [0,1]$ с функцией принадлежности, являющейся функцией двух аргументов: $e \in E$ и предиката $P \in \wp$, – определённой на одном и том же носителе, – позволяет ставить задачу классификации или кластеризации: отнесения элемента $e \in E$ к одному из К классов, определяемых предикатами $P_k \in \wp, k = \overline{1,K}$ набора МНМ $(E, \mu_k^{(P_k)}(\cdot)), P_k \in \wp, k = \overline{1,K}$. Следует отметить, что набор МНМ может быть как полным:

$$\forall e \in E \sum_{e \in E} \mu_k^{(P_k)}(e) = 1, \tag{1}$$

так и необязательно полным.

Набор предикатов $P_k \in \wp, k = \overline{1,K}$ набора МНМ можно интерпретировать как набор альтернатив – не обязательно взаимно исключающих, – которые могут быть реализованы для того или иного значения $e \in E$ с вероятностями, которые задаются значениями соответствующих функций принадлежности Н $\mu_k^{(P_k)}(e), k = \overline{1,K}, e \in E$.

Значения функций принадлежности могут рассматриваться и в классическом варианте: как степень уверенности. Правда, в этом случае необходимо добавить список объектов, которые характеризуются через степень уверенности, – для каждой функции принадлежности свой.

Будем говорить, что набор МНМ $(E,\mu_k^{(P_k)}(\cdot)),P_k \in \wp,k=\overline{1,K}$,описывает ситуацию для элементов $e \in E$ или ситуацию, конкретизируемую элементом $e \in E$. Отнесение исследуемого элемента $e \in E$ к одному из К классов, определимых предикатами $P_k \in \wp,k=\overline{1,K}$ будем называть классификацией ситуации для $e \in E$ или ситуации, конкретизируемой этим элементом.

Оценка ситуации

Интерпретации набора МНМ – множеств как ситуации для элементов $e \in E$, в которой $e \in E$ конкретизирует ситуацию, а набор $\mu_k^{(P_k)}(e),k=\overline{1,K}$ для фиксированного $e \in E$ описывает „степень проявления" свойств $P_k \in \wp,k=\overline{1,K}$, собственно, – вероятностей вариантов её развития, – позволяет „оценивать" вариант наиболее вероятного её развития: в соответствии с наиболее вероятным вариантом её реализации. Такой подход к построению «оценки», собственно, является вариантом идеи, реализованной в методе максимального правдоподобия (ММП), только апостериорно: при наличии серии наблюдений.

Определение. Функцию $\hat{P}(e),e \in E,\hat{P}:E \to \{P_1,...,P_K\}$, определённую по набору МНМ $\mu_k^{(P_k)}(e),k=\overline{1,K}$ соотношением

$$\hat{P}(e)=P_{k*},k*=\arg\max_{k=\overline{1,K}}\mu_k^{(P_k)}(e),e \in E \tag{2}$$

будем называть оценкой развития ситуации для $e \in E$ по максимуму функции принадлежности или просто оценкой развития ситуации для элемента $e \in E$.

Замечание 1. Вообще говоря, оценка развития ситуации может иметь множественный характер, если для того или иного $e \in E$ максимальное значение функций набора $\mu_k^{(P_k)}(e),k=\overline{1,K}$, достигается одновременно для нескольких номеров $k=\overline{1,K}$. В этом случае значение $\hat{P}(e),e \in E$ приобретают множественный характер: $\hat{P}(e) \subseteq \{P_{1,},...,P_K\},e \in E$ – и определяются в соответствии с модифицированным вариантом:

$$\hat{P}(e)=\{P_k:k \in K*=\mathrm{Arg}\max_{k=\overline{1,K}}\mu_k^{(P_k)}(e),e \in E\}. \tag{3}$$

Замечание 1. Термин «оценка развития ситуации» для функции $\hat{P}(e),e \in E$, никоим образом не ограничивает „классификационного" характера этого объекта в случае, когда набор МНМ задаёт вероятности отнесения элементов $e \in E$ к одному из классов альтернативного набора $\{P_1,...,P_K\}$, не обязательно взаимно исключающих. В этом случае оценку $\hat{P}(e),e \in E$ будем называть также классифицирующей функцией. При альтернативах, исключающих друг друга, естественным является условие полноты МНМ: выполнение условия (1).

Замечание 2. Оценка ситуации в соответствии с (2) или (3), которая использует одну из операций нечёткой логики: в данном случае максимум– демонстрирует, что в случае применения к функциям принадлежности уместным является использования не результата операции, а рассмотрения тех объектов, для которых максимум при нечётком описании достигается для тех или иных конкретизирующих значений $e \in E$.

Кластеризация значений для нескольких распределений вероятностей, множества-зонды

Примером классификации ситуаций в смысле, рассмотренном выше, является задача кластеризации значений $e \in R^m$, которые могут быть представлять значения одной из К случайных величин (с.в.) со значениями R^m. Вариантом функций принадлежности МНМ в такой классической статистической задаче может быть такой, в котором реализуется концепция «зондирования» определённым множеством

распределений, которые фигурируют в задаче. Такими множествами – зондами могут быть сдвиги $e+\pi$, $e \in E$ фиксированного множества $\pi \subseteq R^m$, включающего начало координат. К примеру, ниже в качестве множеств-зондов используются всевозможные замкнутые шары $S_\rho(e) = e + S_\rho(0)$ радиуса $\rho > 0$ с центром в $e \in R^m$ или множества $V_\rho(e) = e + \rho V$ с симметричным замкнутым множеством V единичного радиуса, включающем некоторый шар ненулевого радиуса с центром в нуле.

Действительно, пусть в R^m могут наблюдаться $e \in R^m$, которые являются значением одной из K в.в. $\varepsilon_k, k = \overline{1,K}$ из R^m, каждая со своим, известным распределением. Известным распределением $P^{(k)}(B) = P\{\varepsilon_k \in B\}, k = \overline{1,K}$ на борелевских множествах B из R^m.

Определим набор МНМ $(R^m, \mu_{k,\rho}^{(P_k)}(\cdot))$, задав $\mu_{k,\rho}^{(P_k)}(\cdot), e \in E$ в соответствии со следующим соотношением:

$$\mu_{k,\rho}^{(P_k)}(e) = P^{(k)}(S_\rho(e)) = P^{(k)}(e + S_\rho(0)) = P\{\varepsilon_k \in e + S_\rho(0)\},\ k = \overline{1,K}, e \in R^m. \quad (4)$$

Очевидным образом, каждая из функций $\mu_{k,\rho}^{(P_k)}(e), e \in R^m, \rho > 0, k = \overline{1,K}$ принимает значения из интервала $[0,1]$, т.е. является функцией принадлежности. Crisp-предикат P_k в её определении описывается как свойство «иметь распределение $P^{(k)}$», $k = \overline{1,K}$. Таким образом набор $(R^m, \mu_{k,\rho}^{(P_k)}(\cdot)), k = \overline{1,K}$, определённый в (4), является набором МНМ.

Шары $S_\rho(e) = e + S_\rho(0)$ радиуса $\rho > 0$ с центром в точке $e \in R^m$ естественным образом «зондируют» имеющиеся распределения на предмет их значений. Результаты зондирования представлены соответствующими функциями принадлежности

Уместно заметить, что функции принадлежности набора МНМ в рассматриваемом варианте имеют очевидное статистическое: теоретико вероятностное значение. Вид стандартной статистической интерпретации для функции принадлежности классической НМ они приобретают, если вероятности, определяющие функции принадлежности, представить в виде:

$$\mu_{k,\rho}^{(P_k)}(e)) = P\{\varepsilon_k \in S_\rho(\xi) \mid \xi = e\}, k = \overline{1,K} \quad (5)$$

с с.в. ξ со значениями в R^m, определённой на одном вероятностном пространстве с, $\varepsilon_k, k = \overline{1,K}$, независимая от них с нетривиальным распределением в R^m.

Действительно,

$$P\{\varepsilon_k \in S_\rho(\xi) \mid \xi\} = M\{\chi_{S_\rho(\xi)}(\varepsilon_k) \mid \xi\} = M\{\chi_{S_\rho(0)}(\varepsilon_k - \xi) \mid \xi\} =$$

$$= M\{\chi_{S_\rho(0)}(\varepsilon_k - e)\Big|_{e=\xi} = P\{\varepsilon_k - e \in S_\rho(0)\}\Big|_{e=\xi} = P\{\varepsilon_k \in S_\rho(e)\}\Big|_{e=\xi}.$$

Таким чином, (5) является примером стандартного представления классической функции принадлежности в соответствии с универсальной статистической интерпретацией, обсуждавшейся в [Донченко В.С.2005].

Возвращаясь к задаче оценки ситуаций в рассматриваемой задаче классификации значений набора распределений, заметим, что в исследуемом случае в соответствии с (2) или (3) она – будем обозначать её $\hat{P}_\rho(e), e \in R^m$ – определяется одним из двух соотношений::

$$\hat{P}_\rho(e) = P_{k*} : k* = \arg\max_{k=\overline{1,K}} \mu_{k,\rho}^{(P_k)}(e), e \in R^m\}, \quad (6)$$

$$\hat{P}_\rho(e) = \{P_k : k \in K* = \operatorname{Arg}\max_{k=\overline{1,K}} \mu_{k,\rho}^{(P_k)}(e), e \in R^m\}. \quad (7)$$

Оценка ситуации в соответствии (6) или (7) очевидным образом является оценкой по максимуму вероятности появления ρ- окрестности $e \in R^m$. Заметим, что рассматриваемые альтернативы не являются исключающими.

Предельный переход по геометрическим размерам зондов в задаче кластеризация

В предыдущем пункте задача классификации ситуаций имеет в соответствии с (6) или (7): отнесение к тому или иному по максимуму функции принадлежности – имеет прямой статистический (теоретико-вероятностный) смысл классификации по максимуму соответствующей вероятности. Ниже обсуждаются вопросы, связанные с представлением задачи классификации ситуации при уменьшающихся геометрических размерах множеств – зондов. Естественным образом функции принадлежности подвергаются подходящей нормировке. Показывается, что в таких условиях задача классификации ситуации в соответствии с (6) или (7) сводится к классификации по максимуму значений плотностей распределений, для которых рассматривается задача.

Действительно, пусть распределения, отвечающие $\varepsilon_k, k = \overline{1,K}$, имеют непрерывные плотности $h_k(z), z \in R^m, k = \overline{1,K}$, а размеры ρ множеств - зондов $S_\rho(e) = e + S_\rho(0)$ неограниченно уменьшаются: $\rho \to 0$. Как уже отмечалось, для исследования предела функций принадлежности из набора МНМ, построенного в соответствии с (4), необходимо осуществить подходящую нормировку. Таких подходящих вариантов нормировки для $e \in R^1$ два: с самим радиусом $\rho > 0$ в одном варианте, и с лебеговской мерой ϑ в R^1 шара, собственно, интервала – $S_\rho(e) = e + S_\rho(0), e \in R^1$ – в другом. В общем случае: для $e \in R^m$ нормировка единственная м совпадает с лебеговской мерой ϑ в R^m шара, $S_\rho(e) = e + S_\rho(0), R^m$.Естественно, что в случае нормировки лебеговской мерой, распределения $\varepsilon_k, k = \overline{1,K}$ должны быть нетривиальными в том смысле, что соответствующие вероятности для всех шаров $S_\rho(e) = e + S_\rho(0), e \in R^m$, должны иметь ненулевые значения. Это условие, в частности, выполняется, для для распределений, задаваемых плотностями $h_k(z), z \in R^m, k = \overline{1,K}$, которые определённы на всём R^m, почти наверное не равны нулю и не имеют областей постоянства ненулевой лебеговской меры. Будем называть это условие достаточным условием невырожденности распределения.

Теорема 1. Если распределения с.в. $\varepsilon_k, k = \overline{1,K}$, задаются непрерывными плотностями, $h_k(z), z \in R^m, k = \overline{1,K}$, удовлетворяющими условию невырожденности,, то

$$\lim_{\rho \to 0} \rho^{-1} \mu_{k,\rho}^{(P_k)}(e)) = h_k(e) \| \operatorname{grad}_z h_k(e) \|, e \in R^1, k = \overline{1,K}, \tag{8}$$

$$\lim_{\rho \to 0} \{\vartheta(S_\rho(0))\}^{-1} \mu_{k,\rho}^{(P_k)}(e)) = h_k(e), e \in R^m, \quad k = \overline{1,K}. \tag{9}$$

Аналогичный результат имеет место и для зондирования множествами $V_\rho(e) = e + \rho V$, определёнными выше, т.е. с функциями принадлежности вида:

$$\mu_{k,\rho}^{(P_k)}(e) = P^{(k)}(V_\rho(e)) = P^{(k)}(e + \rho V) = P\{\varepsilon_k \in e + \rho V\}, k = \overline{1,K}, e \in R^m. \tag{10}$$

Теорема 2.В условия теоремы 1 найдётся $0 < \varphi \le 1$ такое, что:

$$\lim_{\rho \to 0} \rho^{-1} \mu_{k,\rho}^{(P_k)}(e)) = \varphi h_k(e) \| \operatorname{grad}_z h_k(e) \|, e \in R^1, k = \overline{1,K}, \tag{11}$$

$$\lim_{\rho\to 0}\{\vartheta(\rho V))\}^{-1}\mu_{k,\rho}^{(P_k)}(e))=h_k(e), e\in R^m, \quad k=\overline{1,K}. \tag{12}$$

Соотношения (8)-(9), (11)-(12) демонстрируют прямую связь оценивания ситуации по максимуму функций принадлежности, собственно – по максимальным значениям вероятностей множеств-зондов по соответствующим распределениям с классификацией по максимуму плотности соответствующих распределений вероятностей.

Определение. Функция $\hat{P}(e), e\in E, \hat{P}: E\to\{P_1,...,P_K\}$, определённая по набору МНМ $(R^m, \mu_{k,\rho}^{(P_k)}(\cdot)), k=\overline{1,K}$,

одним из соотношений

$$\hat{P}_\infty(e)=P_{k*}, k^*=\arg\max_{k=\overline{1,K}} h_k(e), e\in R^m,$$
$$\hat{P}_\infty(e)=P_{k*}, k^*=\arg\max_{k=\overline{1,K}} h_k(e)\,\|\,\mathrm{grad}_z h_k(e)\,\|, e\in R^m, \tag{13}$$

называется оценкою развития ситуации по максимуму предельных нормированных значений функции принадлежности.

Очевидным образом справедливо следующее утверждение.

Теорема 3. В условиях теоремы 1 оценка ситуации в соответствии с (13) является предельным вариантом определения оценки ситуации для МНМ определяемых с помощью множеств – зондов в соответствии с (4) или (10). Теоремы 1-3 устанавливают связь между классификацией значений К распределений по максимуму функций принадлежности, построенных по множествам - зондам и классификациями типа main –shift [Comaniciu,2002], когда кластеризация проводится, собственно, по максимуму плотностей определённым образом связанных или порождённых наблюдениями. Указанные плотности порождаются элементами обучающей выборки.

Заключение

В предлагаемой работе в развитие работы [Донченко В.С. 2005], рассматриваются модифицированный вариант нечёткости (там же) в рамках которого рассматриваются задачи классификации для К классов.

Доказывается, что при предельном переходе для модифицированных нечётких множеств, построенных на основе множеств – зондов для классов, определяемых распределениями вероятностей в R^m, переходе функция распознавания ситуации, введённая в статье для модифицированных нечётких множеств, превращается в функцию распознавания ситуации, построенную по максимуму плотностей, отвечающих исследуемым классам.

В одном из вариантов используются сами плотности, в другом – умножаются на градиенты.

Литература

[Zadeh, Lotfi, 1965]. Fuzzy Sets.// Information and Control, 8(3). June 1965. pp. 338-53.

[Кофман А.1982]. Введение в теорию нечетких множеств.- Г.: Радио и связь. 1982.- 322 с.

[Донченко В.С.2005]. Нечёткие множества: аксиома абстракции, статистическая интерпретация наблюдения нечётких множеств. //Proceedings: XI-th International Conference “Knowledge – Dialogue - Solution”.– June 20-30 2005. –Varna (Bulgaria) V.1.–с.218-223.

[Comaniciu,2002]. D. Comaniciu. A Robust Approach towards Feature Space Analysis. //IEEE Transactions on Pattern Analysis and Machine Intelligence.– V.24,№5, May 2002.– p. 603-617

Информация об авторе

Донченко Владимир С. *– профессор, Киевский национальный университет имени Тараса Шевченко, факультет кибернетики, кафедра Системного анализа и теории принятия решений; e-mail: vsdon@unicyb.kiev.ua*

A MULTI-AGENT FRAMEWORK FOR DISTRIBUTED DECISION-MAKING SYSTEMS

Vira Lyubchenko

***Abstract**: An approach of building distributed decision support systems is proposed. There is defined a framework of a distributed DSS and examined questions of problem formulation and solving using artificial intellectual agents in system core.*

***Keywords**: decision support system, distributed decision making.*

Introduction

Many definitions for decision support system (DSS) are given in research or review papers. Typical for these definitions is that they all require the involvement of computers to produce information to the decision maker. But often the complexity of a decision problem is a hindrance to the rapid development of safe and effective software for DSS. Many tasks are simply too large for one DSS and require the efforts of many DSSs for distributed decision making. Following Brehmer [Brehmer, 1991] a problem requiring distributed decision making is defined as a problem, which requires

- cooperation from a number of decision makers;
- each decision maker owns part of the resources needed to solve the problem;
- no decision maker has a complete overview of the problem as a whole, and therefore the decision makers must communicate to achieve a shared "situational awareness" with respect to the state of the task.

Framework Description

Let's define a framework of a distributed DSS, which is constructed by some domains specializing in concrete problem field. One of the ways to express the result of such knowledge structuring is a task-oriented stance [Cuena, 1999]. A task is an abstract description of how the world needs to be transformed in order to achieve to desired behavior or functionality. Problem solving methods are used to cope with the task. They indicate how a task is achieved, by describing the different steps by which its inputs are transformed into its outputs. The complexity of the decision problem required compound problem-solving methods that decompose the task into subtask. These subtasks may again be decomposed by some methods, giving rise to a task-methods-subtasks tree, whose leave are given by specialized domain. Each of them has own specialized knowledge base (KB). Each domain has a set of the agents, which are capable to simulate behavior of problem field's objects and are used only for solving tasks of this concrete problem field.

Idea of agent-based structuring integrates a collection of functionalities, achieved by the interplay two kind of knowledge: about certain problem types and about the environment in which the agent operates. By this, the agent can react to the environment situation and can interact with other agents to look for solution to its problems. The notion of agents allows a design of modules that balance two aspects:

- specialty level: it is possible to model a detailed functional decomposition by designing agents that specialized in basic functions;
- autonomy level: it is possible to integrate in an agent a significant set of the functions required for the whole application but limited in scope.

The key concept of our approach to the decision of some problem in distributed environment is analogy with group decision making in human community. For this we define an agent in the system as analogue of the man-expert specializing in the solving of the certain class of tasks. The agent can have wide, but superficial knowledge and skills within the bounds of specialization. But skills of the specialized task decision are not obligatory for it. Such type of the agents is useful, because besides participation in the problem solving it can index highly tailored agents and knowledge. Due to this opportunity such agent can delegate authorities of the problem solving to more “qualified” agents or interrogate the agents for realization of an optimal simulation step.

The set of the specialized agents and knowledge, which are stored in the knowledge base, form some area of specialization that further will be called the domain. Domain can be arranged on the separate physical devices that provide for possibility of high-speed interaction between the agents and high-speed access to data and knowledge. There are no restrictions on number of such domains in distributed DSS. The domains form common distributed system with using of Internet infrastructure. The price of data transfer in the Internet is high. But despite of it, it is necessary to realize interaction between domains of system, because the decision problems, which system needs to solve, are seldom highly tailored and exceed the limits of the domain.

The special search mechanism is determined on a network of the specialized domains. This procedure assists to the user during problem's formalization, and searches the information and agents, which are capable to represent objects of problem field and to simulate their behavior [Choi, 1999]. The idea of search consists in broadcasting search keywords of a problem to all domains. The agents of the domain having such information, form the answer from knowledge base, and send an information package with the identifier of the agent and domain, which has given this information. The user separates packages, which are interest, and thus defines the agents, with which he continues interaction in a process of decision making.

When the problem formulation is terminated, the user has some set of the agents, which he intends to use as the actors for the problem solving. The user determines the characteristics and rules of environment, in which agents will participate, sets the goals of each agent, and defines its strategies.

The characteristics, rules, the initial states of environment and termination condition are transferred to domains, whose agents take part in simulation, so the agents have access to the environment. The agents make the plan of actions according to the goal, given them, and/or make query to the knowledge base for a behavioral model, which was defined by the trainer or was made by other agents and was marked as successful. The agent analyzes the information received by sensors about a status of environment and behavior of the agents-participants, makes a decision, and prepares influence on environment. This influence is reflected on own copy of environment. The change of environment is broadcasted to the agents of other domains through communications channels. The change of environment is distributed to the agents of other domains through lines of the communications. After having received changes of environment from all agents participating in modeling, the agent passes on to the following iteration of reception of the information by sensors. Procedure reiterates. Such approach reduces amount of information transmitted between domains.

During the process of decision making the agent can consult with other agents of the domain. The agent keeps a history of environment states for improved decision making, taking into consideration features of agents' behavior. It can also transfer history to other agents, which take part in problem solving. When agent deviates from the chosen plan, decision about changing the strategy of behavior can be made.

The agents are realize with applications of neural networks and evolutionary technologies, that allows to train them for solution of some class of the tasks, and allows them to store experience and evolutionary in the process of problem solving. Besides this the agents index the information of their domain KB and other agents having the necessary information.

The agent architecture for distributed DSS in such case is built around three major components

- a perception subsystem allows the agent to be situated in the environment by data acquisition and in the society by perceiving agent messages;
- an intelligent subsystem manages the different aspects of information processing as well as full or partial decision making;
- an action subsystem enact the decisions produced by intelligent subsystem, displaying messages to the control personnel or sending messages to other agents.

The agents' dynamic beliefs about the world itself and the others are stored in the KB. We can distinguish two types of information in this KB:

- problem-solving information refers to inputs, outputs and intermediate results of tasks;
- control information specifying in an agenda what is intended to be done.

The task solution is carried out in the domains' network without separation of the agent from the domain, in the virtual environments, which is unique for each separate task. The agents interact due to the special communication protocol that makes illusion of working in common space. The stopping moment for the task

solution is the moment of achievement by the environment of some state, which was defined by user as final. After that the analysis of behavior of the agents, interesting for user, is carried out.

Conclusion

This paper has outlined the potential of multiagent framework for decision support. From an abstract point of view, the concept of an agent has been used as modularization principle for the DSSs' software and knowledge. The results of such modularization are specialized domains. The presented framework is flexible and easily scalable, because domains are independence.

Bibliography

[Brehmer, 1991] Brehmer, B.: Time scales, distributed decision making and modern information technology. In (Rasmussen, J.; Brehmer, B.; Leplat, J. Eds): Distributed decision making: Cognitive models of cooperative work. Wiley, New York, 1991.

[Choi, 1999] Choi, Y.S.; Yoo, S.I.: Multi-Agent Learning Approach to WWW Information Retrieval Using Neural Network. Intelligent User Interfaces, 1999; pp. 23-30.

[Cuena, 1999] Cuena, J.; Ossowski S.: Distributed Models for Decision Support. In (Weiss, G. Ed): Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence. MIT Press, London, 1999.

Author's Information

Vira Lyubchenko *- Odessa National Polytechnic University, Shevchenko av., 1, Odessa 65044, Ukraine; e-mail: vira.lyubchenko@gmail.com*

СИНЕРГЕТИЧЕСКИЕ МЕТОДЫ КОМПЛЕКСИРОВАНИЯ В ЗАДАЧАХ ПРИНЯТИЯ РЕШЕНИЙ

Альберт Воронин, Юрий Михеев

Аннотация*. Предложены синергетические методы комплексирования данных, позволяющие при ограниченном числе каналов получать максимальное количество доступной информации. Вместо редукторов степеней свободы предлагается использовать механизм дискриминаторов степеней свободы, что дает возможность всем каналам, в меру их информативности в текущей ситуации, принимать участие в выработке кооперативного решения.*

Ключевые слова*: Синергетика, комплексирование данных, информация, принятие решений*

Введение

В развитых информационных системах данные, характеризующие состояние одного и того же объекта (процесса) передаются по *нескольким* каналам. Проблема состоит в определении относительной степени достоверности данных, поступающих по каждому из каналов в текущий момент времени и в выработке наиболее достоверной оценки характеристики объекта (процесса) по имеющейся совокупности данных.

Учитывая роль информационных систем в построении современных систем принятия решений и управления, задачу создания методов решения поставленной проблемы следует считать актуальной.

Анализ состояния проблемы

Задача извлечения максимальной информации из имеющейся совокупности различных данных, характеризующих процесс или объект, возникает в самых разнообразных предметных областях. В качестве примера приведем задачу определения высоты самолета по показаниям барометрического,

бортового и наземного радиолокационных высотомеров и, возможно, по визуальному каналу. Каждый из указанных каналов имеет свои преимущества и недостатки в различных условиях полета. Требуется объединить (комплексировать) получаемые данные для наиболее достоверной оценки высоты в текущий момент времени.

В монографии [1] поставлена задача комплексирования показаний приборов, имеющих различный класс точности. При этом каждый из приборов вносит свою лепту в результирующее показание в соответствии со своим классом точности. Здесь же поставлена и решена задача комплексирования данных экспертных оценок с учетом различной степени компетентности экспертов в рассматриваемом вопросе.

В статье [2] описан метод автоматической классификации состояния лесов по материалам аэрокосмической съемки на основе синергетического принципа слияния данных. Здесь определяются наиболее информативные (доминирующие) спектральные каналы сенсора, и по их показаниям принимается искомое решение.

В работе [3] поставлена задача комплексирования сигналов от навигационных полей различной физической природы (радионавигационные поля типа GPS, геофизические поля, поле звезд и тел Солнечной системы и пр.) для наиболее достоверной оценки текущих координат космического аппарата.

В [4] описан метод комплексирования сигналов для бистатической радиолокации малых небесных тел. Для повышения точности измерений при исследовании параметров движения малых небесных тел используется бистатическая конфигурация радиолокационных систем. Информация от каждой из приемных антенн, разнесенных на значительные расстояния, подвергается обработке и сопоставлению между собой так, чтобы результирующий сигнал был наиболее достоверным.

В приведенных примерах используется концепция синергетики [5,6] – науки о кооперативных процессах. В иерархии системных теорий синергетика занимает верхнюю ступень. В отличие от общей теории систем, синергетика изучает и организует процессы, развивающиеся не под централизованными воздействиями, а за счет коллективного взаимодействия компонент в соответствии с поставленной целью. Кооперация компонент позволяет использовать резервные возможности системы и существенно повышает степень эмерджентности (системный эффект).

По определению создателя теории функциональных систем в биологии П.К. Анохина, «системой можно назвать только такой комплекс избирательно вовлеченных компонент, у которых взаимодействие и взаимоотношение приобретают характер *взаимосодействия* компонент на получение фиксированного полезного результата» [7]. Выделенное фундаментальное свойство взаимосодействия представляет собой ярко выраженный и повсеместно проявляющийся в биологических системах синергетический процесс [5].

При синтезе синергетической функциональной системы вначале создаются избыточные степени свободы (закон Эшби [8] о необходимом разнообразии), дающие дополнительные возможности в свойствах системы. Затем в процессе адаптивного взаимосодействия компонент эти степени свободы преодолеваются (редуцируются) по механизму доминанты в процессе функционирования системы. Для этого в системе имеются «редукторы степеней свободы» [5].

Синергетическая концепция комплексирования (слияния) данных активно применяется для извлечения максимальной информации из имеющейся совокупности различных данных не только в биологии, но и в других предметных областях, о чем свидетельствуют приведенные примеры.

Содержательный анализ проблемы

В отличие от биологических и подобных им синергетических систем управления, системы комплексирования, как правило, не располагают избыточным количеством каналов получения данных. Число степеней свободы априори является ограниченным и суть задачи состоит в том, чтобы при этих ограничениях извлечь максимальное количество доступной информации. В приведенных выше примерах действие «редукторов степеней свободы» приводило к отсечению малоинформативных и к выделению одного или нескольких наиболее информативных (доминирующих) в текущей ситуации каналов получения данных, на основе которых формировалось искомое решение.

При таком подходе некоторые полезные нюансы, содержащиеся в отсеченных каналах, не принимают участия в процессе поиска решения, т.е. часть информации теряется. Образно говоря, из всего ансамбля

данных искусственно выделяется один или несколько доминирующих «солистов», в звучании которых отсутствуют те обертоны, которые придают исполнению особую ценность.

Целесообразно при синтезе синергетической системы комплексирования данных отказаться от концепции доминанты и вместо «редукторов степеней свободы» включать механизмы, позволяющие *всем* каналам получения данных участвовать в формировании решения с весами, соответствующими степени их информативности в текущей ситуации («дискриминаторы степеней свободы»). В результате вся доступная информация будет использована надлежащим образом, а «звучание» ансамбля данных будет слаженным и объемным.

Постановка задачи

Дано: количество каналов передачи данных (число степеней свободы в синергетической системе комплексирования) $m \geq 3$. Массив исходных данных представляется в виде матрицы-столбца

$$A^T = \| \alpha_1 \ \alpha_2 \ \ldots \alpha_m \ \|, \tag{1}$$

где $\alpha_j, j \in [1,m]$ – данные о некоторой числовой величине α, полученные по *j*-м каналам (компоненты системы комплексирования).

Ставится задача: получить наиболее достоверную оценку α^* величины α.

Метод решения

Если количество каналов достаточно велико и известно, что степень их информативности приблизительно одинакова, то задача решается простым осреднением по каналам:

$$\alpha^* = \frac{1}{m}\sum_{j=1}^{m}\alpha_j \ .$$

Проблема повышения достоверности оценки возникает, когда число каналов передачи данных невелико, а относительная степень доверия к ним различна и заранее не известна.

В этом случае для решения поставленной задачи воспользуемся механизмом «дискриминаторов степеней свободы» и организуем итерационный синергетический процесс адаптивного взаимосодействия компонент системы комплексирования данных.

Поскольку в начале процесса не известно, какому из каналов больше верить, то сначала считаем, что степень доверия ко всем каналам одинакова и при осреднении их данные принимаются с одним весовым коэффициентом $k_j^I = 1, \ j \in [1,m]$.

В результате осреднения на первой итерации получается средняя оценка

$$\alpha^I = \frac{1}{m}\sum_{j=1}^{m}k_j^I\alpha_j = \frac{1}{m}\sum_{j=1}^{m}1\cdot\alpha_j = \frac{1}{m}\sum_{j=1}^{m}\alpha_j.$$

Назовем ее оценкой первой итерации. Операция осреднения в матричном виде представляет собой умножение матрицы-столбца данных слева на единичную *m*-матрицу-строку (суммирующий вектор)

$$E = \|1 \quad 1 \quad \ldots \quad 1\|$$

и деление произведения на количество каналов:

$$\alpha^I = \frac{1}{m}EA\,.$$

Теперь в нашем распоряжении имеется информация о средней оценке α^I, с которой можно сравнивать оценки по отдельным каналам α_j из матрицы (1). Естественно, что разница между усредненной оценкой (мнение большинства) и оценкой, полученной по данному каналу, может служить основанием для изменения весового коэффициента, с которым воспринимается оценка по данному каналу. Тем каналам, чья оценка на первой итерации ближе к средней, целесообразно повысить коэффициент k_j и, наоборот,

каналам, оценки которых далеки от средней, его следует понизить. В нашей процедуре опускаются те сравнительно редкие случаи, когда "истина" оказывается на стороне меньшинства.

Введем меру («дискриминаторы степеней свободы»)

$$\delta_j^{II} = \left|\alpha^I - \alpha_j\right|, j \in [1,m],$$

которая служит количественным выражением степени доверия к *j*-му каналу на второй итерации. Целесообразно подобрать такие коэффициенты k_j^{II}, которые представляли бы собой функции, обратно пропорциональные δ_j^{II}:

$$k_j^{II} = a/\delta_j^{II},\ a = \text{const}, \tag{2}$$

при условии

$$\sum_{j=1}^{m} k_j^{II} = m. \tag{3}$$

Решая систему уравнений (2) и (3), исключаем неизвестный коэффициент пропорциональности *a* и получаем

$$k_j^{II} = \left(\frac{m}{\delta_j^{II}}\right) / \sum_{t=1}^{m} \left(\frac{1}{\delta_t^{II}}\right).$$

После этого производится осреднение на второй итерации уже с учетом доверия к каналам по результатам первой итерации

$$\alpha^{II} = \frac{1}{m} \sum_{j=1}^{m} k_j^{II} \alpha_j. \tag{4}$$

Введя в рассмотрение матрицу-строку

$$K^{II} = \left\| k_1^{II} \quad k_2^{II} \quad \ldots \quad k_j^{II} \quad \ldots \quad k_m^{II} \right\|,$$

представим выражение (4) в матричном виде

$$\alpha^{II} = \frac{1}{m} K^{II} A.$$

Процесс третьей итерации начинается с установления меры

$$\delta^{III} = \left|\alpha^{II} - \alpha_j\right|, j \in [1, m],$$

и т.д.

Итерационная процедура

$$\alpha^{(g)} = \frac{1}{m} K^{(g)} A,\ g \in [1,h],\ K^I = E$$

продолжается до тех пор, пока не выполняется условие останова

$$\left|\alpha^{(h)} - \alpha^{(h-1)}\right| \le \varphi,$$

где φ – заданная малая величина. Результатом описанной итерационной процедуры является получение уточненной оценки $\alpha^* = \alpha^{(h)}$, определенной с учетом разнородности каналов. В практических случаях итерационный процесс сходится за 3-4 итерации. Заметим, что в данном случае с уменьшением величины φ оценка по принципу «дискриминаторов степеней свободы» асимптотически вырождается в оценку по принципу «редукторов степеней свободы».

Синергетические аспекты математической статистики

Синергетический принцип комплексирования данных имеет много общего с идеями математической статистики [9]. Действительно, если синергетическая концепция комплексирования (слияния) данных применяется для наиболее достоверной оценки характеристик процессов (объектов) по имеющейся совокупности данных, то математическая статистика изучает методы наиболее достоверной оценки моментов распределения случайных величин по имеющейся совокупности элементов выборки. Общность проблем обеих теорий делает задачу исследования синергетических аспектов математической статистики актуальной как для синергетики, так и для развития статистических методов.

Рассмотрим задачу обработки информации при ограниченном числе каналов передачи данных как вычисление уточненной оценки θ^* параметра θ распределения $f(x/\theta)$ случайной величины X на основе статистического материала ограниченного объема $x=x^{(n)}=(x_1, x_2, \ldots, x_n)$ – аналог степеней свободы синергетической системы комплексирования данных.

Для решения задачи применим байесовский подход [9]. Используется априорная информация о том, что несмещенная оценка параметра θ, рассматриваемая как случайная величина, распределена по тому же закону, что и X. Минимизация функции риска при квадратичной функции потерь дает выражение для оптимальной оценки как апостериорного математического ожидания параметра θ, вычисляемого по заданному вектору наблюдений:

$$\theta^* = \int_{-\infty}^{+\infty} \theta f(\theta|x) d\theta \Big|_{x=x^{(n)}}. \tag{5}$$

Воспользуемся определением апостериорной плотности по теореме Байеса [9]:

$$f(\theta|x) = \frac{f(x|\theta) f_a(\theta)}{f(x)},$$

где маргинальное распределение $f(x)$ выражается формулой

$$f(x) = \int_{-\infty}^{+\infty} f(x|\theta) f_a(\theta) d\theta.$$

Тогда выражение (5) преобразуется к виду

$$\theta^* = \frac{\int_{-\infty}^{+\infty} \theta f(\theta|x) f_a(\theta|\theta') d\theta}{\int_{-\infty}^{+\infty} f(\theta|x) f_a(\theta|\theta') d\theta} \Big|_{\bar{x}=\bar{x}^{(n)}}, \tag{6}$$

где θ' – неизвестная константа. Поскольку искомая оценка $\theta^* = \theta^*(x_1, x_2, \ldots, x_n)$ должна вычисляться по заданному вектору наблюдений, то мы должны перейти в выражении (6) от интегралов к суммированию по элементам заданной выборки и заменить неизвестные константы их оценками:

$$\theta^* = \frac{\sum_{i=1}^{n} x_i f(x_i|\theta^*) f_a(x_i|\theta^*)}{\sum_{i=1}^{n} f(x_i|\theta^*) f_a(x_i|\theta^*)}. \tag{7}$$

Формула (7) выражает зависимость

$$\theta^* = \varphi(x_1, x_2, \ldots, x_n; \theta^*).$$

Как известно [10], уравнение в такой форме можно решать итерационным методом. Итерационная процедура организуется в соответствии с рекуррентной формулой

$$\theta^*[l] = \varphi(x_1, x_2, ..., x_n; \theta^*[l-1]), l \in [1, L],$$

причем итерационный процесс заканчивается при выполнении условия

$$\theta^*[L] - \theta^*[L-1] \le \lambda_\theta ,$$

где l - номер текущей итерации; λ_θ – заданная точность вычисления оценки θ^*. Если необходимо проанализировать вопросы сходимости, то можно применить известную теорему [10], в соответствии с которой для сходимости итерационного процесса достаточно, чтобы на рассматриваемом интервале уточнения оценки θ^* соблюдалось неравенство

$$\left| d\varphi(x_1, x_2, ..., x_n; \theta^*) / d\theta^* \right| < 1 .$$

Общее выражение для уточненной оценки (7) полностью соответствует следующей идее Гаусса [11]. Наиболее вероятно такое значение оцениваемого параметра, при котором минимизируется сумма квадратов разностей между действительно наблюдаемыми и вычисленным значениями, умноженных на весовой коэффициент k_i , отражающий относительное доверие к наблюдениям:

$$\theta^* = \arg\min_{\theta^*} \sum_{i=1}^{n} k_i (x_i - \theta^*)^2 . \quad (8)$$

В [12,13] показано, что выражение (7) действительно получается из (8), если в качестве меры относительного доверия к наблюдениям ввести апостериорную плотность распределения вероятностей («дискриминаторы степеней свободы»).

Таким образом, предложенная методика предусматривает индивидуальный подход к каждой реализации случайной величины (взвешивание в соответствии с апостериорной вероятностью ее появления), что позволяет [14] устранить потери информации при вычислении искомых оценок по малой выборке.

Важно отметить, что выработка наиболее достоверной оценки осуществляется посредством организации итерационного процесса, в котором элементы выборки на каждой итерации адаптационно взаимодействуют друг с другом. Аналогичным образом, синергетика предусматривает процесс, характеризующийся самоорганизацией в соответствии с поставленной целью. Адаптационные процессы развиваются посредством коллективного взаимодействия компонент. Кооперация компонент включает резервные возможности системы и значительно увеличивает степень эмерджентности (системный эффект).

Литература

1. *Воронин А.Н., Зиатдинов Ю.К., Харченко А.В.* Сложные технические и эргатические системы: методы исследования. – Харьков: Факт, 1997. – 240 с.
2. *Лялько В.И., Федоровский А.Д., Попов М.А.* Использование данных спутниковой съемки для изучения природоресурсных проблем // Космические исследования в Украине (2002-2004) – Киев: НКАУ, 2004. – С.7-14.
3. *Варламов І.Д., П'ясковський Д.В., Водоп'ян С.В.* Адаптивний кореляційно-екстремальний алгоритм навігації космічного апарата по геофізичних полях на основі диференціально-тейлорівських перетворень // Космічна наука і технологія. – 2001. – №4. –С.141-146.
4. *Воронин А.Н.* Метод комплексирования сигналов для бистатической радиолокации малых небесных тел // Тезисы докладов 9-й международной конференции «Системный анализ и управление». – М.: изд-во МАИ, 2004. – С.113-114.
5. *Колесников А.А.* Синергетическая теория управления. – М.: Энергоатомиздат, 1994. – 344с.
6. *Хакен Г.* Синергетика. – М.: Мир, 1980. – 248 с.
7. *Анохин П.К.* Очерки по физиологии функциональных систем. – М.: Медицина, 1975. – 184с.
8. *Эшби У.Р.* Введение в кибернетику. – М.: Изд-во иностр. лит-ры, 1959. – 318 с.
9. *Кокс Д., Хинкли Д.* Теоретическая статистика. – М.: Мир, 1978. – 560 с.
10. *Гутер Р.С., Резниковский П.Т.* Программирование и вычислительная математика. М.: Наука, 1971. Вып.2. 273с.
11. *Сейдж Э., Мелс Дж.* Теория оценивания и ее применение в связи и управлении. – М.: Связь, 1976. – 496 с.

12. *Воронин А.Н.* О повышении эффективности статистических оценок параметров эргатических систем // Кибернетика и вычислительная техника. – 1980. – Вып.50. – С. 29-31

13. *Voronin A.N.* On the rise of efficiency of statistical estimates for parameters of ergatic systems // Zentralblatt fur Mathematik und ihre Grenzgebiete.Mathematics Abstracts. - Band 484. - Berlin. Heidelberg. New York. – 24.01.1983. – P.375.

14. *Гаскаров Д.В., Шаповалов В.И.* Малая выборка. – М.: Статистика, 1978. – 248 с.

Сведения об авторах

Воронин Альберт Николаевич *– профессор, доктор технических наук, профессор кафедры компьютерных информационных технологий Национального авиационного университета; проспект Комарова, 1, Киев-58, 03058 Украина;*

Михеев Юрий Иванович *– адъюнкт Житомирского военного института радиоэлектроники им. С.П. Королева, проспект Мира, 22, Житомир, 10004, Украина.*

OPERATING MODEL OF KNOWLEDGE QUANTUM ENGINEERING FOR DECISION-MAKING IN CONDITIONS OF INDETERMINACY

Liudmyla Molodykh, Igor Sirodzha

Abstract*: The operating model of knowledge quantum engineering for identification and prognostic decision-making in conditions of α-indeterminacy is suggested in the article. The synthesized operating model solves three basic tasks: A_t-task to formalize tk-knowledge; B_t-task to recognize (identify) objects according to observed results; C_t-task to extrapolate (prognosticate) the observed results. Operating derivation of identification and prognostic decisions using authentic different-level algorithmic knowledge quantum (using tRAKZ-method) assumes synthesis of authentic knowledge quantum database (BtkZ) using induction operator as a system of implicative laws, and then using deduction operator according to the observed tk-knowledge and BtkZ a derivation of identification or prognostic decisions in a form of new tk-knowledge.*

Keywords*: operating model, decision-making object, knowledge quantum database, target feature, method of different-level algorithmic knowledge quantum, implicative law.*

ACM Classification Keywords*: I.2.3 Deduction and Theorem Proving; I.2.4 Knowledge Representation Formalisms and Methods; I.2.5 Programming Languages and Software*

Introduction

Knowledge-oriented modelling of human being's intellectual skills to make decisions in conditions of indeterminacy to recognize patterns and prognostic situations for artificial intelligence systems (AIS) is being developed in the article. Operating model for knowledge quantum engineering for decisions derivation in conditions of indeterminacy, which is based on using the **method of authentic different-level algorithmic knowledge quantum or portions (tRAKZ-method)** is suggested. The existing artificial neural networks (ANN) and knowledge engineering methods, based on frame, production and other knowledge models, are not effective enough because of the imperfection of representation ways and computer knowledge manipulation. Unlike these approaches the suggested model has a form of strictly formalized knowledge quantum, different in the level of complexity (tk-knowledge). Such tk-knowledge as substantial algorithmic structures of authentic data allow computer manipulation of knowledge using an finite predicates algebra and vector-matrix operators, and also inductive synthesis of **knowledge quantum database (BtkZ)** while teaching computer using selective plot examples of situations from the concrete data domain.

1. Target Setting

The model-based process of **human's classification** and **prognostic** decision-making in conditions of *indeterminacy* is always aimed (motivated by a target criterion) at the **decision-making object** (**DMO**), which can be described with a set of characteristics (features), measured in different scales and allowing logical representation. **Target** features are also contained in this set. Their values determine the **class** and **pattern** of the considered **DMO**. To **identify** the class (pattern) of DMO, i.e. to **make a classification decision**, means to define a value of the **target feature** according to the observed initial characteristics, relying on the **knowledge quantum database** (BtkZ), represented by **classification law** systems. Analogically to make a **prognostic decision** it is necessary to have a **prognostic BtkZ**, allowing to define the value of the **target prognostic feature** on the segment **t+Δt**, according to the situation on the time segment **t**.

The discussed **α-indeterminacy** is characterized by such limitations:

- data about DMO are of different type (i.e. measured in quantitative as well as in qualitative scales) and can be reached in incomplete volumes and from different sources (experts, technical documentation, reference books, instruments measurements etc.);
- the target criteria are given implicitly, it is unknown which ones, in what quantity and how to select informative features of DMO according to targets of decision-making;
- the rules of making classification and prognostic decisions are unknown, and also the inductive principles of their building by teaching on selective experimental data are unknown too;
- the sought rules of decision-making are impossible to be defined by regular calculus of approximations directly, but it is possible to create knowledge engineering tools to model and imitate intellectual human's skills to find solutions, relying on intuition and knowledge database.

In α-indeterminacy the **authentic** k-knowledge (**tk-knowledge**) are used.

The main task of this article is to create a method of synthesis for operating model in knowledge quantum engineering to derive classification and prognostic decisions in conditions of α-indeterminacy. In general this task is deducted to solving three basic tasks [Sirodzha, 2002]:

1. **A_t-task** for formalization of tk-knowledge;
2. **B_t-task** for object recognition (identification) according to observation results;
3. **C_t-task** for extrapolation (prognostic) of observation results.

In the **A_t-task** it is required to define the terms "**tk-knowledge**" and "**tRAKZ-models**" formally in conditions of **α-**indeterminacy, to describe their algorithmic design using quantum structuring of different-type data about DMO considering its semantics in a concrete data domain.

A_t-task is described formally using the multiple four:

$$A_t = \langle S, K_t, \Pi_t, Q_t \rangle \quad (1)$$

and consists in building the class M_t of substantial algorithmic structures and operating tools for manipulating them on a character language S from a set of letters, numbers, special symbols and algorithmic operations of algorithm theory on the basis of using rules for constructing t-quantum Π_t to terminal t-quantum from K_t with a help of finite set Q_t of semantic codes. Under semantic code $tk_s \in Q_t$ (s=0,1,2,…) we assume symbols, coding t-quantum, which corresponds the form and content of authentic knowledge of level s.

The **B_t-task** is to synthesize **recognizing** tRAKZ-models and algorithms to manipulate tk-knowledge to define values of **target characteristic** for the recognized DMO, i.e. its **identification** with the given reliability according to the external observations, relying on the preliminary cumulated BtkZ.

The **C_t-task** is to synthesize **prognostic** tRAKZ-models and algorithms for manipulation tk-knowledge to **predict** with the given reliability of **DMO permanent characteristics** values according to the measured values of the observed characteristics, relying on the preliminary built BtkZ.

To solve B_t- and C_t-tasks it is required:

1) **to synthesize the induction operator** $\mathrm{IND}\$\left(\mathrm{tk}_2\Sigma_0; \mathrm{AZ}; \mathrm{tk}_2\overline{\Sigma_{\mathrm{BM}}}\right)$ for inductive derivation of the sought BtkZ from a set of selected teaching tk-knowledge, where in brackets the parameters of INDS operator are shown: $\mathrm{tk}_2\Sigma_0$ - teaching selective tk-knowledge of the 2nd level; AZ – operating algorithm of inductive derivation for BtkZ as new knowledge; $\mathrm{tk}_2\overline{\Sigma_{\mathrm{BM}}}$ - minimized BtkZ in the form of a matrix t-quantum of the 2nd level as a system of implicative laws.
2) **to synthesize the deduction operator** DED($\mathrm{tk}_2\Sigma_0$;$\mathrm{tk}_1\mathrm{Y}_\omega$;AL;$\mathrm{tk}_s$R) for deductive derivation of the sought decision as a new tk-knowledge of the level s (s=1,2) tk_sR in observations $\mathrm{tk}_1\mathrm{Y}_\omega$ for DMO ω, relaying on BtkZ= $\mathrm{tk}_2\overline{\Sigma_{\mathrm{BM}}}$, where AL is a deduction *algorithm*.

2. Algorithmic Formalization and Vector-Matrix Representation of tk-knowledge (A_t-task)

The general structure of **t-quantum of knowledge (tk-knowledge)** has two compounds: ***semantic*** and ***informational*** to represent a ***knowledge portion*** about DMO conditions in **semantic**, **informational** and **algorithmic** aspects at the same time. It is supposed that a portion (quantum) of knowledge about the DMO condition describes some authentic quantum event (QE) in a production form "***message - consequence***" according to the scheme (2)

IF (logical combination of messages e_i), THEN (consequence C_j),
i =1, k; j = 1, h. (2)

Semantic compound of t-quantum in a form of **special structure of data** represents **meaning information** about this **QE**, showing the *scales for measuring* the DMO features, *semantic code* and quantum purpose as *knowledge model* about facts or laws. **Semantic code** from the set $\mathbf{Q}_t$ has a symbolic form $\mathbf{tk_sY_\omega}$, **k** is a quantum symbol; **s** ∈ {0,1,2…} is a level**, Y** is a name and **ω ∈ {p, tr, b, t…}** – quantum status (**p**recondition, **tar**get, **b**asic, **t**erminal).

Information compound describes different-type features (characteristics) of DMO in a sectioned (domain) vector-matrix form, suitable to ***manipulate tk-knowledge*** and ***logical derivation*** using **computer algebra**. In a substantial and formal representation the domains $\mathbf{d}_j$ meet ***non-target*** (precondition) and ***target*** features of DMO, they are called **active** and are separated by a symbol **" : "**. Binary components of active domains $\alpha^i_j \in \mathbf{d}_j$ correspond to the features values. All the active domains define the QE logics, as far as a postulate is taken about the fact that active domains are connected with a **conjunction** (**":"** is a strap "$\wedge$"), the compounds in domains – with a **disjunction** (**","** is a strap "$\vee$"), and precondition domains to a target – with an **implication** («$\Rightarrow$») in a form of (2). The logics of QE can be described in *sentential formulas* of ***propositional logic*** or in *finite predicates*, where the arguments are components of α_i^j domains.

The main idea of **strictly formalization** is in axiomatic building of tRAKZ-model on the basis of postulating the three **terminal** quanta $\mathbf{tk_1y_T}$, $\mathbf{tk_0a_T}$, $\mathbf{tk_1b_T}$ and using operators of **superposition (∏-operator)** known in the theory of algorithm, a **string concatenation (CON⟨•⟩-operator)** and a **column concatenation (CON[•]-operator).**

The *generalized* **terminal** quantum $\mathbf{tk_1y_T}$ represents a **vector of domains**, corresponding to different-type features $\mathbf{x_1, \dots, x_n}$ DME with values (in the domain components) from the finite sets $\mathbf{X}^i$, (**j**=1,2,…, **n**): $\mathrm{X}^1 = \{\alpha_1^1,\dots,\alpha_{r_1}^1\},\dots,\mathrm{X}^n = \{\alpha_1^n,\dots,\alpha_{r_n}^n\}$, .The *generalized* quantum $\mathbf{tk_1y_T}$ has a form:

$$\mathrm{tk}_1\mathrm{y}_\mathrm{T} = [\overset{x_1}{\mathrm{d}_1} : \overset{x_2}{\mathrm{d}_2} : \dots : \overset{x_n}{\mathrm{d}_n}] = [\alpha_1^1,\dots,\alpha_{r_1}^1 : \alpha_1^2,\dots,\alpha_{r_2}^2 : \dots : \alpha_1^n,\dots,\alpha_{r_n}^n], \quad (3)$$

where $\mathrm{tk}_1 \in \mathrm{Q}_\mathrm{T}$; name $y_T \in S_v$.

The *generalized* **terminal selecting** quantum $\mathbf{tk_0a_T}$ is described with a **selection function $V_k^{(t)}$** of the argument α_k from t-consequence of numbers or symbols:

$$\mathbf{tk_0a_T} = [\mathrm{V}_\mathrm{k}^{(\ell)}(\alpha_1,\dots,\alpha_\mathrm{k},\dots,\alpha_\ell) = \alpha_\mathrm{k}],$$
$$\text{where } \mathrm{tk}_0 \in \mathrm{Q};\quad \text{name } \mathrm{a}_\mathrm{T},\ \mathrm{V}_\mathrm{k}^{(\ell)} \in \mathrm{S}; \quad (4)$$

The *generalized* **terminal characteristic** quantum $\mathbf{tk_1b_T}$ is described with a characteristic function χ_{Y_j} of a set $\mathbf{Y}_j$ for admissible values α_i^j of the j feature x_j:

$$\mathbf{tk_1b_T} = [\chi_{Y_j}(\alpha_k^j)] = \begin{cases} 1, \text{if } \alpha_k^j \in Y_j, \\ 0, \text{if } \alpha_k^j \notin Y_j, \end{cases} \qquad k = (1,2,...,r_j). \tag{5}$$

Definition 1. The different-level algorithmic structures, being received from terminal quantum $\mathbf{tk_1y_T}$(3), $\mathbf{tk_0a_T}$(4) and $\mathbf{tk_1b_T}$(5) with a help of finite number of applying **∏-operator**, **CON⟨•⟩-operator** and **CON[•]-operator**, are called **different-level algorithmic tk-knowledge** or **tRAKZ-models** of knowledge in conditions of **α-indeterminacy**, which form a class of authentic tRAKZ-models M_t.

In Fig.1 a quantum area $B_t^{(3)}$ of tRAKZ-model of DMO is shown, being described by three features: $\mathbf{x_1}$ with $\mathbf{r_1}$ = 2 values from $\mathbf{X^1} = \{\alpha_1^1, \alpha_2^1\}$; $\mathbf{x_2}$ with $\mathbf{r_2}$ = 4 values from $\mathbf{X^2} = \{\alpha_1^2, \alpha_2^2, \alpha_3^2, \alpha_4^2\}$ and $\mathbf{x_3}$ with $\mathbf{r_3}$ = 3 values from the set $\mathbf{X^3} = \{\alpha_1^3, \alpha_2^3, \alpha_3^3\}$.

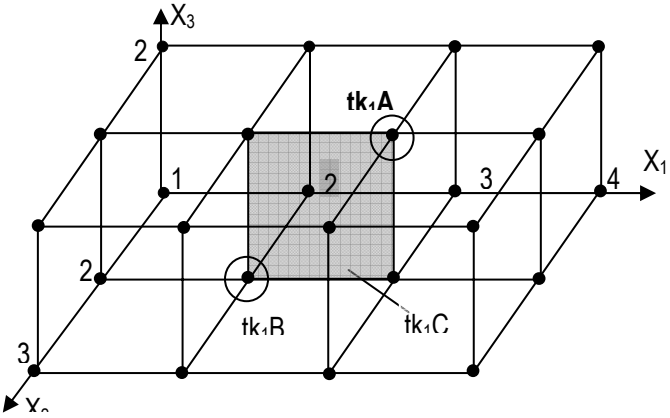


Fig.1. Area $B_t^{(3)}$ of **tRAKZ**-model

Vector domains are separated with a semicolon «:» and meet the different-type features of DMO, and components of domains – for the features values so that **i** component of **j** domain should contain «1», if we observe **i** value of **j** feature, otherwise **i** component equals to «0». If every domain of a **quantum of the 1st level** contains strictly only one «1», it is called an **element** one, otherwise it is called – an **interval** vector quantum. The points A and B of the area $B_t^{(3)}$ are responsible for element vector tk-knowledge $\mathbf{tk_1A}$ and $\mathbf{tk_2B}$:

$$tk_1A = \left[\overbrace{01}^{x_1} : \overbrace{0010}^{x_2} : \overbrace{010}^{x_3}\right], \qquad tk_1B = [10 : 0100 : 010], \tag{6}$$

The **interval C** $\subset B_t^{(3)}$ corresponds with an authentic **interval vector quantum of the 1st level**

$$tk_1C = [\overbrace{11}^{x_1} : \overbrace{0110}^{x_2} : \overbrace{010}^{x_3}], \tag{7}$$

which can be represented by a **matrix t-quantum of the 2nd level $\mathbf{tk_1C}$**, containing the joint 4 element vector t-quantum of the 1st level:

$$tk_2C = \begin{bmatrix} \overbrace{01}^{x_1} : \overbrace{0010}^{x_2} : \overbrace{010}^{x_3} \\ 10 : 0010 : 010 \\ 01 : 0100 : 010 \\ 10 : 0100 : 010 \end{bmatrix} \tag{8}$$

Besides, is t-quantum tk_1C (7) represents a conjunct, an elementary conjunction corresponds to it:

$$\left(x_1 \in \left\{\alpha_1^{(1)}, \alpha_2^{(1)}\right\}\right) \wedge \left(x_2 \in \left\{\alpha_2^{(2)}, \alpha_3^{(2)}\right\}\right) \wedge \left(x_3 \in \left\{\alpha_2^{(3)}\right\}\right) \tag{9}$$

The elementary conjunction (9) can be represented as a predicate equation:

$$\left(\left(x_1 = \alpha_1^{(1)}\right) \vee \left(x_1 = \alpha_2^{(1)}\right)\right) \wedge \left(\left(x_2 = \alpha_2^{(2)}\right) \vee \left(x_2 = \alpha_3^{(2)}\right)\right) \wedge \left(x_3 = \alpha_2^{(3)}\right) = 1 \tag{10}$$

So, the class $\mathbf{M_t}$ of **tRAKZ**-models represents a set of *uniform quantum tools* for describing **implicative laws**, and also different **facts** to represent them in the three equivalent forms: ***multiple*** (points, intervals of area $\mathbf{B_t^{(n)}}$); ***vector-matrix*** (domain structures); ***analytic*** (finite predicates).

3. Inductive Search and Deductive Derivation of Solutions as tk-knowledge

Under the **facts** we understand the *measured* DMO features of different type and their logical combinations, and also any *observed* events and situations, having relation to DMO and being represented by ***knowledge quantum*** of different levels, i.e. by **tRAKZ-models**. The tables **of empirical data** (TED) $\mathbf{T_o(m,n)}$ are typical examples of real facts.

Under the **laws** (DMO are subordinated to them) we consider **implicative** (***forbidden***) **logical connections** between *features* of **DMO**, they are rather **stable** to be defined while analyzing a limited TED $\mathbf{T_o(m,n).}$

Definition 2. A **stable connection** between **r** characteristics of DMO from the general number of **n**, (**r≤n**), expressing ***inadmissibility*** of at least one combination of their values on a set of **tk-knowledge**, is called an **implicative law** or a **prohibition** of **r rank**.

In **tRAKZ**-method of decision-making the ***inductive derivation of tk-knowledge*** is used to build a general "**world model**" in a form of **BtkZ** as a range of ***implicative*** laws being found by ***learning tk-knowledge***, represented in a form of TED.

The ***deductive derivation*** of **tk-knowledge** is necessary to receive partials ***conclusions*** for the *observed* facts, basing on the BtkZ.

3.1. Inductive derivation operator of implicative BtkZ (INDS-operator)

The existence of implicative law as some forbidden knowledge quantum of s-level $\mathrm{tk}_s\overline{Y}$ from $\mathbf{T_r}$, according to TED $\mathbf{T_0(m,N)}$, (**s**=1,2), is defined by the ***evaluation*** of its ***certainty***, satisfying the inequality

$$\mathbf{M_S\{m,N,r\}}=\frac{N!\cdot 2^{r(1-m)}\cdot(2^r-1)^m}{r!\cdot(N-r)!}\le M_S^* \qquad (11)$$

where the given **possible** *limit* **value** (*threshold*) of $\mathbf{M^*_S}$ [Sirodzha,1992] evaluation**.**

In a practical diapason of values **m** and **N** rank $\mathbf{r_{max}}$ turns out to be **small.** This allows defining all the **implicative** laws using a check for intervals «**forbiddances**» of a rank that is *not more than* $\mathbf{r_{max}}$. The disjunctive union of all the found forbidden intervals as conjunctions of combinations of informative features of DMO forms an analytic (predicate) description of the ***forbidden area***, corresponding BtkZ.

Definition 3. The algorithmic procedure

$$\mathrm{INDS}\left(\mathrm{tk}_2\Sigma_0;\mathrm{AZ};\mathrm{tk}_2\overline{\Sigma_{BM}}\right)=\mathrm{tk}_2\Sigma_0\frac{\mathrm{INDS}}{\mathrm{AZ}}\to\mathrm{tk}_2\overline{\Sigma_{BM}}, \qquad (12)$$

implementing **inductive derivation** of non-odd **BtkZ**= $\mathrm{tk}_2\overline{\Sigma_{BM}}$ in a form of a set of **simple** *prohibitions* from the learning knowledge quantum $\mathbf{tk_2\Sigma_0}$ using the algorithm **AZ**, is called an **operator of inductive derivation of implicative tk-knowledge (INDS-operator)** [Sirodzha, 1992].

Algorithm AZ

Input: TED in a form of quantum $\mathrm{tk}_2\Sigma_0$ of size m×n, threshold $M^*_S=10^{-2}$, maximal rank r_{max} = 3.

Output: minimized BtkZ = $\mathrm{tk}_2\overline{\Sigma_{BM}}$ as a system of simple forbidden quanta, i.e. that do not result one from another.

Steps:

1. according to r_{max} patterns of features prohibitions combinations are formed. For r_{max} = 3 there are 8 patterns: <000>, <001>, <010>, <011>, <100>, <101>, <110>, <111>. Forbidden combinations are searched between domains components, but not inside a domain.
2. In the cycle in $\mathrm{tk}_2\Sigma_0$ all the combinations of features values are taken as doubles, and then as triples, etc. till r_{max}. The non-found in $\mathrm{tk}_2\Sigma_0$ pattern combinations are added to $\mathrm{tk}_2\overline{\Sigma_B}$.
3. The formed quantum of prohibitions $\mathrm{tk}_2\overline{\Sigma_B}$ is ***minimized*** in BtkZ = $\mathrm{tk}_2\overline{\Sigma_{BM}}$ using operators of gluing, merging and compression.

Let's assume that in the result of step 2 in the algorithm AZ we got a quantum $\mathrm{tk}_2\overline{\Sigma_B}$. DMO is characterized by three features x_1, x_2, x_3.

$$tk_2\overline{\Sigma_B} = \begin{bmatrix} \underbrace{x_1} & \underbrace{x_2} & \underbrace{x_3} \\ 01-:-1:-1-- \\ 01-:0-:-1-- \\ -10:1-:---0 \\ ---:1-:---0 \\ 1--:-0:--1- \\ 0--:-0:--1- \end{bmatrix},$$ where «-» defines «it is indifferent if it is 0 or 1».

3.1. Gluing $(xy \vee x\bar{y} = x)$

$$tk_2\overline{\Sigma_B} = \begin{bmatrix} 01- : -1 : -1-- \\ 01- : 0- : -1-- \\ -10 : 1- : ---0 \\ --- : 1- : ---0 \\ \mathbf{1-- : -0 : --1-} \\ \mathbf{0-- : -0 : --1-} \end{bmatrix} \Rightarrow tk_2\overline{\Sigma_{B1}} = \begin{bmatrix} 01- : -1 : -1-- \\ 01- : 0- : -1-- \\ -10 : 1- : ---0 \\ --- : 1- : ---0 \\ \mathbf{--- : -0 : --1-} \end{bmatrix}$$

3.2. Merging $(xy \vee x = x)$

$$tk_2\overline{\Sigma_{B1}} = \begin{bmatrix} 01- : -1 : --0- \\ 01- : 0- : -1-- \\ \mathbf{-10 : 1- : ---0} \\ \mathbf{--- : 1- : ---0} \\ --- : -0 : --1- \end{bmatrix} \Rightarrow tk_2\overline{\Sigma_{B2}} = \begin{bmatrix} 01- : -1 : -1-- \\ 01- : 0- : -1-- \\ \mathbf{--- : 1- : ---0} \\ --- : -0 : --1- \end{bmatrix}$$

3.3. Compression *(union of quanta different with one domain only)*

$$tk_2\overline{\Sigma_{B2}} = \begin{bmatrix} \mathbf{01- : -1 : -1--} \\ \mathbf{01- : 0- : -1--} \\ --- : 1- : ---0 \\ --- : -0 : --1- \end{bmatrix} \Rightarrow tk_2\overline{\Sigma_{BM}} = \begin{bmatrix} \mathbf{01- : 01 : -1--} \\ --- : 1- : ---0 \\ --- : -0 : --1- \end{bmatrix}$$

After steps 3.1-3.3 under the whole forbidden quantum database we get the searched minimized implicative BtkZ= $tk_2\overline{\Sigma_{BM}}$.

3.2. Deductive derivation operator of decisions from implicative tk-knowledge.

It is necessary to solve the task of building the algorithm AL, implementing ***deductive*** *operating process* to search the *needed* decisions being correspondent with the ***logical consequence*** $tk_2\|Y\|, tk_1Y, tk_0\beta_{ik}^{(j)}$**:**

$$tk_2\overline{\Sigma_{BM}} \underset{AL1}{\overset{DED}{\Rightarrow}} tk_2\|Y\|, \quad tk_2\overline{\Sigma_{BM}} \underset{AL3}{\overset{DED}{\Rightarrow}} tk_1Y, \quad tk_2\overline{\Sigma_{BM}} \underset{AL2}{\overset{DED}{\Rightarrow}} tk_0\beta_{ik}^{(j)}, \qquad (13)$$

where $tk_2\overline{\Sigma_{BM}}$ is a known database of ***implicative*** **tk-knowledge**.

The searched sequences $tk_2\|Y\|$, tk_1Y $tk_0\beta_{ik}^{(j)}$ (13) represent the different-level tk-knowledge, characterizing the decisions being made in basic tasks $\mathbf{B_t}$ and $\mathbf{C_t}$ according to the observed results.

Let a base of implicative tk-knowledge $tk_2\overline{\Sigma_{BM}}$ is given and a quantum tk_1Y_ω of knowledge about the *observed* DMO $\omega \in \Omega$ of a data domain being investigated. The **algorithm AL** to evaluate the ***possible condition of DMO*** ω according to *quanta of observations* tk_1Y_ω, based on a ***known*** **BtkZ,** is a implementation of ***deductive derivation*** for the searched decision according to the scheme (13). Let's note that under the possible condition of DMO ω we understand a *class* or *pattern* and the DMO ω is concerned to it while solving the $\mathbf{B_t}$**-**task or a *category* (*value*) of prognosis connected with DMO ω if we solve the $\mathbf{C_t}$**-task**.

Algorithm AL

Input**:** tk-knowledge **BtkZ**= $tk_2\overline{\Sigma_{BM}}$ and observations tk_1Y_ω for DMO ω.

Output**:** deductively derived **tk**-knowledge $tk_2\|\overline{Y}_\omega^*\|$ from **BtkZ** about the possible condition of DMO ω, according to the observations tk_1Y_ω.

Steps**:**

1. To make a substitution of quantum values tk_1Y_ω in **BtkZ**= $tk_2\overline{\Sigma_{BM}}$ in this way: to delete columns in a matrix quantum $tk_2\overline{\Sigma_{BM}}$, meeting the features of the observed quantum tk_1Y_ω.

2. To delete the rows, which are orthogonal to the observation $tk_1 Y_\omega$ row, from the formed minor (respectively to the known features; 'orthogonal' means those having opposite in the meaning). In such a way we get $tk_2 \left\| Y_\omega^* \right\|$.

3. To invert the received quantum and consider it to be the result $tk_2 \Sigma_\omega^* = tk_2 \left\| Y_\omega^* \right\|$.

The algorithm is analogical for deriving the logical sequences $tk_1 Y$, $tk_0 \beta_{ik}^{(j)}$ [Sirodzha, 2002].

Let's assume the DMO is characterized with 4 features (x_1,x_2,x_3,x_4), and the BtkZ has been inductively received in a form of:

$$tk_2 \overline{\Sigma_{BM}} = \begin{bmatrix} \overbrace{01-}^{x1} : \overbrace{-1}^{x2} : \overbrace{-1---}^{x3} : \overbrace{01}^{x4} \\ 01- : 0- : -1--- : 1- \\ -10 : 1- : ---0 : 1- \\ --- : 1- : ---0 : 0- \\ 1-- : -0 : --1- : -0 \\ 0-- : -0 : --1- : -1 \end{bmatrix}$$

There is also a quantum to observe the DMO $tk_1 Y_\omega$=[001:10:0100:--]. It is required to define the possible value of the non-measured feature **x_4**. According to the algorithm steps we get:

$tk_1 Y_\omega$ **= [001 : 10 : 0100 : --]**

$$tk_2 \overline{\Sigma_{BM}} = \begin{bmatrix} 01- : -1 : -1-- : 01 \\ 01- : 0- : -1-- : 1- \\ -10 : 1- : ---0 : 1- \\ --- : 1- : ---0 : 0- \\ 1-- : -0 : --1- : -0 \\ 0-- : -0 : --1- : -1 \end{bmatrix} \Rightarrow \begin{bmatrix} 01- : -1 : -1-- : 01 \\ 01- : 0- : -1-- : 1- \\ -10 : 1- : ---0 : 1- \\ \mathbf{--- : 1- : ---0 : 0-} \\ 1-- : -0 : --1- : -0 \\ 0-- : -0 : --1- : -1 \end{bmatrix}$$

After applying algorithm AL steps 1,2 a quantum [--- : 1- : ---0 : 0-] is left. After the inversion (step 3 of the algorithm AL) $tk_0 \beta_{ik}^{(j)}$=[1], i.e. the 4th feature (the 4th domain corresponds to it) takes the first value. Analogically the tasks to prognosis the several features values are being solved. In such a way the B_t-, C_t-tasks have been solved with a help of the algorithm AL.

Conclusion

Operating derivation of identification and prognostic decisions using tRAKZ-method suppose such a sequence of ***operating*** transformations of *different-level tk-knowledge*: using **induction operator** according to the given **table of empirical data** (TED) as learning tk-knowledge the **database of *authentic* knowledge quanta** (BtkZ) is synthesized. Then using **deduction operator** according to the observed (*input*) tk-knowledge of DMO, the searched ***identification*** or ***prognostic*** decisions are derived on the basis of BtkZ in a form of *resulting* tk-khowledge.

Operating method of decision derivation is based on the computer manipulation of vector-matrix structures (unlike the existing methods), that allows to abbreviate the time for BtkZ synthesis as a conclusive rule and to increase the efficiency of computer decision-making.

Bibliography

[Sirodzha, 2002] Sirodzha, I.B. Quantovye modeli I metody iskusstvennogo intellekta dlya prinyatiya reshenij I upravleniya. (Quantum models and methods of artificial intelligence for decision-making and management). Naukova dumka. – Kyiv: 2002. – 420 pp.

[Sirodzha,1992] Sirodzha, I.B. Matematicheskoe I programmnoe obespechenie intellektualnykh compiuternykh sistem. (Mathematical provision and programming software of intellectual computer systems.) – Kharkiv: KhAI,1992.

Authors' Information

Liudmyla Molodykh *– a post-graduate student of Computer System Software Department, National Aerospace University named after N.I. Zhuckovsky "Kharkov Aviation Institute"; room 518, Impulse Building, Chkalova st., 17, Kharkiv, Ukraine, 61070; e-mail: molodykh@onet.com.ua; flamelia@mail.ru*

Igor B. Sirodzha *– Professor, Doctor of Technical Sciences, Head of Computer System Software Department, National Aerospace University named after N.I. Zhuckovsky "Kharkov Aviation Institute"; room 414, Impulse Building, Chkalova st., 17, Kharkiv, Ukraine, 61070.*

CONSTRUCTING OF A CONSENSUS OF SEVERAL EXPERTS STATEMENTS*

Gennadiy Lbov, Maxim Gerasimov

Abstract: *Let Γ be a population of elements or objects concerned by the problem of recognition. By assumption, some experts give probabilistic predictions of unknown belonging classes γ of objects $a \in \Gamma$, being already aware of their description $X(a)$. In this paper, we present a method of aggregating sets of individual statements into a collective one using distances / similarities between multidimensional sets in heterogeneous feature space.*

Keywords: *pattern recognition, distance between experts statements, consensus.*

ACM Classification Keywords: *I.2.6. Artificial Intelligence - knowledge acquisition.*

Introduction

We assume that $X(a) = (X_1(a),...,X_j(a),...,X_n(a))$, where the set X may simultaneously contain qualitative and quantitative features X_j, $j = \overline{1,n}$. Let D_j be the domain of the feature X_j, $j = \overline{1,n}$. The feature space is given by the product set $D = \prod_{j=1}^{n} D_j$. In this paper, we consider statements S^i, $i = \overline{1,M}$; represented as sentences of type "if $X(a) \in E^i$, then the object a belongs to the γ-th pattern with probability p^i", where $\gamma \in \{1,...,k\}$, $E^i = \prod_{j=1}^{n} E_j^i$, $E_j^i \subseteq D_j$, $E_j^i = [\alpha_j^i, \beta_j^i]$ if X_j is a quantitative feature, E_j^i is a finite subset of feature values if X_j is a nominal feature. By assumption, each statement S^i has its own weight w^i. Such a value is like a measure of "assurance".

Without loss of generality, we can limit our discussion to the case of two classes, $k = 2$.

Distances between Multidimensional Sets

In the works [1, 2] we proposed a method to measure the distances between sets (e.g., E^1 and E^2) in heterogeneous feature space. Consider some modification of this method. By definition, put

$\rho(E^1, E^2) = \sum_{j=1}^{n} k_j \rho_j(E_j^1, E_j^2)$ or $\rho(E^1, E^2) = \sqrt{\sum_{j=1}^{n} k_j (\rho_j(E_j^1, E_j^2))^2}$,

where $0 \le k_j \le 1$, $\sum_{j=1}^{n} k_j = 1$.

Values $\rho_j(E_j^1, E_j^2)$ are given by: $\rho_j(E_j^1, E_j^2) = \dfrac{|E_j^1 \Delta E_j^2|}{|D_j|}$ if X_j is a nominal feature,

$\rho_j(E_j^1, E_j^2) = \dfrac{r_j^{12} + \theta |E_j^1 \Delta E_j^2|}{|D_j|}$ if X_j is a quantitative feature, where $r_j^{12} = \left| \dfrac{\alpha_j^1 + \beta_j^1}{2} - \dfrac{\alpha_j^2 + \beta_j^2}{2} \right|$.

It can be proved that the triangle inequality is fulfilled if and only if $0 \le \theta \le 1/2$.

The proposed measure ρ satisfies the requirements of distance there may be.

Consider the set $\Omega_{(1)} = \{S_{(1)}^1,...,S_{(1)}^{m_1}\}$, where $S_{(1)}^u$ is a statement concerned to the first pattern class, $u = \overline{1,m_1}$. Let E^u be the relative sets to statements $S_{(1)}^u$, $E^u \subseteq D$, $u = \overline{1,m_1}$. By analogy, determine $\Omega_{(2)} = \{S_{(2)}^1,...,S_{(2)}^{m_2}\}$, $S_{(2)}^v$, $\widetilde{E}^v$ as before, but for the second class.

* The work was supported by the RFBR under Grant N04-01-00858.

By definition, put $k_j = \dfrac{\tau_j}{\sum_{i=1}^{n} \tau_i}$, where $\tau_j = \sum_{u=1}^{m_1} \sum_{v=1}^{m_2} \rho_j(E_j^u, \widetilde{E}_j^v)$, $j = \overline{1,n}$.

Consensus

We first treat single expert's statements concerned to a certain pattern class: let Ω be a set of such statements, $\Omega = \{S^1,...,S^m\}$, E^i be the relative set to a statement S^i, $i = \overline{1,m}$.

Denote by $E^{i_1 i_2} := E^{i_1} \oplus E^{i_2} = \prod_{j=1}^{n} (E_j^{i_1} \oplus E_j^{i_2})$, where $E_j^{i_1} \oplus E_j^{i_2}$ is the Cartesian join of feature values $E_j^{i_1}$ and $E_j^{i_2}$ for feature X_j and is defined as follows.

When X_j is a nominal feature, $E_j^{i_1} \oplus E_j^{i_2}$ is the union: $E_j^{i_1} \oplus E_j^{i_2} = E_j^{i_1} \bigcup E_j^{i_2}$.

When X_j is a quantitative feature, $E_j^{i_1} \oplus E_j^{i_2}$ is a minimal closed interval such that $E_j^{i_1} \bigcup E_j^{i_2} \subseteq E_j^{i_1} \oplus E_j^{i_2}$.

Denote by $r^{i_1 i_2} := d(E^{i_1 i_2}, E^{i_1} \bigcup E^{i_2})$.

The value $d(E,F)$ is defined as follows: $d(E,F) = \max\limits_{E' \subseteq E \setminus F} \min\limits_{j \| E'_j| \neq |E_j|} \dfrac{k_j \mid E'_j \mid}{diam(E)}$, where E' is any subset such that its projection on subspace of quantitative features is a convex set.

By definition, put $I_1 = \{\{1\},...,\{m\}\}$, ..., $I_q = \{\{i_1,...,i_q\} \mid r^{i_u i_v} < \varepsilon \quad \forall u,v = \overline{1,q}\}$, where ε is a threshold decided by the user, $q = \overline{2,Q}$; $Q \le m$.

Take any set $J_q = \{i_1,...,i_q\}$ of indices such that $J_q \in I_q$ and $J_q \not\subset J_{q+1} \quad \forall J_{q+1} \in I_{q+1}$.

Now, we can aggregate the statements S^{i_1}, ..., S^{i_q} into the statement S^{J_q}:

S^{J_q} = "if $X(a) \in E^{J_q}$, then the object a belongs to the γ-th pattern with probability p^{J_q}", where

$E^{J_q} = E^{i_1} \oplus ... \oplus E^{i_q}$, $p^{J_q} = \dfrac{\sum_{i \in J_q} c^{iJ_q} w^i p^i}{\sum_{i \in J_q} c^{iJ_q} w^i}$, $c^{iJ_q} = 1 - \rho(E^i, E^{J_q})$.

By definition, put to the statement S^{J_q} the weight $w^{J_q} = \left(1 - d(E^{J_q}, \bigcup_{i \in J_q} E^i)\right) \dfrac{\sum_{i \in J_q} c^{iJ_q} w^i}{\sum_{i \in J_q} c^{iJ_q}}$.

The procedure of forming a consensus of single expert's statements consists in aggregating into statements S^{J_q} for all J_q under previous conditions, $q = \overline{1,Q}$.

After coordinating each expert's statements separately, we can construct an agreement of several independent experts for each pattern class. The procedure is as above, except the weights: $w^{J_q} = \sum_{i \in J_q} c^{iJ_q} w^i$.

Solution of Disagreements

After constructing of a consensus for each pattern, we must make decision rule in the case of contradictory statements. Take any sets $E_{(1)}^u$ and $E_{(2)}^v$ such that $E_{(1)}^u \bigcap E_{(2)}^v = E^{uv} \neq \varnothing$, where the set $E_{(\gamma)}^u$ corresponds to a statement $S_{(\gamma)}^u$ from the experts agreement concerned to the γ-th pattern class, $\gamma = 1,2$.

Consider the sets $I_{(\gamma)}^{uv} = \left\{i \mid (S^i \in \Omega_{(\gamma)}) \text{ and } (\rho(E^i, E^{uv}) < \varepsilon^*)\right\}$, where ε^* is a threshold, $0 < \varepsilon^* < 1$.

By definition, put $p^{uv}_{(\gamma)} = \dfrac{\sum_{i \in I^{uv}_{(\gamma)}} (1 - \rho(E^i_{(\gamma)}, E^{uv}))w^i p^i}{\sum_{i \in I^{uv}_{\{\gamma)}} (1 - \rho(E^i_{\{\gamma)}, E^{uv}))w^i}$. Denote by $\gamma^* := \arg\max_{\gamma}(p^{uv}_{(\gamma)})$.

Thus, we can make decision statement:

$S^{uv} =$" if $X(a) \in E^{uv}$, then the object a belongs to the γ^*-th pattern with probability $p^{uv}_{(\gamma^*)}$"

with the weight $w^{uv} = \left| \dfrac{\sum_{i \in I^{uv}_{(1)}} (1 - \rho(E^i_{(1)}, E^{uv}))w^i - \sum_{i \in I^{uv}_{(2)}} (1 - \rho(E^i_{(2)}, E^{uv}))w^i}{\sum_{i \in I^{uv}_{(\gamma^*)}} (1 - \rho(E^i_{(\gamma^*)}, E^{uv}))} \right|$.

Bibliography

[1] G.S.Lbov, M.K.Gerasimov. Determining of distance between logical statements in forecasting problems. In: Artificial Intelligence, 2'2004 [in Russian]. Institute of Artificial Intelligence, Ukraine.

[2] G.S.Lbov, V.B.Berikov. Decision functions stability in pattern recognition and heterogeneous data analysis [in Russian]. Institute of Mathematics, Novosibirsk, 2005.

Authors' Information

Gennadiy Lbov *– Institute of Mathematics, SB RAS, Koptyug St., bl.4, Novosibirsk, Russia; e-mail: lbov@math.nsc.ru*

Maxim Gerasimov *– Institute of Mathematics, SB RAS, Koptyug St., bl.4, Novosibirsk, Russia, e-mail: max_post@mail.ru*

ANALYSIS AND COORDINATION OF EXPERT STATEMENTS IN THE PROBLEMS OF INTELLECTUAL INFORMATION SEARCH[1]

Gennadiy Lbov, Nikolai Dolozov, Pavel Maslov

Abstract*: The paper is devoted to the matter of information presented in a natural language search. The method using the statements agreement process is added to the known existing system. It allows the formation of an ordered list of answers to the inquiry in the form of quotations from the documents.*

Keywords*: Search engine, natural language, coordination of statements, semantic graph*

ACM Classification Keywords*: I.2.7 Computing Methodologies - Text analysis*

Introduction

Efficiency of the search engine is determined by the use of various methods of relevant documents revealing and insignificant ones eliminating, as well as methods peculiar to the specific search engine or their certain kind (for example, specialized search engines). Existing search engines are based on the oversight of index databases of the processed documents. The purpose is revealing the objects satisfying some criteria. However, such systems do not analyze the sentences of the document for revealing their structure and interrelations.

In the paper an approach to the search engine construction based on the analysis of semantic structure of sentences and their interrelations in the document is offered. Such method allows to do the search considering the logic of sentences thus taking into account the sense of a document. Generally it provides a stricter criterion

[1] This work was financially supported by RFBF-04-01-00858

of significant documents selection, based on accordance to a certain logic structure reflecting the sense of inquiry.

The main issue solved by the offered algorithm consists in doing the logic analysis of sentences for the subsequent search, i.e. in formation of the ranged list of answers to the inquiry in the form of quotations from documents instead of the list of these documents. Intellectuality of this method lies in its simplification of sentences perception and analysis by a person.

This system was developed as a superstructure over an existing search engine ISS2 (Internal Search System) [1]. However, independent functioning of the offered system, for example, for doing the analysis in some interesting documents is also possible. The purpose is in providing search service on local and public network catalogues being storehouses of the information. For the effective search within several storehouses there is an option for aggregation of several search servers to a distributed system. The software contains the means of carrying out a safe remote management as well as all components status analysis done by a search engine.

Selection of Search System

To derive sentences structure the system uses working results a natural text translation system [2]. It describes the methods of translated documents processing for "natural" translation considering specific features of languages. In [5] various systems of parse such as «Dialing»: L. Gershenzon, T. Kobzareva, D. Pankratov, A. Sokirko, I. Nozhov (www.aot.ru); the program of scientific group FtiPL (Institute of linguistics) RGGU (T.Yu. Kobzareva, D.G. Lakhuti, I. Nozhov); LinkParser (www.link.cs.cmu.edu/link). The selection of basis for the developed method was stipulated among other things by a good description and demonstration of system abilities [2]. In this system the analysis is done through several steps, which simplified sequence is as follows: primary, morphological, parse and semantic. Each step uses the results achieved on the previous one. The purpose of primary analysis is in the analysis of the initial document which identifies its sentences, paragraphs, notes, stable statements, electronic addresses etc. As a result the table consisting of some fragments of the initial text and their descriptors is formed. At the following step words morphoanalysis and lemmatization is done, that is each word becomes respectfully attributed with its normal form, morphological part of speech and the set of grammems, defining its grammatical gender, number, case etc. In parse syntactic groups characterized by certain parameters (type of a group, position, parental group) are defined. On the step of the semantic analysis semantic relations describing certain binary links between dependent and operating members are formed. These binary relations are just used in the offered algorithm. Resulting semantic graph characterizes interrelated binary links in the initial text sentences which reflect their logic.

For the solution of the search issue the agreement of statements described in [3] is required on a certain step. So far the resulting sets of relations in the initial text are determined by multiple expert statements whereas in the inquiry text the are defined by a set of certainly true and agreed statements. The algorithm is offered for cases with one or several experts. At first the algorithm agrees the statements of one expert which leads to a number of formulas, and then a process of overall agreement of already agreed opinions of each expert is accomplished. The specific feature if this algorithm is that he identifies absolutely all regularities. Therefore the paper [4] describes an approach to reduction of dimension statements set given on sample with the purpose of the maximal reduction of its dimension at the minimal loss of information.

Co-ordination

Basing on the intermediate results of the system work [2], which are the semantic graphs of the sentences, the logic form is constructed for each sentence. This form is a model in the language of predicates calculus of two variables united in conjunctions. Each of such predicates is an elementary statement. The following problem is to accomplish the procedure of statements agreement in the models on the base of these received models of sentences in the text and inquiry. To do it the predicates of one type are isolated and their set (for each type of predicates) corresponding to the sentences the text is a set of agreed statements whereas their set corresponding to the inquiry is an agreed in advance statement. Considering that each predicate is a part of a sentence model, the crossing of the sets corresponding to agreed predicates of different types is taken. This crossing can be considered as the result of search in the document.

Hypotheses

For the further description of algorithm it is necessary to introduce the following assumptions:

1. Sentences having different predicate structures and different variables in them are considered as the facts of different types supplementing each other.
2. A sentence with the same predicates and with the same (i.e. synonymous) variables are considered supplementing each other, therefore one-type variables are designated by the same letter with an identical index.
3. In case of crossing variables from different predicates we obtain more complicated variant of sense addition.

Each semantic link in the graph defines some type of a two variables predicate. Let's designate with letters Xi, Yi, Zi etc. each predicate variable. As the predicate defines the relation between its variables, the sets of the one-type variables standing in a certain position in the predicate are designated by the same letter with different indexes. Variables in predicates crossing are respectively designated by the same letter with an identical index. Predicates are designated by the name of semantic links. Synonymous words standing in identical positions and in identical predicates are designated by the same variables.

Analysis and Co-ordination

For the sentences and inquiry agreement, inquiry predicates are considered separately. The predicates are picked out one by one from the inquiry and in the same time the predicates of respective types are picked out from the text sentences. Expert statements are agreed with the elementary inquiry predicate which is considered to be agreed in advance.

Decision of the Formulated Task Requires Some Modification of the Algorithm Offered in [3]

Let some statement with known characteristics requires to define its belonging to the certain image. The predicate sets corresponding one or another image are considered separately. The general formal writing of a sentence is done in the form of two-place predicates conjunction. The area of predicate is defined by nominal variables satisfying the list of admissible values. We shall designate T_{ji}^{k} the truthful areas of function and argument variables in the initial sentences inquiry, where i, j, k are the numbers of predicates, statements and the links between argument and function variables, respectively.

As variables are nominal the area of true statements is defined by variables satisfying the list of admissible values. Such list has to be based on a synonyms dictionary. Besides the lists of synonyms it is also necessary for such a dictionary to contain also factors of words affinity. For example, each word from a synonymic group corresponds to the list of synonyms with decreasing weights. To simplify the finding of truthful area it is possible to define the truthfulness of statement on the base of variables satisfying the list consisting of one admissible value. But this list can be expanded with synonyms. Aprioristic probabilities of statements are equal to 1/n (S), where n is a number of statements S.

In the offered system it is enough to accomplish the agreement at a level of one expert as for simplification the analysis is done only in one document, not between many documents. Since predicates are two-placed and variables in them are from different truthful areas, then for the agreement of one expert statement it is necessary to consider separately variables in predicates. Assuming that the statement obtained from the inquiry is true and agreed we define truthful areas from each predicates included in it. The further procedure is done for each separate predicate. T_{pi}^{1} is a truthful area of the first variable in the predicate i the inquiry p. T_{pi}^{2} is the same for the second variable. The order of choice of the first and the second (the function and the argument) variable can be interchanged for altering the character of agreement, but the choice of the second variable in a predicate as the main one is more logical. Lets designate T_{ji}^{1}, T_{ji}^{2} truthful areas of variables in predicates of the initial text. Respectively, the statement satisfying:

1. $m(T_{ji}^{2} \wedge T_{pi}^{2}) \geq \beta_{r1}$ and $m(T_{ji}^{1} \wedge T_{pi}^{1}) \geq \beta_{r2}$ is true,
2. $m(T_{ji}^{2} \wedge T_{pi}^{2}) \geq \beta_{r1}$ and $\neg\, m(T_{ji}^{1} \wedge T_{pi}^{1}) \geq \beta_{r2}$ is not likely
3. $\neg\, m(T_{ji}^{2} \wedge T_{pi}^{2}) \geq \beta_{r1}$ and $\neg\, m(T_{ji}^{1} \wedge T_{pi}^{1}) \geq \beta_{r2}$ is denying
4. $\neg\, m(T_{ji}^{2} \wedge T_{pi}^{2}) \geq \beta_{r1}$ and $m(T_{ji}^{1} \wedge T_{pi}^{1}) \geq \beta_{r2}$ is denying at a choice of the second variable as the main, and not likely in other case. β_{r2} is a parameter.

Thus we receive sets of statements: ω_1 - not likely, ω_2 - true, Ω - denying.

The following steps of the one expert statements agreement are similar to described in [3].

Ranging

Let's designate Nsi the number of all predicates in a sentence, Nsoi the number of agreed predicates of a sentence, Nr the number of predicates in an inquiry. Then for determination of the sentences relevance we have to calculate the ratio:

$$k = \frac{(N_{so_i})^2}{N_{s_i} \cdot N_r}$$

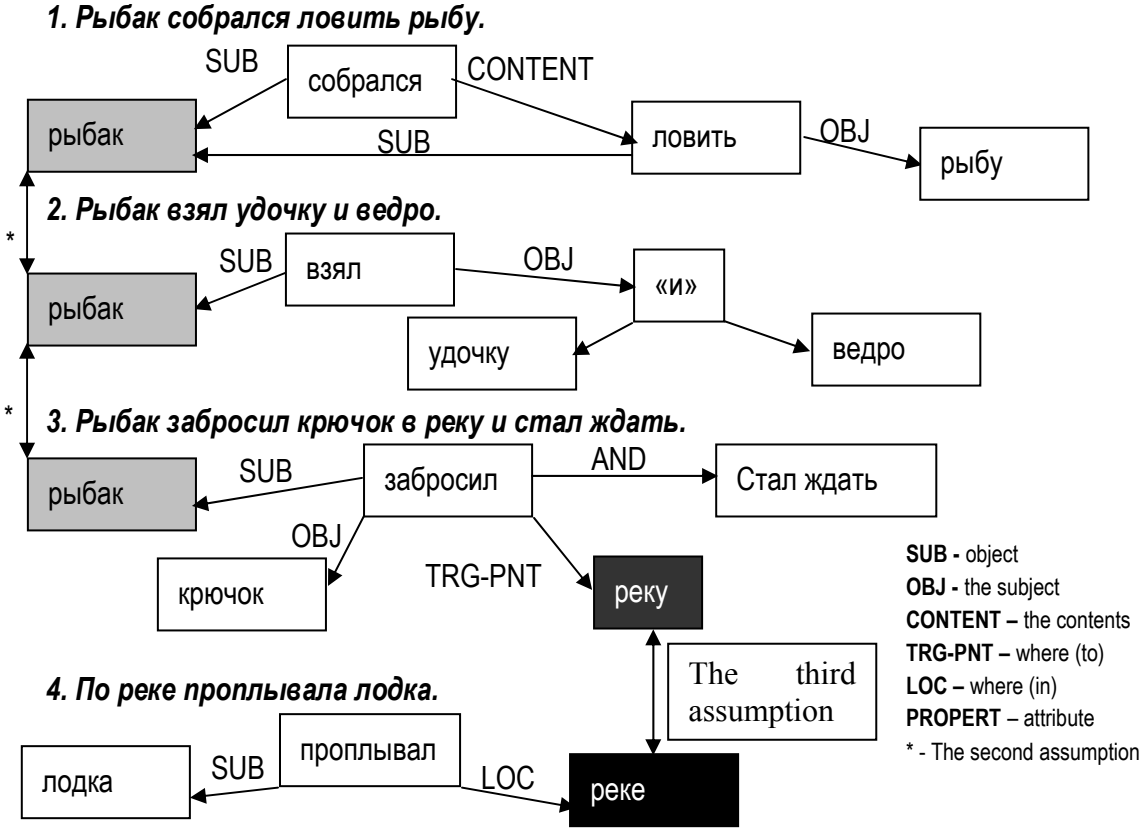
As a result we receive a set of agreed statements for the first type of predicates. The procedure of agreement is repeated separately for all other predicates and we obtain the sets of agreed statements of different type, each of which defines the sentence. Finding the crossing of all these sets we receive the set of sentences satisfying to the inquiry. The outcoming set forms the result in a usual language considering text paragraphs and document headings. Thus the trial algorithm of significant sentences allocation in the text is obtained; it reflects the first and the second assumption about the usual language.

Example (in Russian)

The simple text: **Рыбак собрался ловить рыбу. Рыбак взял удочку и ведро. Рыбак забросил крючок в реку и стал ждать. По реке проплывала лодка.**

And simple inquiries: **1. Рыбак ловит рыбу. 2. Рыбак взял наживку. 3. Мокрый рыбак.**

The sentence graphs constructed by the system [1] look as follows:



Sentences in the text:

The formula of the 1st sentence: $SUB(z_1, x_1) \cup OBJ(z_1, y_1)$

The formula of the 2nd sentence: $SUB(z_2, x_1) \cup OBJ(z_2, y_2) \cup OBJ(z_2, y_3)$

The formula of the 3rd sentence: $SUB(z_3, x_1) \cup OBJ(z_3, y_4) \cup TP(z_3, t_1) \cup SUB(z_4, x_1)$

The formula of the 4th sentence: $SUB(z_5, x_2) \cup LOC(z_5, l_1)$

Sentences of the inquiry:

The formula of the 1st sentence: $SUB(z_1, x_1) \cup OBJ(z_1, y_1)$

The formula of the 2nd sentence: $SUB(z_2, x_1) \cup OBJ(z_2, y_5)$

The formula of the 3rd sentence: $PRT(x_1, p_1)$

Conclusion

For the inquiry 1 the structure of inquiry and predicate variables are similar to one of the text sentences, therefore at least one sentence is in complete agreement with such inquiry. In the second inquiry there the structure is concurrent, variables in a predicate are distinct - the full agreement is not present, therefore the ranging will show only 25%, whereas a simple phrase «рыбак взял» will show 100%. The third inquiry contains the single predicate PRT designating the property of an object. Such predicate is not present in the text, therefore the algorithm agrees nothing. In other words, the sense of inquiry is not crossed with the sense of the text.

Bibliography

[1] P.P. Maslov. Designing Materials of the All-Russian scientific conference of young scientists in 7 parts. Novosibirsk: NGTU, 2006. Part 1. - 291 p. // pp. 250-251

[2] Automated text processing “DIALING” // www.aot.ru

[3] G.S. Lbov T.I. Luchsheva. The analysis and the coordination of expert’s knowledge in problems of recognition // 2’2004, NAS of Ukraine, pp. 109-112.

[5] Nozhov I. The Parse // http://www.computerra.ru/offline/2002/446/18250/

Authors' Information

Gennadiy Lbov - *SBRAS, The head of laboratory; P.O.Box: 630090, Novosibirsk, 4 Acad. Koptyug avenue, Russia; e-mail: lbov@math.nsc.ru*

Nikolai Dolozov - *NSTU, The senior lecturer, Cand.Tech.Sci.; P.O.Box: 6300092, Novosibirsk, 20 Marks avenue, Russia; e-mail: dnl@interface.nsk.su*

Pavel Maslov - *NSTU, The post-graduate student of of FAMI; P.O.Box: 6300092, Novosibirsk, 20 Marks avenue, Russia; e-mail: altermann@ngs.ru*

RECOGNITION OF THE HETEROGENEOUS MULTIVARIATE VARIABLE[1]

Tatyana Stupina

Abstract*: An application of the multivariate prediction method to solving the problem pattern recognition with respect to the sample size is considered in this paper. The criterion of multivariate heterogeneous variable recognition is used in this approach. The relation of this criterion with probability of error is shown. For the fixed complexities of probability distribution and logical decision function class the examples of pattern recognition problem are presented.*

Keywords*: the prediction of multivariate heterogeneous variable, the pattern recognition, the complexity of distribution.*

[1] This work was financially supported by RFBR-04-01-00858

Introduction

The reducing relation problem with respect to sample is one of important problem in data mining. The quality of sample decision function depends on the size of the sample, the complexity of the distributions, and the complexity of the class of functions used by the algorithm for constructing sample decision functions. When the distribution is known the quality of decision function, for example, is risk function for one prediction variable. For the pattern recognition problem, for example, it is well-known probability of error. When the distribution is unknown the problem estimating of this quality with respect to the complexity of the distributions, of the functions class and sample size is appeared. At present time there are approaches solving this problem [Vapnik V.N., Chervonenkis A.Ya, 1970]. In addition the complexity of the functions class is assigned differently [Lbov G.S., Starceva N.G, 1999]. But that approaches consider the case of one prediction variable and one variable type.

However there are many important applied when we what to predict or recognize several heterogeneous variables. In work [Lbov G.S., Stupina T.A., 2002] was presented this problem statement. It is necessary to construct the sample decision function on the small sample in the multivariate heterogeneous space, so the most proper class is a class of logical decision functions [Lbov G.S., Starceva N.G, 1999]. In this paper for the fixed probability distribution the relation of the criterion with probability of error is shown for pattern recognition problem.

Problem Statement

In the probabilistic statement of the problem, the value *(x,y)* is a realization of a multidimensional random variable (*X,Y*) on a probability space $< \Omega, \mathrm{B}, \mathrm{P} >$, where $\Omega = D_X \times D_Y$ is μ-measurable set (by Lebeg), B is the borel σ-algebra of subsets of Ω, P is the probability measure (we will define such as *c*, the strategy of nature) on B, D_X is heterogeneous domain of under review variable, $\dim D_x = n$, D_y is heterogeneous domain of objective variable, $\dim D_y = m$. The given variables can be of arbitrary types (quantitative, ordinal, nominal). For the pattern recognition problem the variable *Y* is nominal. Let us put $\Phi_\circ$ is a given class of decision functions. Class $\Phi_\circ$ is μ-measurable functions that puts some subset of the objective variable $E_y \subseteq D_Y$ to each value of the under review variable $x \in D_X$, i.e. $\Phi_\circ = \{f : D_x \to 2^{D_y}\}$. For example the domain E_y can contain the several patterns. The quality $F(c,f)$ of a decision function $f \in \Phi_\circ$ under a fixed strategy of nature *c* is determined as $F(c,f) = \int_{D_x} (P(E_y(x)/x) - \mu(E_y(x)))dP(x)$, where $E_y(x) = f(x)$ is a value of decision functions in *x*, $P(y \in E_y(x)/x)$ is a conditional probability of event $\{y \in E_y\}$ under a fixed *x*, $\mu(E_y(x))$ is measurable of subset E_y. Note that if $\mu(E_y(x))$ is probability measure, than criterion $F(c,f)$ is distance between distributions. If the specified probability coincides with equal distribution than such prediction does not give no information on predicted variable (entropy is maximum). The measure $\mu(E_y(x)) = \frac{\mu(E_y)}{\mu(D_Y)} = \prod_{j=1}^{m} \frac{\mu(E_{y_j})}{\mu(D_{y_j})}$ is the normalized measure of subset E_y and it is introduced with taking into account the type of the variable. The measure $\mu(E_y(x))$ is measure of interval, if we have a variable with ordered set of values and it is quantum of set, if we have a nominal variable (it is variable with finite non-ordering set of values and we have the pattern recognition problem). Clearly, the prediction quality is higher for those E_y whose measure is smaller (accuracy is higher) and the conditional probability $P(y \in E_y(x)/x)$ (certainty) is larger. For a fixed strategy of nature *c*, we define an optimal decision function $f_\circ(x)$ as such that $F(c,f_\circ) = \sup_{f \in \Phi_\circ} F(c,f)$, where $\Phi_\circ$ is represented above class of decision functions. If the strategy of nature is unknown the sampling criterion $F(\bar{f})$ is used. When we solve this problem in practice the size of sample is very smaller and type of variables different. In this case is used class of logical decision function Φ_m complexity *M* [Lbov G.S., Starceva N.G, 1999].

Properties of the Criterion

For the fixed strategy of nature *c* the relation of the criterion $F(c,f)$ with probability of error P_f is shown.

Statement 1. For any strategy of nature *c* the quality criterion $F(c,f)$ is represented by risk function such that $1-R(c,f)=\int\limits_{D_X}\int\limits_{D_Y}(1-L(y,f(x)))p(x,y)dxdy$, where the loss function $L(y,f)$ such as $L(y,f)=\begin{cases}\mu(E_y), & y\in E_y\\ 1+\mu(E_y), & y\notin E_y\end{cases}$.

Remark that risk function $R(c,f)$ is probability of error P_f if the loss function $L(y,f)$ is indicator function.

Statement 2. For recognition *k* patterns by decision function *f* the quality criterion is $F(c,f)=\frac{k-1}{k}-P_f$.

Consequence 1. For recognition two patterns we have equation $F(c,f)=1/2-P_f$.

Consequence 2. For pattern recognition problem the optimal decision function coincides with bayes function such as $\sup\limits_{x\in(-\infty,+\infty)} F(c,f)=\inf\limits_{x\in(-\infty,+\infty)} P_f$.

Definition 1. Define a nature strategy c_M (generated by logical decision function $f\in\Phi_M$, $f\sim<\alpha,r(\alpha)>$) such as $c_M=\{p^t(x,y)=p_x^t p_{y/x}^t=P(x\in E_X^t)P(y\in E_Y^t/x\in E_X^t),t=1,...,M\}$, where 1) $\sum_{t=1}^M p_x^t=1$; 2) $P(E_Y^t/E_X^t)=p_{y/x}^t$, 3) $P(\bar{E}_Y^t/E_X^t)=1-p_{y/x}^t$, where $E_X^t\in\alpha$, $E_Y^t\in r(\alpha)$, $<\alpha,r(\alpha)>\in\Phi_M$, 4) $\forall\ A_X\subseteq E_X^t\ P(A_X)=p_x^t\frac{\mu(A_X)}{\mu(E_X^t)}$, $\forall\ A_Y\subseteq E_Y^t\ P(A_Y/E_X^t)=p_{y/x}^t\frac{\mu(A_X)}{\mu(E_Y^t)}$.

In the paper [Lbov G.S., Stupina T.A., 2002] is proved that $F(c,f)=\sum_{t=1}^M p_x^t(p_{y/x}^t-\mu(E_y^t))$ for this nature strategy. Let for *k* pattern recognition the domain D_Y is the set $\{\omega_1,...,\omega_k\}$.

Statement 3. Let the nature strategy c_k for *k* pattern recognition is generated by logical decision function f^* such as $f^*(x)=E_y^i$ for $x\in E_X^i$, then the probability of error P_f for decision function *f* such as $f(x)=\omega_i$, $\omega_i\in E_y^i$, for $x\in E_X^i$ is $P_f=1-\sum\limits_{i=1}^k\frac{1}{k\mu(E_y^i)}p_x^i p_{y/x}^i$.

Consequence 3. From the statement 3 it follows equation $P_f+F(c_k,f^*)=1+\sum\limits_{i=1}^k p_x^i\left[p_{y/x}^i\left(1-\frac{1}{k\mu(E_y^i)}\right)-\mu(E_y^i)\right]$.

Let us illustrate these statements. Let there is *n=1, m=1*, *X*- continuous variable, *Y*-nominal variable, *M*=2. The nature strategy c_2 is generated by f^*, that $\alpha^*=\{E_X^1,E_X^2\}$, $E_X^1=(0.364,\ 1.0]$, $E_X^2=[0.0,\ 0.364]$, $r(\alpha^*)=\{E_y^1,E_y^2\}$, $E_y^1=\{1\}$, $E_y^2=\{1,0\}$, $\omega_1=$'1', $\omega_2=$'0', $p_x^1=\frac{19}{50}$, $p_x^2=\frac{31}{50}$, $p_{y/x}^1=0{,}95$, $p_{y/x}^2=1$. So we have $F(c_2,f^*)=0{,}171$. Obviously that c_2 is such that conditional distribution $p(x/\{1\})$ and $p(x/\{0\})$ is intersected for every pattern, if we have f' ($f'(x)=\{1\}$, if $x\in E_X^1$, and $f'(x)=\{0\}$, if $x\in E_X^2$) or f'' ($f''(x)=\{1\}$, if $x\in E_X^1$, and $f''(x)=\{1\}$, if $x\in E_X^2$). Let calculate $P_{f'}$ using definition and compare with criterion $F(c_2,f^*)$: $P_{f'}=P(\{1\})P(E_X^2/\{1\})+P(\{0\})P(E_X^1/\{0\})=P(E_X^2)P(\{1\}/E_X^2)+P(E_X^1)P(\{0\}/E_X^1)$ $=\frac{31}{50}\cdot\frac{1}{2}\cdot 1+\frac{19}{50}\cdot(1-0{,}95)=0{,}329$. Similarly we can provide the probability of error $P_{f''}$. For this case we have $F(c_2,f^*)=\frac{1}{2}-P_{f'}=0{,}5-0{,}329=0{,}171$ (statement 2 and 3).

Conclusion

An approach to solving the problem of heterogeneous multivariate variable recognition with respect to the sample size was considered in this paper. The solution of this problem was assigned by means of presented criterion. The universality of the logical decision function class with respect to presented criterion makes the possible to introduce a measure of distribution complexity and solve this problem for small sample size. For the nature strategy and the class of logical decision function the criterions properties are presented by means of statements and consequences for pattern recognition problem.

Bibliography

[Lbov G.S., Starceva N.G, 1999] Lbov G.S., Starceva N.G. Logical Decision Functions and Questions of Statistical Stability. Inst. Of Mathematics, Novosibirsk.

[Lbov G.S., Stupina T.A., 2002] Lbov G.S., Stupina T.A. Performance criterion of prediction multivariate decision function. Proc. of international conference "Artificial Intelligence", Alushta, pp.172-179.

[Vapnik V.N., Chervonenkis A.Ya, 1970] Vapnik V.N., Chervonenkis A.Ya. Theory of Pattern Recognition, Moscow: Nauka.

Author's Information

Tatyana A. Stupina *– Institute of Mathematics SBRAS, Koptuga 4 St, Novosibirsk, 630090, Russia; e-mail: stupina@math.nsc.ru*

ON THE QUALITY OF DECISION FUNCTIONS IN PATTERN RECOGNITION

Vladimir Berikov

Abstract*: The problem of decision functions quality in pattern recognition is considered. An overview of the approaches to the solution of this problem is given. Within the Bayesian framework, we suggest an approach based on interval estimates on a finite set of events.*

Keywords*: Bayesian learning theory, generalization ability.*

Introduction

The problem of decision functions quality consists in need to find a decision function, not too distinguishing from the optimal decision function in the given family, provided that the probability distribution is unknown, and learning sample has limited size. Under optimal decision function we shall understand such function for which the risk (the expected losses of wrong forecasting for a new object) is minimal. In particular, the following questions should be solved at the analysis of the problem.

a) With what conditions the problem has a decision?

b) How the quality of decision function can be evaluated most exactly on learning sample?

c) What principles should be followed at the choice of decision function (in other words, what properties must possess a class of decision functions and learning algorithm) under the given sample size, dimensionality of variable space and other information on the task?

The principle possibility to decide the delivered problem is motivated by the following considerations. Firstly, for the majority of real tasks of forecasting on statistical information it is possible to expect a priori the existence of certain more or less stable mechanism being the basis of under study phenomena. Secondly, it is possible to expect that available empirical information in one or another degrees reflects the functioning of this unknown mechanism. Thirdly, it is required for the successful solution that the class of decision functions and learning algorithm should possess certain characteristics, on which will be said below.

Basic Approaches

A number of different approaches to the solution of the problem can be formulated. In experimental approaches [1] (one-hold-out, bootstrap, cross-validation) the data set is divided on learning sample (for decision function finding) and test sample (for evaluation of quality of the decision function) repeatedly. The performance of the given method for decision function construction is then evaluated as the average quality on test samples. The shortcoming of this approach is its computational expensiveness.

Probabilistic approach is based on the preliminary estimation of the distribution law. A learning problem can be solved if this law can be reconstructed from the empirical data. The given approach can be used only when sufficiently large a priori information on the distribution is available. For instance, it is possible to show that the problem of distribution density reconstruction from the empirical data is in general an ill-posed problem [2].

Two directions within the framework of this approach are known. The first one deals with the asymptotic properties of learning algorithms. In [3], the asymptotic evaluations of quality for such methods as a K-nearest neighbor rule, minimum Euclidean distance classifier, Fisher's linear discriminant etc are given.

The second direction takes into account the fact that the size of learning sample is limited. This approach uses the principles of multivariate statistical analysis [4] and restricts the set of probabilistic models for each class, for instance, assumes the Gauss distribution low. The determined types of decision functions (for instance, linear or quadratic; perceptron) are considered.

Vapnik and Chervonenkis [2] suggested an alternative approach to the solution of the given problem ("statistical learning theory"). This approach is distribution-free and can be applied to arbitrary types of decision functions. The main question is "when minimization of empirical risk leads to minimization of true unknown risk for arbitrary distribution?". The authors associate this question and the question of existence of the uniform convergence of frequencies to probabilities on the set of events related to the class of decision functions. The fundamental notions of growing function, entropy, VC-dimension that characterize the difficulty of decision functions class are suggested. It is proved that the frequency converges to probability uniformly if and only if the amount of entropy per element of sample converges to zero at the increase of sample length. As far as these evaluations are received for the worst case, on the one hand they are distribution-independent, but on the other hand give too pessimistic results. In [5] the notion of efficient VC-dimension is offered, which is dependent from distribution. With this notion, the authors managed to perfect greatly the accuracy of evaluations.

Within the framework of statistical learning theory the structured risk minimization method was suggested. The idea of the method consists in consequent consideration of classes of decision functions, ranked on growth of their complexity. The function minimizing empirical risk in the corresponding class and simultaneously giving the best value for guaranteed risk is chosen. Support vectors machine [6] uses an implicit transformation of data to the space of high dimensionality by means of the given kernel function. In this space, a hyperplane maximizing the margin between support vectors of different classes is found. It is proved that the main factor influences the risk is not the dimensionality of the space but margin width.

In "PAC-learning" approach [7,8] ("Probably Approximately Correct"; developed within the framework of computational learning theory) the complexity of learning algorithm is taken into consideration. It is required that learning algorithm, with probability not smaller than η finding the decision function for which the probability of mistake does not exceed ε, has time of work polynomially depended on sample size, complexity of class of decision functions, and on values $1/\eta$, $1/\varepsilon$. The existence of such algorithms for some classes of recognition decision functions is proved, for instance, for conjunctions of Boolean predicates, linear decision functions, some types of neural networks, decision trees.

Statistical and computational learning theories suggest the worst-case analysis. From standpoints of statistical decision theory, their evaluations are of minimax type. However it is possible to use the average-case analysis (in Bayesian learning theory) for which it is necessary to define certain priory distribution (either on the set of distribution parameters or on the set of decision functions) and to find the evaluations at the average [9,10]. The main task is to find the decision function for which a posterior probability of error is minimal. As a rule, the finding of such function is computationally expensive, so the following rule of thumb can be used. Instead of optimum decision function search, less expensive function which is close (under determined conditions) to optimum is found. An example is minimum description length principle (minimizing the sum of the code length describing the function and the code length describing the data misclassified by this function). Another example is maximum a posterior probability function. From the other hand, the estimations can be done by statistical modeling (Markov Chain Monte Carlo method).

The main problem at the motivation of the Bayesian approach is a problem of choice of a priori distribution. In the absence of a priori information, it is possible to follow Laplace principle of uncertainty, according to which uniform a priori distribution is assumed. If the uncertainty in the determining of a priory distribution presents, the robust Bayesian methods can be applied.

Bayesian learning theory was used for discrete recognition problem [10], for decision trees learning algorithms [11] etc. Within the Bayesian framework, the case of one discrete variable is mostly suitable for analytical calculations. Hughes [10] received the expression for the expected probability of recognition error depending on sample size and the number of values of the variable. It was shown that for the given sample size, an optimum number of values exists for which the expected probability of error takes minimum value. Lbov and Startseva [12] received the expressions for the expected misclassification probability for the case of available additional knowledge about the probability of mistake for the optimum Bayes decision function. In [13-15] this approach was generalized. Below we give the summary of the obtained results.

Bayesian Interval Estimates of Recognition Quality on Finite Set of Events

a) The functional dependencies are obtained between the quality of an arbitrary method of decision functions construction and learning sample size, number of events [14,15].
b) The theoretical investigation of empirical risk minimization method is done [14,15].
c) The posterior estimates of recognition quality for the given decision function are found (with respect to number of events, empirical risk, sample size) [13].
d) New quality criteria for logical decision functions are suggested on the basis of above mentioned results. An efficient method for classification tree construction is proposed [15].
e) New methods are suggested for the following data mining tasks: regression analysis, cluster analysis, multidimensional heterogeneous time series analysis, rare events forecasting [15].

Acknowledgements

This work was supported by the Russian Foundation of Basic Research, grant 04-01-00858a

Bibliography

[1] Breiman L. Bagging predictors // Mach. Learn. 1996. V. 24. P. 123-140.
[2] Vapnik V. Estimation of dependencies based on empirical data. Springer- Verlag. 1982.
[3] Fukunaga K. Introduction to statistical pattern recognition. Academic Press, NY and London. 1972.
[4] Raudys S. Statistical and Neural Classifiers: An integrated approach to design. London: Springer-Verl., 2001.
[5] Vapnik V., Levin E. and Le Cun Y. Measuring the VC-Dimension of a Learning Machine // Neural Computation, Vol. 6, N 5, 1994. pp. 851--876.
[6] Vapnik V.N. An Overview of Statistical Learning Theory // IEEE Transactions on Neural Networks. 1999. V.10, N 5. P.988-999.
[7] Valiant L.G. A Theory of the Learnable, CACM, 17(11):1134-1142, 1984.
[8] Haussler D. Probably approximately correct learning // Proc. 0f the 8th National Conference on Artificial Intelligence. Morgan Kaufmann, 1990. pp. 1101-1108.
[9] D.Haussler, M.Kearns, and R.Schapire. Bounds on sample complexity of Bayesian learning using information theory and the VC dimension // Machine Learning, N 14, 1994. pp. 84-114.
[10] Hughes G.F. On the mean accuracy of statistical pattern recognizers // IEEE Trans. Inform. Theory. 1968. V. IT-14, N 1. P. 55-63.
[11] W. Buntine. Learning classification trees // Statistics and Computing. 1992. V. 2. P. 63--73.
[12] Lbov, G.S., Startseva, N.G., *About statistical robustness of decision functions in pattern recognition problems.* Pattern Recognition and Image Analysis, 1994. Vol 4. No.3. pp.97-106.
[13] Berikov V.B., Litvinenko A.G. *The influence of prior knowledge on the expected performance of a classifier.* Pattern Recognition Letters, Vol. 24/15, 2003, pp. 2537-2548.
[14] Berikov, V.B. A Priori Estimates of Recognition Accuracy for a Small Training Sample Size // Computational Mathematics and Mathematical Physics, Vol. 43, No. 9, 2003. pp. 1377- 1386
[15] Lbov G.S., Berikov V.B. Stability of decision functions in problems of pattern recognition and heterogeneous information analysis. Inst. of mathematics Press, Novosibirsk. 2005. (in Russian)

Author's Information

Vladimir Berikov *– Sobolev Institute of Mathematics SD RAS, Koptyug pr.4, Novosibirsk, Russia, 630090; e-mail: berikov@math.nsc.ru*

К ВОПРОСУ О РАССТОЯНИЯХ НА ВЫСКАЗЫВАНИЯХ ЭКСПЕРТОВ И МЕРЕ ОПРОВЕРЖИМОСТИ (ИНФОРМАТИВНОСТИ) ВЫСКАЗЫВАНИЙ ЭКСПЕРТОВ НА КЛАССАХ МОДЕЛЕЙ ТЕОРИЙ

Александр Викентьев

***Аннотация**: При анализе знаний, заданных в виде высказываний экспертов, для различия содержащейся в них информации и группирования их по похожести, возникает необходимость введения расстояния между высказываниями экспертов и меры опровержимости (информативности) высказываний экспертов. Этой проблемой занимались Загоруйко Н.Г., Лбов Г.С., Викентьев А.А. [1-4]. Вводим расстояние не на всем множестве моделей, а на моделях некоторой, заранее фиксированной теории Г. Такой подход является естественным при изучении некоторой конкретной прикладной проблемы, (поскольку тогда расстояние и информативность не будут искажены моделями, не относящимися к изучаемой области) заданной например, некоторыми знаниями о ней, далее - теорией. Работа проделана в рамках проекта РФФИ 04-01-00858а.*

***Ключевые слова:** базы знаний,высказывания экспертов, теория моделй, метрика.*

***ACM Classification Keywords**: I.2.6. Artificial Intelligence - knowledge acquisition.*

Введение

Мы фиксируем теорию Г, суть, набор таких высказываний, например, с которыми согласились все эксперты. Возможно, что теория Г может сформулирована на языке более высокого порядка и рассматриваются только те модели на которых выполнеы все аксиомы Г. Пусть $S(\Sigma) = \{v_1,...,v_n\}$ - набор элементарных высказываний. Теорией Г назовем набор формул (- гипотез) $\{\varphi_1,...,\varphi_k\}$ - высказываний экспертов, с которыми все эксперты согласны. Предполагается, что теория Г удовлетворяет следующим требованиям:

1) непротиворечивости (совместности);

2) замкнутости относительно выводимости (это требование не обязательно, но для полноты можно считать, что эксперты могут доказывать формулы с помощью гипотез);

Расстояние на высказываниях экспертов и его свойства

Пусть База Знаний Σ состоит из формул исчисления высказываний.

Определение 1. Множество элементарных высказываний $S(\Sigma) = \{v_1,...,v_n\}$, используемых для написания высказываний из Σ, назовем носителем совокупности знаний. Рассматриваем $P(S(\Sigma))$-множество всевозможных подмножеств $S(\Sigma)$, его элементы, суть наборы $\{v_i \mid i \in I\}$, где $I \subseteq \{1,...,n\}$ истинностных значений элементарных высказываний, называем моделями. Мощность множества моделей исчисления высказываний равна $|P(S(\Sigma))| = 2^{|S(\Sigma)|}$.

Обозначим через $Mod\Gamma = Mod_{S(\Sigma)}\Gamma = \{M \in P(S(\Sigma)) \mid M \models \Gamma\}$ все модели теории Г. Множество моделей из ModГ на которых формула А – истинна, обозначим через $Mod_\Gamma(A)$.

Теорема 1. Для логической теории Г справедливы следующие свойства расстояния:

1) $0 \le \rho_\Gamma(\varphi,\psi) \le 1$;

2) $\rho_\Gamma(\varphi,\psi) = \rho_\Gamma(\psi,\varphi)$;

3) если $\varphi \equiv_\Gamma \psi$,то $\rho_\Gamma(\varphi,\psi)=0$;

4) $\rho_\Gamma(\varphi,\psi)=1 \Leftrightarrow \varphi \equiv_\Gamma \neg\psi$;

5) если $\varphi \equiv_\Gamma \varphi_1$ и $\psi \equiv_\Gamma \psi_1$,то $\rho_\Gamma(\varphi,\psi)=\rho_\Gamma(\varphi_1,\psi_1)$;

6) $\rho_\Gamma(\varphi,\psi)=1-\rho_\Gamma(\varphi,\neg\psi)=\rho_\Gamma(\neg\varphi,\neg\psi)$;

7) $\rho_\Gamma(\varphi,\psi)\le\rho_\Gamma(\varphi,\chi)+\rho_\Gamma(\chi,\psi)$;

8) $\rho_\Gamma(\varphi,\neg\varphi)=\rho_\Gamma(\varphi,\psi)+\rho_\Gamma(\psi,\neg\varphi)$;

9) $\rho_\Gamma(\varphi,\psi)=\rho_\Gamma((\varphi\ \&\ \psi),(\varphi\vee\psi))$.

Доказательство аналогично [4] с использованием моделей теории Г.

Мера опровержимости (информативности)высказываний экспертов

Мера опровержимости (информативности) высказываний.

Определение 2. Мерой опровержимости высказывания φ назовем относительное число моделей теории Г на которых высказывание ложно. Для высказываний совместных с теорией определим меру информативности на множестве ModГ, как меру опровержимости высказывания φ .

$$\mu_\Gamma(\varphi)=\rho_\Gamma(\varphi,\Gamma)=\frac{|\mathrm{Mod}_\Gamma(\varphi\ \&\ \neg\Gamma)\bigcup\mathrm{Mod}_\Gamma(\neg\varphi\ \&\ \Gamma)|}{|\mathrm{Mod}\Gamma|}=$$

$$=\frac{|\mathrm{Mod}_\Gamma(\neg\varphi\ \&\ \Gamma)|}{|\mathrm{Mod}\Gamma|}=\frac{|\mathrm{Mod}_\Gamma(\neg\varphi)|}{|\mathrm{Mod}\Gamma|}$$

Теорема 2. Для логической теории Г справедливы следующие свойства меры опровержимости (информативности для формул, имеющих модель теории Г) высказываний:

1) $\mathbf{0\le\mu_\Gamma(\varphi)\le 1}$;

2) $\mathbf{\mu_\Gamma(\neg\varphi)=1-\mu_\Gamma(\varphi)}$;

3) $\mu_\Gamma(\varphi)\le\mu_\Gamma(\varphi\ \&\ \psi)$;

4) $\mathbf{\mu_\Gamma(\varphi\vee\psi)\le\mu_\Gamma(\varphi)}$;

5) если $\mathbf{\rho_\Gamma(\varphi,\psi)=1}$, то $\mathbf{\mu_\Gamma(\varphi\ \&\ \psi)=1}$ и $\mathbf{\mu_\Gamma(\varphi\vee\psi)=0}$;

6) если $\mathbf{\rho_\Gamma(\varphi,\psi)=0}$, то $\mathbf{\mu_\Gamma(\varphi)=\mu_\Gamma(\psi)=\mu_\Gamma(\varphi\ \&\ \psi)=\mu_\Gamma(\varphi\vee\psi)}$;

7) $\mathbf{\mu_\Gamma(\varphi\ \&\ \psi)=\rho_\Gamma(\varphi,\psi)+\mu_\Gamma(\varphi\vee\psi)}$;

8) $\min\{\mu_\Gamma(\varphi),\mu_\Gamma(\psi)\}\ge\mu_\Gamma(\varphi\vee\psi)\ge\max\{\mu_\Gamma(\varphi),\mu_\Gamma(\psi)\}-\rho_\Gamma(\varphi,\psi)$
и $\min\{\mu_\Gamma(\varphi),\mu_\Gamma(\psi)\}+\rho_\Gamma(\varphi,\psi)\ge\mu_\Gamma(\varphi\ \&\ \psi)\ge\max\{\mu_\Gamma(\varphi),\mu_\Gamma(\psi)\}$

9) если $\mathbf{\mu_\Gamma(\varphi)=\mu_\Gamma(\varphi_1)}$, $\mathbf{\mu_\Gamma(\psi)=\mu_\Gamma(\psi_1)}$ и $\mathbf{\rho_\Gamma(\varphi,\psi)\le\rho_\Gamma(\varphi_1,\psi_1)}$,
то $\mathbf{\mu_\Gamma(\varphi\ \&\ \psi)\le\mu_\Gamma(\varphi_1\ \&\ \psi_1)}$;

10) если $\mathbf{\mu_\Gamma(\varphi)\le\mu_\Gamma(\varphi_1)}$, $\mathbf{\mu_\Gamma(\psi)=\mu_\Gamma(\psi_1)}$ и $\mathbf{\rho_\Gamma(\varphi,\psi)=\rho_\Gamma(\varphi_1,\psi_1)}$,
то $\mathbf{\mu_\Gamma(\varphi\ \&\ \psi)\le\mu_\Gamma(\varphi_1\ \&\ \psi_1)}$;

11) $\mu_\Gamma(\varphi\ \&\ \psi)=\dfrac{\mu_\Gamma(\varphi)+\mu_\Gamma(\psi)+\rho_\Gamma(\varphi,\psi)}{2}$;

12) $\mu_\Gamma(\varphi\vee\psi)=\dfrac{\mu_\Gamma(\varphi)+\mu_\Gamma(\psi)-\rho_\Gamma(\varphi,\psi)}{2}$.

Доказательство аналогично [3, 4, 1] с использованием теоремы 1.

Заключение

Имеет место обобщение этих результатов на теории первого порядка, как и выше с аналогичными обобщениями расстояния и меры опровержимости для предложений и формул первого порядка со свободными переменными. Результаты планируется использовать для кластизации данных знаний.

Список литературы

1. **Г.С.Лбов, Н.Г.Старцева.** Логические решающие функции и вопросы статистической устойчивости решений. Новосибирск: Издательство Института математики, 1999. С. 85-102.
2. **Н.Г.Загоруйко, М.В.Бушуев.** Меры расстояния в пространстве знаний // Анализ данных в экспертных системах. Новосибирск,1986. Выпуск 117:Вычислительные системы. С.24-35.
3. **А.А.Викентьев, Г.С.Лбов.** О метризации булевой алгебры предложений и информативности высказываний экспертов // Доклад РАН 1998.Т.361, №2 С.174-176.
4. **A.A.Vikentiev, G.S.Lbov.** Setting the metric and informativeness on statements of experts // Pattern Recognition and Image Analysis. 1997 V.7, N2, P.175-183.
5. **Г.Кейслер, Ч.Ч.Чэн.** Теория моделей. Москва:Мир,1977.
6. **Ю.Л.Ершов, Е.А.Палютин.** Математическая логика. Санкт-Петербург, 2004.

Информация об авторе

Александр А. Викентьев – доцент, канд.физ-мат. Наук, Институт математики СО РАН, лаборатория анализа данных; e-mail: vikent@math.nsc.ru

THE MEASURE REFUTATION, METRICS ON STATEMENTS OF EXPERTS (LOGICAL FORMULAS) ON A CLASS MODELS SOME THEORY[1]

Alexander Vikent'ev

Abstract. The paper discusses a logical expert statements represented as the formulas with probabilities of the first order language consistent with some theory T. Theoretical-models methods for setting metrics on such statements are offered. Properties of metrics are investigated. The research allows solve problems of the best reconciliation of expert statements, constructions of decision functions in pattern recognition, creations the bases of knowledge and development of expert systems.

***Keywords**: pattern recognition, distance between experts statements.*

***ACM Classification Keywords**: I.2.6. Artificial Intelligence - Knowledge Acquisition.*

Introduction

As the increasing interest to the analysis of the expert information given as probabilities logic statements of several experts is now shown, questions on knowledge of the experts submitted by formulas of the first order language with probabilities are interesting also. With the help of suitable procedure the statement of experts it is possible to write down as formulas of Sentence Logic or formulas of the first order language. Clearly, the various statements of experts (and the formulas appropriate to them) carry in themselves different quantity of the information. To estimate and analyses this information it is necessary to define the degree of affinity of

[1] This work was financially supported by the Russian Foundation for Basic Research, project no. 04-01-00858a.

statements that allows to estimate a measure of refutation statements of experts (a measure of refutation above at formula of the first order language with smaller number of elements satisfying it) and to specify their probabilities (an average share probabilities realizations for formulas with variables). It allows solve problems of the best reconciliation of expert statements, constructions of decision functions in pattern recognition, creation of bases of knowledge and expert systems [1].

A number of natural metrics on probabilities knowledge of experts is offered with use of suitable class of models (with metrics) some theory and modifications symmetric difference, by analogue, par example [4] for logical Lbov's the predicat for unique model. Properties of these metrics, connected to them measures of refutation of formulas (distance from the formula up to class of equivalence of identically true formula) and probability are established. From the point of view of importance of the information presented by an expert, it is natural to assume that a measure of refutation of the formula (nonempty predicate) the above, than it is les measure of elements satisfying it (i.e. a measure determined on subsets, set by predicate formulas).

We introduce the measure of refutation similarly to a case of formulas without probabilities. We call

$$R\left(P\left(\overline{x}\right)\right)=\rho_{вер}\left(P\left(\overline{x}\right),\ 1\right)$$

the measure of refutation of formula $P\left(\overline{x}\right)$, where 1 is an identical true predicate, that is, $\overline{x}=\overline{x}$. All stated for predicates (and Lbov's predicate aussi without probability) fairly and for formulas of the first order language with probabilities.

The distance between the formulas of Sentence Logic is entered in [1], properties of the entered distance are given and proved in the same place. Ways of introduction of distance between the formulas of the first order language are offered in [2]. Measures of refutation and probabilities of formulas are entered and their properties are formulated in [3]. The distance between the formulas of Sentence Logic with probabilities is entered in [3,5]. In the given work the way of introduction of distances between probabilities statements of experts represented as the formulas of the first order language theory T with probabilities is offered.

Distance between Statements of Experts Represented as the Formulas of the First Order Language with Probabilities in Theory

Let experts speak about probabilities of predicates on the product $\prod_{j=1}^{p} D_{x_j}$.

Then the given by expert probability is interpreted as follows: "the knowledge" $B_l^i =< P_l^i(x_1,...,x_p), p_l^i >$ means, that the predicate $P_l^i(x_1,...,x_p)$ is true on $n_{P_l^i}=\left\lfloor n\cdot p_l^i \right\rfloor$ trains of length p in model M_i, where $n=\prod_{j=1}^{k}\left|D_{x_j}\right|$ - measure of model.

Let's find distance between predicates P_l and P_j. For this purpose all over again we shall calculate distance $\rho^i(B_l^i, B_j^i)$ between probabilities interpretations $B_l^i =< P_l^i(\overline{x}), p_l^i >$ and $B_j^i =< P_j^i(\overline{x}), p_j^i >$ of predicates P_l and P_j in each model M_i. Distances are calculated between predicates of identical district and from the same variables plus measure protjaga p.e. [4], and without in stable theory.

Then interpreting the probabilities given by experts the described above way we receive that the predicate $P_l^i(\overline{x})$ is true on $n_{P_l^i}$ trains in model M_i and the predicate $P_j^i(\overline{x})$ is true on $n_{P_j^i}$ trains in model M_i theory T. We shall note that is not known on what trains each predicate true and number (or mera) of trains on which these predicates are simultaneously true.

We shall consider the following task. Let the predicate $P_l^i(\overline{x})$ is true on $n_{P_l^i}$ trains in model M_i and the predicate $P_j^i(\overline{x})$ is true on $n_{P_j^i}$ trains in model M_i and k^i - number of trains on which these predicates are simultaneously true. It is required to calculate distance between $B_l^i =< P_l^i(\overline{x}), p_l^i >$ and $B_j^i =< P_j^i(\overline{x}), p_j^i >$. Distances arising in further we shall designate through $\rho_{k^i}(B_l^i, B_j^i)$, where, $k^i = t, t+1, \ldots, \min(n_{P_l^i}, n_{P_j^i})$, $t = \max(0, n_{P_l^i} + n_{P_j^i} - n)$ hereinafter.

Distance $\rho_{k^i}(B_l^i, B_j^i)$ we shall define as a modifies (as ask above) symmetric difference, i.e.

$$\rho_{k^i}(B_l^i, B_j^i) = \frac{1}{n}(n_{P_l^i} + n_{P_j^i} - 2k^i), \tag{1}$$

for every one $k^i = t, t+1, \ldots, \min(n_{P_l^i}, n_{P_j^i})$. All properties of distances formulated in [1] are fair for $\rho_{k^i}(B_l^i, B_j^i)$. Let's offer some ways of calculation distance $\rho^i(B_l^i, B_j^i)$ between probabilities interpretations $B_l^i =< P_l^i(\overline{x}), p_l^i >$ and $B_j^i =< P_j^i(\overline{x}), p_j^i >$ of predicates P_l and P_j in each model M_i theory $\underline{T}$. If the number k^i is not known (the number of trains on which these predicates are simultaneously true in model M_i) and if there are no preferences for value k^i (preference can be stated by experts) it is possible to act as follows. We shall assume, that all values for number k^i are equally probability. Then distance between probabilities interpretations $B_l^i =< P_l^i(\overline{x}), p_l^i >$ and $B_j^i =< P_j^i(\overline{x}), p_j^i >$ of predicates P_l and P_j in model M_i we shall define as average of distances $\rho_{k^i}(B_l^i, B_j^i)$ on all values k^i, i.e.

$$\rho(B_l^i, B_j^i) = \frac{\sum_{k^i=t}^{\min(n_{P_l^i}, n_{P_j^i})} \rho_{k^i}(B_l^i, B_j^i)}{\min(n_{P_l^i}, n_{P_j^i}) + 1 - t}. \tag{2}$$

For this distance all properties of distances formulated in [1] also are executed.

If by experts it is stated what value for k^i is more preferable in quality $\rho^i(B_l^i, B_j^i)$ it undertakes $\rho_{k^i}(B_l^i, B_j^i)$, i.e.

$$\rho(B_l^i, B_j^i) = \rho_{k^i}(B_l^i, B_j^i). \tag{3}$$

In the offered formulas (1) – (3) of distances the kind of formulas between which the distance is calculated is not taken into account. Therefore it is natural to offer distance by which takes into account a kind of formulas. Applying the model approach [1-3] to elements of set $\{M_i\}_{i=1}^s$ we shall find probabilities $P_{M_i}(P_j^i)$, $P_{M_i}(P_j^i)$ ([3]) and distance $\rho_{M_i}(P_l^i, P_j^i)$ in model M_i ([2]), then we shall calculate probability $P_{M_i}(P_l^i \wedge P_j^i) = \frac{1}{2}(P_{M_i}(P_l^i) + P_{M_i}(P_j^i) - \rho(P_l^i, P_j^i))$ ([3]) and we shall find $k_0^i = [P_{M_i}(P_l^i \wedge P_j^i) \cdot n]$ - the number of trains on which predicates are simultaneously true. Having k_0^i (calculated on models), it is possible to reduce number of possible values for k^i. Three cases here are possible:

1) if $t < k_0^i < \min(n_{P_l^i}, n_{P_j^i})$, then $k^i = k_0^i - 1, k_0^i, k_0^i + 1$;

2) if $k_0^i = t$ or $k_0^i = \min(n_{P_l^i}, n_{P_j^i})$, then, for example, $k^i = k_0^i, k_0^i + 1$; or $k^i = k_0^i - 1, k_0^i$;

3) if $k_0^i < t$ or $k_0^i > \min(n_{P_l^i}, n_{P_j^i})$, then $k^i = t$ or $k^i = \min(n_{P_l^i}, n_{P_j^i})$.

And already to these values for k^i apples offered above formulas (1) - (3) of distances. As required probably some expansion of number of values for k^i.

The offered ways it is possible to calculate distance between the following statements: $B_l^i =< P_l^i(\overline{x}), p_l^i >$ - the information received from one expert, and $B_l^j =< P_l^j(\overline{x}), p_l^j >$ - the information received from other expert. Thus we have calculated distance $\rho^i(B_l^i, B_j^i)$ between probabilities interpretations $B_l^i =< P_l^i(\overline{x}), p_l^i >$ and $B_j^i =< P_j^i(\overline{x}), p_j^i >$ of predicates P_l and P_j and degree elongate in each model M_i theory T. Then as distance $\rho_{вер}(P_l, P_j)$ between predicates P_l and P_j we shall take size

$$\rho_{вер}(P_l, P_j) = \frac{1}{s}\sum_{i=1}^{s} \rho^i(B_l^i, B_j^i).$$

For all properties of distance formulated in ([5]) are carried out for $\rho_{вер}(P_l, P_j)$.

Acknowledgements

This work was financially supported by the Russian Foundation for Basic Research, project no. 04-01-00858a.

Bibliography

[1] Lbov G.S., Startseva N.G. Decision Logical Functions and Statistical Robustness. Novosibirsk: Izd. Inst. Math., 1999.

[2] Vikent'ev A.A., Koreneva L.N. "Setting the metric and measures of informativity in predicate formulas corresponding to the statements of experts about hierarchical objects", *Pattern Recognition and Image Analysis*, V. 10, N. 3, (2000), 303--308.

[3] Vikent'ev A.A., Koreneva L.N. "Model approach to probabilities expert statements", "*Mathematical Methods for Pattern Recognition – 10", Moscow,* (2001), 25-28.

[4] G.S.Lbov, M.K.Gerasimov. Determining the distance between logical statements in forecasting problems. In: Artificial Intelligence, 2'2004 [in Russian]. Institute of Artificial Intelligence, Ukraine.

[5] Викентьев А.А., Лбов Г.С., Коренева Л.Н. "Расстояние между вероятностными высказываниями экспертов", *Искусственный интеллект, 2'2002, НАН Украины,* 58-64.

[6] Keisler G., Chang C. Model theory. M.: Mir, 1977.

Author's Information

Alexander Vikent'ev – *Institute of Mathematics, SB RAS, Acad. Koptyuga St., bl.4, Novosibirsk, Russia; e-mail: vikent@math.nsc.ru*

DECOMPOSITION OF BOOLEAN FUNCTIONS BY LOOKING FOR TRACKS OF A GOOD SOLUTION

Arkadij Zakrevskij

***Abstract**: The problem of two-block Boolean function decomposition, generally non-disjunctive, is regarded in case when a good solution does exist. A new heuristic combinatorial algorithm is offered, optimized on speed. The problem is reduced mainly to finding an appropriate weak partition on the set of arguments of the considered Boolean function, which should be decomposable at this partition. At first the randomized search for "tracks" of the appropriate partition is fulfilled. The recognized tracks are represented by some "triads" - the simplest weak partitions corresponding to non-trivial decompositions. After that the whole sought-for partition is restored from the discovered track. The results of computer experiments testify the high practical efficiency of the algorithm.*

***Keywords**: Boolean function, non-disjunctive decomposition, appropriate partition, combinatorial search, recognition, randomization, computer experiment.*

Introduction

The well-known problem of Boolean functions decomposition consists in possible replacement of the considered function by an equivalent composition of several functions with smaller number of variables. Different sorts of the decomposition were offered, including *sequential two-block decomposition*, called simple decomposition, disjunctive or non-disjunctive.

As a result of the disjunctive decomposition the initial Boolean function $f(\boldsymbol{x})$ is substituted by a composition $g(h(\boldsymbol{u}), \boldsymbol{v})$ at the partition $\boldsymbol{u}/\boldsymbol{v}$ on the set of arguments $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$: $\boldsymbol{u} \cup \boldsymbol{v} = \boldsymbol{x}$, $\boldsymbol{u} \cap \boldsymbol{v} = \varnothing$. In this case the partition $\boldsymbol{u}/\boldsymbol{v}$ is named strong) [1, 2]. The more general case is represented by non-disjunctive decomposition, when function $f(\boldsymbol{x})$ is substituted by the composition $g(h(\boldsymbol{u}, \boldsymbol{w}), \boldsymbol{w}, \boldsymbol{v})$ at the weak partition $\boldsymbol{u}/\boldsymbol{v}$, where $\boldsymbol{x} = \boldsymbol{u} \cup \boldsymbol{w} \cup \boldsymbol{v}$, $\boldsymbol{u} \cap \boldsymbol{w} = \boldsymbol{u} \cap \boldsymbol{v} = \boldsymbol{w} \cap \boldsymbol{v} = \varnothing$ [3, 4]. In both cases the conditions $|\boldsymbol{u}| > 1$ and $|\boldsymbol{v}| > 0$ must be satisfied, otherwise the decomposition is trivial – it exists always.

Solving the following two tasks are laying in the basis of decomposition methods.

The task 1. For a given function $f(\boldsymbol{x})$ and partition $\boldsymbol{u}/\boldsymbol{v}$ (strong either weak) to find out, whether $f(\boldsymbol{x})$ is decomposable at $\boldsymbol{u}/\boldsymbol{v}$, i. e. whether there exists a non-trivial composition ($g(h(\boldsymbol{u}), \boldsymbol{v})$ or $g(h(\boldsymbol{u}, \boldsymbol{w}), \boldsymbol{w}, \boldsymbol{v})$) equivalent to function f (and, maybe, to find functions g and h).

If such a composition does exist, we shall term partition $\boldsymbol{u}/\boldsymbol{v}$ as *appropriate*.

The task 2. For a given function $f(\boldsymbol{x})$ to find an appropriate partition.

Solving Task 1

The task of checking a Boolean function $f(\boldsymbol{x})$ for decomposability at a given partition $\boldsymbol{u}/\boldsymbol{v}$ on the set of arguments $\boldsymbol{x}$ was regarded in [1-3], where a necessary and sufficient condition for that was found, which could be formulated as in the following assertion:

Assertion 1. Let $f_i(\boldsymbol{u}, \boldsymbol{v})$ be the coefficients of the Shannon disjunctive decomposition of Boolean function $f(\boldsymbol{x})$ by variables of set $\boldsymbol{w}$. Then function $f(\boldsymbol{x})$ is decomposable at partition $\boldsymbol{u}/\boldsymbol{v}$, if and only if the coefficients of the similar decomposition of each function $f_i(\boldsymbol{u}, \boldsymbol{v})$ by variables from set $\boldsymbol{u}$ accept not more than two different values.

Checking Boolean functions for satisfying this condition was laid into the base of many known methods offered for solving task 1.

A new algorithm for solving the same problem is developed, based on representation of considered functions by Boolean vectors and using efficient component-wise operations above them. It is checking a Boolean function for satisfying a certain condition formulated in terms of symmetry operations introduced in [4] and defined as follows:

$S_{xi}^{\vee} f(\boldsymbol{x}) = f(x_1, \ldots, x_i, \ldots, x_n) \vee f(x_1, \ldots, \neg x_i, \ldots, x_n)$ is disjunctive symmetrization of Boolean function $f(\boldsymbol{x})$ by argument x_i ,

$S_{\boldsymbol{u}}^{\vee} f(\boldsymbol{x}) = S_{u1}^{\vee}(S_{u2}^{\vee} \ldots (S_{uk}^{\vee} f(\boldsymbol{x}))\ldots)$ is disjunctive symmetrization of Boolean function $f(\boldsymbol{x})$ over the given subset $\boldsymbol{u} = (u_1, u_2, \ldots, u_k)$ of set $\boldsymbol{x}$.

In result of applying operator $S_{\boldsymbol{u}}^{\vee}$ to Boolean space of variables from set $\boldsymbol{x}$, where function $f(\boldsymbol{x})$ is defined, any interval with inner variables taken from set $\boldsymbol{u}$, which has if only one 1, is filled up with 1s completely.

Operator $S_{\boldsymbol{v}}^{\vee} f(\boldsymbol{x})$ is defined in the same way.

The suggested algorithm checks function $f(\boldsymbol{x})$ for satisfying the condition formulated in Assertion 1. It takes the initial coefficient $f_{\boldsymbol{u}}^0$ of Shannon decomposition of function $f(\boldsymbol{x})$ by set $\boldsymbol{u}$, then, by using operation $f^+(\boldsymbol{x}) = f(\boldsymbol{x}) \oplus f_{\boldsymbol{u}}^0$, modifies function $f(\boldsymbol{x})$ in such a way that all coefficients coinciding with $f_{\boldsymbol{u}}^0$ are changed for 0, and finally finds out if all the remaining coefficients are equal to each other.

In other words, the algorithm is working in accordance with the following assertion.

Assertion 2. A Boolean function $f(\boldsymbol{x})$ is decomposable at partition $\boldsymbol{u}/\boldsymbol{v}$, if and only if

$$(f^+(\boldsymbol{x}) \oplus S_{\boldsymbol{u}}^{\vee} f^+(\boldsymbol{x})) \wedge S_{\boldsymbol{v}}^{\vee} f^+(\boldsymbol{x}) = 0.$$

So, checking a Boolean function for decomposability at a given partition is reduced to several operations over Boolean vectors.

The second task is more difficult. A heuristic combinatorial algorithm is offered below to solve it, optimized on speed.

Relations on the Set of Partitions. Triads

Consider two partitions $\boldsymbol{u}/\boldsymbol{v}$ and $\boldsymbol{u}^*/\boldsymbol{v}^*$, such that $\boldsymbol{u}^* \subseteq \boldsymbol{u}$ and $\boldsymbol{v}^* \subseteq \boldsymbol{v}$. Let's speak, that partition $\boldsymbol{u}^*/\boldsymbol{v}^*$ submits to partition $\boldsymbol{u}/\boldsymbol{v}$. The following assertions are fair by that.

Assertion 3. If the function $f(\boldsymbol{x})$ is decomposable at partition $\boldsymbol{u}/\boldsymbol{v}$, it is decomposable as well at partition $\boldsymbol{u}^*/\boldsymbol{v}^*$. Hence, if the function $f(\boldsymbol{x})$ is not decomposable at partition $\boldsymbol{u}^*/\boldsymbol{v}^*$, it is not decomposable also at partition $\boldsymbol{u}/\boldsymbol{v}$.

Let's assume $|\boldsymbol{u}| = k$ and $|\boldsymbol{v}| = m$. Partition with $k = 2$ and $m = 1$ we shall term as a *triad*. It is the simplest of partitions, at which a nontrivial decomposition can be defined.

Assertion 4. The number of triads is equal to $C_n^2 (n-2) = \dfrac{n(n-1)(n-2)}{2}$.

Assertion 5. The Boolean function $f(\boldsymbol{x})$ is not decomposable, if it is not decomposable at any of triads.

It follows from here that the function is decomposable, if and only if it is decomposable if only at one of triads.

Let's estimate the probability of decomposability of a random Boolean function $f(\boldsymbol{x})$ of n variables. Consider some triad $\boldsymbol{u}/\boldsymbol{v}$, having put for example $\boldsymbol{u} = (a, b)$ and $\boldsymbol{v} = (c)$, and some arbitrary coefficient $\varphi(a, b, c)$ of function $f(\boldsymbol{x})$ Shannon decomposition by variables of set $\boldsymbol{w} = \boldsymbol{x} \setminus (\boldsymbol{u} \cup \boldsymbol{v})$.

Assertion 6. The number of different values of the coefficients of the Shannon decomposition of an arbitrary Boolean function $\varphi(a, b, c)$ by variables a and b does not exceed two, if the coefficients φ_0 and φ_1 of the decomposition of function φ by c obey to the condition: $\varphi_0 = \neg\varphi_1$ or at any rate one of the coefficients φ_0 and φ_1 represents constant 0 or 1.

It follows from this condition

Assertion 7. Among 256 different Boolean functions $\varphi(a, b, c)$ there are 74 functions, for which the number of different values of coefficients of Shannon decomposition by variables a and b does not exceed two.

Let's designate through γ the share of such functions ($\gamma = 74/256$).

Considering the above assertions and taking into account, that in any triad $|\boldsymbol{w}| = n - 3$ and, therefore, the number of coefficients of the function $f(\boldsymbol{x})$ Shannon decomposition by $\boldsymbol{w}$ is equal to 2^{n-3}, we shall formulate

Assertion 8. At equiprobable sampling of function $f(\boldsymbol{x})$ from the set of all Boolean functions of n variables the probability of the function $f(\boldsymbol{x})$ decomposability is bounded above by the value

$$C_n^2 (n-2) \cdot \gamma 2^{n-3}.$$

It is important to note, that this value fast tends to zero with growth of n. For example, at n = 6, 12, 18, 24 it receives accordingly values 0.003, 10^{-135}, 10^{-8827}, $10^{-565200}$.

It is obvious from this that decomposability of arbitrary Boolean functions is rather improbable. Therefore, the task of decomposition has practical sense only for such a function, which, as it must be known a priori, is decomposable, and it is necessary only to find the appropriate composition. The task in this setting is considered below.

Search for Tracks of an Appropriate Partition

Let's assume, that function $f(\boldsymbol{x})$ is known, obtained as a result of the composition $g(h(\boldsymbol{u}, \boldsymbol{w}), \boldsymbol{w}, \boldsymbol{v})$ of two random Boolean functions g and h on a given, also random, weak partition $\boldsymbol{u}/\boldsymbol{v}$ of the set of arguments $\boldsymbol{x}$. It is required to detect this composition by the analysis of function $f(\boldsymbol{x})$. This problem can be reduced mainly to finding the appropriate partition $\boldsymbol{u}/\boldsymbol{v}$.

The elementary way of finding this partition consists in exhaustive search of all nontrivial weak partitions on the set of arguments $\boldsymbol{x}$ (such, that $k > 1$ and $m > 0$) and checking the function $f(\boldsymbol{x})$ for the decomposability on everyone of these partitions. However such a way is rather time-consuming, what is evident from the following assertion.

Assertion 9. The number of all nontrivial weak partitions on the set of n arguments equals $3^n - 2^{n+1} - n2^{n-1} + n + 1$.

This estimate is obtained by deleting from the set of all three-block partitions (their number equals 3^n) such ones, which do not obey to the condition "$k > 1$ and $m > 0$".

For example, at n = 6, 12, 18, 24 this number receives accordingly following values: 416; 498,686; 384,536,924; 282,194,655,482.

Analyzing such big number of partitions one by one is practically impossible. Search for appropriate partition $\boldsymbol{u}/\boldsymbol{v}$ can be accelerated by preliminary random looking for some track of partition $\boldsymbol{u}/\boldsymbol{v}$, i. e. some partition $\boldsymbol{u}^*/\boldsymbol{v}^*$ submitting to $\boldsymbol{u}/\boldsymbol{v}$, after which the latter could be found by corresponding expansion of sets $\boldsymbol{u}^*$ and $\boldsymbol{v}^*$.

Assertion 10. The number of partitions submitting to partition $\boldsymbol{u}/\boldsymbol{v}$, is equal to 2^{k+m}. So the random search for partition $\boldsymbol{u}/\boldsymbol{v}$ can be accelerated in 2^{k+m} times.

Let's assume, that during the search q partitions are selected by random and analyzed, until a track of $\boldsymbol{u}/\boldsymbol{v}$ is found.. Suppose, that each appropriate partition submits to the required partition $\boldsymbol{u}/\boldsymbol{v}$ (another case will be analyzed a little later). At this supposition the following assertion is fair.

Assertion 11. The expectation $M(q)$ of the value q is equal approximately to $3^n / 2^{k+m}$.

For example, at n = 6, 12, 18, 24 expectation $M(q)$ receives accordingly values 11.4, 130, 1478, 16834, if $k = m = n/2$ (in that case the formula is simplified to $(3/2)^n$), and values 45; 2,076; 94,577; 4,309,504, if $k = m = n/3$.

Nevertheless, expectation $M(q)$ remains rather big, and that is why more efficient method is suggested, which is looking for tracks between the simplest weak partitions (i.e. triads), as their number is much less. For example, at the same values n = 6, 12, 18, 24 the total number of triads according to the assertion 3 receives values 60, 660, 2448, 6072.

Assertion 12. The number of triads submitting to partition $\boldsymbol{u}/\boldsymbol{v}$, is equal to $C_k^2 m$.

The next assertion follows from here.

Assertion 13. When looking track between triads the expectation $M(q)$ is equal to $C_n^2(n-2) / C_k^2 m$.

That is too far less, compared with preceding methods. For example, on the same series of values n = 6, 12, 18, 24 the expectation $M(q)$ receives accordingly values 30, 27.5, 27.2, 27.1, if $k = m = n/3$ and values 6.7, 7.3, 7.6, 7.7, if $k = m = n/2$. That testifies the essential reducing of the number of checked triads by looking for an appropriate one as well as weak dependence of the run-time on the number of variables, what is very important. Let's remark in this connection, that if $k = m = n/t$, then the ratio representing the value $M(q)$ can be approximated by constant t^3, for example, by constant 27, when t = 3, and constant 8, when t = 2.

False Tracks and Side Solutions

Let's remark, that not any appropriate triad submits to the required partition, what is illustrated by results of a conducted experiment, with parameters n = 6, k = 3, m = 3. During this experiment there were generated the random partition $\boldsymbol{u}/\boldsymbol{v}$ = (a, d, e) / (b, c, f) on the set of variables $\boldsymbol{x}$ = (a, b, c, d, e, f) and a couple of random Boolean functions $h(\boldsymbol{u})$ and $g(\boldsymbol{v})$. Then the composition $g(h(\boldsymbol{u}), \boldsymbol{v})$ was constructed and the corresponding Boolean function $f(\boldsymbol{x})$ obtained, after which all triads on the set $\boldsymbol{x}$ were checked and appropriate ones were selected between them. The discovered appropriate triads are marked by 1s in the following table of triads: for example, in the first row the triads *bd*/*a*, *be*/*a*, *cd*/*a*, *de*/*a* and *df*/*a* are marked. It appears that from 35 appropriate triads only 9 submit to partition $\boldsymbol{u}/\boldsymbol{v}$ - they are marked by the bold font. The remaining 26 triads represent tracks of some side solutions, and that is why they are called false triads below.

	a	*a*	*a*	*a*	*a*	*b*	*b*	*b*	*b*	*c*	*c*	*c*	*d*	*d*	*e*
	b	*c*	*d*	*e*	*f*	*c*	*d*	*e*	*f*	*d*	*e*	*f*	*e*	*f*	*f*
a	.	.	.	.	.	.	1	1	.	1	.	.	1	1	.
b	.	.	**1**	**1**	.	.	.	.	.	1	.	.	**1**	1	.
c	.	.	**1**	**1**	.	.	1	.	.	.	.	.	**1**	1	.
d	1	1	.	1	1	1	.	1	1	.	1	1	.	.	1
e	1	.	1	.	.	.	1	.	.	1	.	.	.	1	.
f	.	.	**1**	**1**	.	.	1	.	.	1	.	.	**1**	.	.

However, with growth of the number of variables n the quota of false triads promptly diminishes.

Let's estimate the probability of violation of the supposition "any appropriate triad submits to partition $\boldsymbol{u}/\boldsymbol{v}$" in the regarded situation of checking function $f(\boldsymbol{x})$ for satisfying the condition of decomposability at the given partition $\boldsymbol{u}/\boldsymbol{v}$ and at a random sampling of functions g and h in the composition $g(h(\boldsymbol{u}, \boldsymbol{w}), \boldsymbol{w}, \boldsymbol{v})$.

Each of three units of a current triad can be selected from one of the sets $\boldsymbol{u}$, $\boldsymbol{v}$ or $\boldsymbol{w}$. Therefore, there are $3^3 = 27$ situations, which can symbolically be presented as *uu/u, uu/v, uu/w, uv/u, uv/v, uv/w, uw/u, uw/v, uw/w, vu/u, vu/v, vu/w, vv/u, vv/v, vv/w, vw/u, vw/v, vw/w, wu/u, wu/v, wu/w, wv/u, wv/v, wv/w, ww/u, ww/v, ww/w.* Only in one of them (namely in *uu/v*) the triad submits to partition $\boldsymbol{u}/\boldsymbol{v}$, and among remaining ones the maximum probability the considered supposition gains in situation *uu/u*.

Assertion 14. The probability that a random triad in situation *uu/u* will appear to be appropriate is equal to $\gamma 2^{n-m-3}$.

For example, if n = 12 or n = 18, the probability receives accordingly values $2.0 \cdot 10^{-3}$, $1.4 \cdot 10^{-5}$ at $m = n/3$ and values $2.4 \cdot 10^{-2}$, $5.8 \cdot 10^{-4}$ at $m = n/2$.

In remaining situations (excepting uu/v) this probability is no more. It follows from here, that in practical situations (when n exceeds 10) any appropriate triad, most likely, submits to the required partition, therefore, we can relay on Assertion 11 for estimating the time needed to find a true track.

This conclusion is confirmed by results of two series of computer experiments, during which 10 random compositions for each considered set of parameters n, k, m were generated and any appearance of a false triad was registered. In the first series, where 150 experiments were carried out at 15 sets (6, 2, 2), (7, 3, 2), (8, 3, 3), …, (20, 7, 7), the false triads have appeared only at n = 6 and n = 7. In the second series (on 15 sets (6, 3, 3), (7, 4, 3), (8, 4, 4), …, (20, 10, 10)) the false triads were met at n = 6, 7, 8, 9, 10. At n = 10 only one such triad on 10 experiments has appeared.

It follows from here, that for finding required partition at $n \geq 10$ it is enough to expand properly sets $\boldsymbol{u}$ and $\boldsymbol{v}$, which constitute the detected appropriate triad.

Meanwhile, when n ≤ 10 and a good solution does exist (with $k > 2$ and $m > 1$), false tracks can be eliminated by using the following procedure of initial expansion. One by one we test variables from set $\boldsymbol{x}$, not coming in the triad, until discover such one, which inclusion into set u^* results in a new (expanded) appropriate value of partition $\boldsymbol{u}^*/\boldsymbol{v}^*$. If such a variable is not found, we eliminate that triad as false and look for another appropriate triad. When we find such variable, we try similarly to expand set $\boldsymbol{v}^*$. If the appropriate element misses, we again eliminate the triad and look for another one. If both checks lead to the positive result, the regarded triad is recognized as a true track and is accepted for the next expansion.

Let's return to reviewing the former example of the set of appropriate triads. Some of its elements are rejected by this procedure because they cannot be expanded by addition of a new element into set $\boldsymbol{u}^*$. Some other elements safely passed the first trial but not sustained the second one (when trying to expand set $\boldsymbol{v}^*$), so they are rejected also. In such a way all 26 false tracks are rejected.

Heuristic Search Algorithm

Taking into account the explained reasons, we shall offer the following heuristic algorithm of detection of a good (with $k > 2$ and $m > 1$) partition $\boldsymbol{u}/\boldsymbol{v}$, at which the considered function $f(\boldsymbol{x})$ is decomposable.

A. We consider a sequence of triads selected by random. As soon as a current triad happens to be appropriate, we apply the described above procedure of elimination of false tracks between appropriate triads. In result we find a perspective appropriate triad and regard it as the initial value of variable partition $\boldsymbol{u}^*/\boldsymbol{v}^*$.

B. We consider successively all elements of set $\boldsymbol{x} \setminus (\boldsymbol{u}^*/\boldsymbol{v}^*)$. Regarding the current element, we include it into set $\boldsymbol{u}^*$. If the obtained by that extended partition $\boldsymbol{u}^*/\boldsymbol{v}^*$ appears not to be appropriate, the element is deleted from $\boldsymbol{u}^*$ and put into $\boldsymbol{v}^*$. If the extended partition is not appropriate now, the element is deleted from $\boldsymbol{v}^*$.

C. Obtained after exhaustive search of all variables the partition $\boldsymbol{u}^*/\boldsymbol{v}^*$ is accepted as the solution. Adduced above estimations allow to affirm, that with a high probability it will coincide with the required partition $\boldsymbol{u}/\boldsymbol{v}$.

In case of large number of variables (ten and more) the algorithm can be simplified by excluding the procedure of elimination of false tracks, because the probability of side solutions becomes negligibly small.

In the case, when a complete partition $\boldsymbol{u}/\boldsymbol{v}$ does exist, at which the function $f(\boldsymbol{x})$ can be decomposed, the search of this partition can be essentially speeded up by looking only for the set $\boldsymbol{v}$, as set $\boldsymbol{u}$ can be obtained as the complement of set $\boldsymbol{v}$ up to set $\boldsymbol{x}$.

The set $\boldsymbol{v}$ can be obtained by way of expansion of set $\boldsymbol{v}^*$ in the found appropriate triad by fast iterative algorithm, which includes the current variable x_i into set $\boldsymbol{v}$, if

$$S_{xi}^{\oplus}(T \wedge S_{xi}^{\wedge}(S_u^{\vee} T)) = 0,$$

where $S_{xi}^{\oplus}$ is the operator of symmetrization a Boolean function by EXOR-operation, known as differential operator, and

$$T = S_v^{\vee} f^* = S_v^{\vee}(f(\boldsymbol{x}) \oplus f_u^0).$$

In case of weak partition, when $\boldsymbol{w} \neq \varnothing$, the redundant check of variables from set $\boldsymbol{x}\backslash\boldsymbol{v}$ for possibility of inclusion into set $\boldsymbol{u}$ will be carried out.

Results of Experiments

The offered heuristic algorithm was programmed in C++ (by I. Vasilkova) and tested on computer (Pentium IV, 2.8 GHz). In a series of experiments the parameters n, k, m were fixed, a random partition $\boldsymbol{u}/\boldsymbol{v}$ on the set $\boldsymbol{x}$ and functions g, h were generated, then function $f(\boldsymbol{x})$ was calculated. After that the above described algorithm was fulfilled, which found partition $\boldsymbol{u}/\boldsymbol{v}$ for the function $f(\boldsymbol{x})$, the number q of triads scanned by search for tracks was ascertained, and the total time t (in seconds) spent during search for the partition was measured. The obtained results are represented in the following table, which right part corresponds to splitting of the set of arguments $\boldsymbol{x}$ in three parts $\boldsymbol{u}$, $\boldsymbol{w}$ and $\boldsymbol{v}$, whenever possible the same size, and left part – in two: $\boldsymbol{u}$ and $\boldsymbol{v}$.

It should be noted that the table begins with $n = 14$, because $t \leq 0.00$, if $n \leq 14$.

n	*k/m*	*q*	*t*	*k/m*	*q*	*t*
14	7/7	3	0.00	5/5	39	0.00
15	8/7	2	0.00	5/5	78	0.02
16	8/8	1	0.00	6/5	18	0.01
17	9/8	11	0.02	6/6	34	0.05
18	9/9	4	0.02	6/6	42	0.09
19	10/9	3	0.03	7/6	39	0.17
20	10/10	9	0.11	7/7	3	0.16
21	11/10	1	0.11	7/7	3	0.39
22	11/11	3	0.48	8/7	6	2.41
23	12/11	9	2.17	8/8	16	7.05
24	12/12	3	2.48	8/8	26	18.17
25	13/12	4	5.61	9/8	19	33.16
26	13/13	4	11.64	9/9	3	52.11
27	14/13	19	60.13	9/9	36	187.55
28	14/14	19	1280.67	10/9	3	1585.03

As can be seen from the table, the regarded task of Boolean function decomposition is solved in less than one minute, when the number of arguments does not exceed 26. The amount of memory for representation of Boolean function $f(\boldsymbol{x})$ grows quickly, reaching 2^{28} = 268 435 456 bits when $n = 28$. This value is critical for the given experiment because of the restrictions on the used operation memory, and that leads to an essential decrease of the calculation speed.

Conclusion

A new heuristic algorithm for Boolean function f(x) decomposition g(h(u, w), w, v) is developed, which recognizes the weak partition u/v on the set of arguments x by finding first some track of it in the form of a triad, and using then this track for restoring the partition u/v as a whole. As computer experiments show, the algorithm is very efficient.

Acknowledgements

The work had been performed thanks to the support of ISTC (Project B-986).

Bibliography

1. Povarov G.N. About functional decomposition of Boolean functions. – Reports of the AS of USSR, 1954. – V. 4, No 5 (in Russian).
2. Ashenhurst R.L. The decomposition of switching functions. Proc. International Symposium on the Theory of Switching, Part 1. – Harvard University Press, Cambridge, 1959, pp. 75-116.
3. Curtis H.A. Design of switching circuits. – Van Nostrand, Princeton, N. J., 1962.
4. Zakrevskij A.D. Algorithm of a Boolean function decomposition. – Annals of Siberian Physical-Technical Institute, 1964. – V. 44, pp. 5-16 (in Russian).

Author's 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*

ПРИНЦИП ИНДИВИДУАЛЬНОЙ ОПТИМАЛЬНОСТИ В ИГРАХ

Сергей Мащенко

***Abstract**: A new conception of a individual-optimum equilibrium in the conditions of the complete being of players inform is offered.*

***Keywords**: Nesh's equilibria, Pareto optimum, a coalition equilibria, a individual-optimum equilibria.*

Введение

Концепции равновесия по Нешу и оптимальности по Парето, а также подходы, идейно основанные на них, несомненно, является важнейшим теоретико-игровым инструментом, который наиболее часто применяется в экономике, социологии, экологии.

Рассмотрим игру *G* в нормальной форме $(X_i, u_i; i \in N)$, где $N = \{1,2,...n\}$ - множество из *n* игроков; X_i - множество стратегий *i*-го игрока; $u_i(x)$ - функция его выигрыша, которая определена на множестве ситуаций игры $X = \prod_{i \in N} X_i$ и максимизируется.

Наиболее привлекательными концепциями оптимальности в условиях полной информированности игроков является оптимальность по Парето и по Нешу [1].

Концепция оптимальности по Парето базируется на, так называемых, сильной и слабой аксиомах Парето [2]. Говорят, что вектор выигрыша игроков $U_N(x) = (u_i(x))_{i \in N}$ в ситуации *x* игры *G* доминирует вектор выигрыша $U_N(y)$ в ситуации *y* (обозначают это через $U_N(x) \succ U_N(y)$), если $u_i(x) \geq u_i(y), \forall i \in N$ и хоть одно неравенство строгое, то есть $\exists j \in N : u_j(x) > u_j(y)$.

В соответствии со слабой аксиомой Парето вектор выигрыша игроков $U_N(x) = (u_i(x))_{i \in N}$ в ситуации *x* игры *G* сильно доминирует вектор выигрыша $U_N(y)$ в ситуации *y* (обозначают это через $U_N(x) \succ\succ U_N(y)$), если $u_i(x) > u_i(y), \forall i \in N$.

Ситуация x^* игры *G* называется оптимальной по Парето (множество этих ситуаций будем обозначать через $PO(G)$), если не существует ситуации, которая бы ее доминировала, то есть

$$x^* \in PO(G) \Leftrightarrow \neg \exists x \in X : U_N(x) \succ U_N(x^*). \quad (1)$$

Ситуация x^* игры *G* называется слабо оптимальной по Парето (оптимальной за Слейтером, множество этих ситуаций будем обозначать через $SO(G)$), если не существует ситуации, которая бы ее сильно доминировала, то есть

$$x^* \in SO(G) \Leftrightarrow \neg \exists x \in X : U_N(x) \succ\succ U_N(x^*). \quad (2)$$

«Коллективная оптимальность» сильной (слабой) Парето-оптимальной ситуации является залогом того, что для всех игроков найти какую-то лучшую (сильно лучшую) ситуацию невозможно. Условия существования Парето-оптимальных ситуаций являются достаточно «слабыми» и, грубо говоря, сводятся к условиям существования максимумов функций выигрыша игроков (конечность множеств стратегий игроков или компактность множеств стратегий и непрерывность функций выигрыша). Поскольку таких ситуаций, как правило не одна [2], то возникает вопрос – которую выбрать? В одних Парето-оптимальных ситуациях больший выигрыш могут иметь одни игроки, в других – другие игроки. Поэтому соглашения, которые базируются на них, являются «неустойчивыми» относительно отклонений игроков от согласованных стратегий.

Концепция равновесия по Нешу, базируется на принципе «собственной оптимальности». Ситуация x^* называется равновесием по Нешу (множество этих ситуаций будем обозначать через $NE(G)$), если:

$$\forall\ i \in N,\ u_i(x^*) \geq u_i(x_i, x^*_{N\setminus i}), \forall\ x_i \in X_i \ \text{ или } \ \neg\exists x_i \in X_i : u_i(x_i, x^*_{N\setminus i}) > u_i(x^*). \quad (3)$$

Из этого определения следует, что данная ситуация является равновесием Неша, если от нее невыгодно отклоняться любому одному игроку (все другие свои стратегии не изменяют), поскольку значение его функции выигрыша не улучшается (является оптимальным для каждого игрока).

Если игроки заключают соглашение о своем будущем поведении и ее основой является равновесие Неша, то собственный оптимум является залогом «стабильности» выбранной ситуации. «Ценой» привлекательности равновесий Неша являются серьезные проблемы, которые связаны с их существованием, сложностью нахождения, проблемой выбора единственного равновесия [3].

Одним из обще принятых подходов к преодолению проблемы отсутствия равновесий Неша является организация игроков в коалиции. В этом случае естественным обобщением равновесия Неша становится коалиционное равновесие. Поскольку в коалиционной игре выигрыш коалиции игроков определяется выигрышами всех членов коалиции, то определение в данном случае понятия предпочтения одной ситуации над другой базируется также на аксиомах Парето.

Пусть $Q = \{N(k)\}_{k\in K}$ некоторое разбиение множества игроков на коалиции $N(k) \subseteq N, k \in K$; $N(i) \cap N(j) = \varnothing, i \neq j$; $\bigcup_{k\in K} N(k) = N$. Тогда в соответствии со слабой аксиомой Парето ситуация x^* называется сильным коалиционным равновесием по Нешу в игре G, если:

$$\forall k \in K \quad \neg\exists x_{N(k)} \in X_{N(k)}:\ U_{N(k)}(x_{N(k)}, x^*_{N\setminus N(k)}) \succ U_{N(k)}(x^*_{N(k)}, x^*_{N\setminus N(k)}) \quad (4)$$

где $X_{N(k)} = \prod_{i\in N(k)} X_i, k \in K$, - множества коалиционных стратегий игроков; $U_{N(k)}(x) = (u_i(x))_{i\in N(k)}, k \in K$, - векторы функций выигрыша коалиций. Будем обозначать множество этих равновесий в игре G через $SKNE_Q(G)$.

Аналогично в соответствии с сильной аксиомой Парето, определяется множество $WKNE_Q(G)$ слабых коалиционных равновесий Неша в игре G:

$$WKNE_Q(G) \Leftrightarrow \quad \Leftrightarrow \forall k \in K, \neg\exists x_{N(k)} \in X_{N(k)} : U_{N(k)}(x_{N(k)}, x^*_{N\setminus N(k)}) \succ\succ U_{N(k)}(x^*_{N(k)}, x^*_{N\setminus N(k)}). \quad (5)$$

Рассмотрим предельные случаи. Если коалиция одна ($|K| = 1$), то есть она совпадает с множеством всех игроков N, то множество коалиционных равновесий Неша будет множеством оптимальных по Парето ситуаций в игре G. Если коалиций столько, сколько игроков ($|K| = n$) и $N(i) = \{i\}, i \in N$, то множество коалиционных равновесий Неша совпадет со множеством равновесий Неша.

В коалиционном равновесии принцип «собственной оптимальности» заменятся уже другим принципом «коалиционной оптимальности», который заключается в выборе коалицией совместных коалиционных стратегий и в формировании коалиционной цели. Если в игре G не существует равновесий Неша, то при достаточно слабых предположениях (именно таких, которые нужны для существования Парето-оптимальных ситуаций) всегда найдется такое разбиение множества игроков на коалиции (в худшем случае это будет одна коалиция, которая совпадает со всем множеством игроков N), когда множество коалиционных равновесий будет непустым. Хотя замена принципа «собственной оптимальности» на «коалиционную оптимальность» увеличивает шансы на существование равновесий, с другой стороны, стабильность этих равновесий существенно уменьшается с укрупнением коалиций.

На практике, например, в Парламенте процесс распределения игроков на коалиции происходит имперически. Методом проб и ошибок создаются и разрушаются те или другие коалиции. Критерием истинности в таком процессе является стабильность парламентских соглашений. В теории,

формулируются достаточно сложные комбинаторно-игровые задачи, например [4], которые позволяют найти оптимальный в том или другом понимании состав коалиций.

Сравнительный анализ рассмотренных выше принципов оптимальности показывает, что они отличаются как уровнем агрегации стратегий (от собственных стратегий через коалиционные стратегии к коллективным стратегиям всего множества игроков), так одновременно и уровнем толерантности игроков. В равновесиях Неша толерантность игроков минимальна, поскольку каждый учитывает лишь собственные интересы. В коалиционных равновесиях она увеличивается, поскольку каждый учитывает интересы членов своей коалиции. В Парето-оптимальных ситуациях толерантность игроков максимальна, поскольку каждый учитывает интересы всех.

В этой работе предлагается использовать принцип «индивидуальной оптимальности», который будет обобщением принципа «собственной оптимальности» в направлении увеличения толерантности игроков до уровня соответствующего «коллективной оптимальности», что приводит к учету каждым игроком функций выигрыша всех других игроков при выборе собственных стратегий. При таком подходе каждый игрок будет фактически максимизировать не только свой выигрыш, а вектор выигрышей всех игроков, но с максимальным предпочтением к своему собственному выигрышу.

Индивидуально-оптимальные равновесия

Допустим, что каждый игрок выбирает свои стратегии независимо от других, но учитывает при этом функции выигрыша всех других игроков. Сформулируем понятие индивидуально-оптимального равновесия.

Сначала введем специальное отношение доминирования, которое будем называть отношением доминирования Неша.

Будем говорить, что ситуация x игры G с вектором выигрыша $U_N(x) = (u_i(x))_{i \in N}$ доминирует (сильно доминирует) по Нешу ситуацию y с вектором выигрыша $U_N(y)$ в и будем обозначать это через $x \overset{NE}{\succ} y$ ($x \overset{NE}{\succ\succ} y$), если y получается из ситуации x изменением каким-то, но одним игроком $i \in N$, своей собственной стратегии, то есть $y = (y_i, x_{N\setminus i})$ и $U_N(x) \succ U_N(y)$, таким образом:

$$x \overset{NE}{\succ} y \Leftrightarrow \exists i \in N: \; U_N(x_i, x_{N\setminus i}) \succ U_N(y_i, x_{N\setminus i}), \; \exists i \in N: \; U_N(x_i, x_{N\setminus i}) \succ U_N(y_i, x_{N\setminus i})$$

$$((x \overset{NE}{\succ\succ} y \Leftrightarrow \exists i \in N: \; U_N(x_i, x_{N\setminus i}) \succ\succ U_N(y_i, x_{N\setminus i})).$$

Ситуацию x^* будем называть сильным индивидуально-оптимальным равновесием, а их множество будем обозначать через $SIOE(G)$, если не существует другой ситуации, которая бы доминировала ее по Нешу, то есть

$$x^* \in SIOE(G) \Leftrightarrow \neg \exists x \in X : x \overset{NE}{\succ} x^*.$$

В соответствии с отношением сильного доминирования Неша определим понятие слабого индивидуально-оптимального равновесия. Ситуацию x^* будем называть слабым индивидуально-оптимальным равновесием (их множество будем обозначать через $WIOE(G)$), если не существует другой ситуации, которая бы сильно доминировала ее по Нешу, то есть

$$x^* \in WIOE(G) \Leftrightarrow \neg \exists x \in X : x \overset{NE}{\succ\succ} x^*.$$

Использование индивидуально-оптимальных равновесий мотивируется следующим сценарием игры. К началу игры игроки заключают необязательное соглашение (за нарушение соглашения не предусмотрена ни одна штрафная санкция) о том, что они выберут ситуацию $x^* = (x_i^*)_{i \in N}$, затем независимо один от другого игроки принимают решение о выборе своей стратегии. В том и только том случае, если основой соглашения будет сильное (слабое) индивидуально-оптимальное равновесие, *i*–му игроку, отдельно, будет невыгодно выбирать стратегию отличающуюся от x_i^*, поскольку выигрыш свой и других игроков, чьи интересы он учитывает, не улучшится (сильно не улучшится). Лояльность (а может надежда, что

другие игроки поделятся выигрышем) каждого игрока по отношению к другим, будет залогом стабильности такого соглашения.

В этой работе дальше мы будем детально исследовать слабые индивидуально-оптимальные равновесия.

Интересно рассмотреть слабые индивидуально-оптимальные равновесия в предельных случаях, а также сравнить их с оптимальными по Парето, Нешевскими и коалиционными равновесиями.

Теорема 1. Справедливы включения: $PO(G) \subseteq SO(G) \subseteq WIOE(G)$; $NE(G) \subseteq WIOE(G)$; $SKNE_Q(G) \subseteq WKNE_Q(G) \subseteq WIOE(G)$.

Доказательство. Включение $PO(G) \subseteq SO(G)$ элементарно выводится из определений (1),(2) и является известным [2].

Докажем включение $SO(G) \subseteq WIOE(G)$. Пусть $x^* \in SO(G) \neq \varnothing$. Предположим противное, что $x^* \notin WIOE(G)$. Тогда $\exists i \in N$, $\exists x_i \in X_i$: $U_N(x_i, x^*_{N\setminus i}) \succ\succ U_N(x^*_i, x^*_{N\setminus i})$. Отсюда следует, что по крайней мере для $y = (x_i, x^*_{N\setminus i})$, $U_N(y) \succ\succ U_N(x^*)$. Получаем противоречие с (2).

Докажем включение $NE(G) \subseteq WIOE(G)$. Пусть $x^* \in NE(G) \neq \varnothing$. Предположим противное, что $x^* \notin WIOE(G)$. Тогда $\exists i \in N$, $\exists x_i \in X_i$: $U_N(x_i, x^*_{N\setminus i}) \succ\succ U_N(x^*_i, x^*_{N\setminus i})$. Отсюда, следует что $u_i(x_i, x^*_{N\setminus i}) > u_i(x^*)$. Отсюда по определению (3) $x^* \notin NE(G)$. Получили противоречие.

Пусть $Q = \{N(k)\}_{k\in K}$ некоторая разбиение множества игроков на коалиции $N(k) \subseteq N, k \in K$; $N(i) \cap N(j) = \varnothing, i \neq j$; $\bigcup_{k\in K} N(k) = N$. Включение $SKNE_Q(G) \subseteq WKNE_Q(G)$ элементарно выводится из определений (4), (5) и является известным [2].

Докажем включение $KNE_Q(G) \subseteq IOE_Q(G)$. Пусть $x^* \in KNE_Q(G)$. Предположим от противного, что $x^* \notin WIOE(G)$. Тогда $\exists i \in N$, $\exists x_i \in X_i$: $U_N(x_i, x^*_{N\setminus i}) \succ\succ U_N(x^*_i, x^*_{N\setminus i})$. Возьмем коалицию $N(k)$, к которой принадлежит *i*-й игрок. Обозначим через $y_{N(k)} = (x_i, x^*_{N(k)\setminus i})$. Отсюда следует, что $u_j(y_{N(k)}, x^*_{N\setminus N(k)}) > u_j(x^*), \forall j \in N(k)$. По определению (5) получим $x^* \notin KNE_Q(G)$. Получили противоречие. ♦

При достаточно слабых предположениях относительно условий игры (например, конечность множеств стратегий игроков или компактность множеств стратегий игроков и непрерывность их функций выигрыша) $SO(G) \neq \varnothing$, поэтому $WIOE(G) \neq \varnothing$. Из теоремы 1 также видно, что

$$WIOE(G) \supseteq NE(G) \cup SKNE_Q(G) \cup WKNE_Q(G) \cup PO(G) \cup SO(G).$$

По построению рассмотренных множеств равновесий очевидно, что наиболее «стабильными» будут равновесия Неша, если они существуют, а наиболее чуствительными к нарушению соглашения будут слабые Парето-оптимальные ситуации, остальные занимают промежуточные места. Поскольку все рассмотренные выше равновесия являются индивидуально оптимальными, поэтому для поиска наиболее стабильного из существующих попробуем параметрически описать все множество индивидуально-оптимальных равновесий.

Теорема 2. Пусть без ограничения общности функции выигрыша всех игроков в ситуации x^* принимают положительные значения, то есть $u_i(x^*) > 0, i \in N$. Ситуация x^* будет слабым индивидуально-оптимальным равновесием тогда и только тогда, когда существуют векторы параметров

$$\mu_i \in M_i^+ = \left\{ \mu_i = (\mu_i^j)_{j\in N} \middle| \sum_{j\in N} \mu_i^j = 1;\ \mu_i^j > 0,\ j \in N \right\},\ i \in N \tag{6}$$

такие, что ситуация x^* будет равновесием Неша в следующей параметрической игре:

$$G(\mu) = (X_i, \min_{j\in N} \mu_i^j u_j(x); i \in N). \tag{7}$$

Доказательство. Докажем достаточность. Пусть *x** - равновесие Неша в параметрической игре (6) при некоторых значениях параметров $\hat{\mu}_i \in M_i^+$, $i \in N$. Отсюда следует, что для любого игрока $i \in N$ и любой стратегии $x_i \in X_i$ имеют место неравенства: $\min\limits_{j \in N} \hat{\mu}_i^j u_j(x_i, x_{N\setminus i}{}^*) \le \hat{\mu}_i^k u_k(x^*),\ \forall k \in N$, поэтому существует такой игрок $j \in N$, что $u_j(x^*) \ge u_j(x_i, x_{N\setminus i}^*), \forall x_i \in X, \forall i \in N$. Следовательно $\neg\exists x = (x_i, x_{N\setminus i}^*) \in X : x \overset{NE}{\succ\succ} x^*$. Отсюда x^* будет слабым индивидуально-оптимальным равновесием.

Докажем необходимость. Для этого возьмем вектор $\mu_i,\ i \in N$ с компонентами, которые определены по следующим формулам: $\bar{\mu}_i^j = \lambda_i / u_j(x^*),\ j \in N;\ \lambda_i = 1 \Big/ \sum\limits_{k \in N} \dfrac{1}{u_k(x^*)},\ i \in N$.

Отметим, что $\bar{\mu}_i \in M_i^+\ \ i \in N$. Из того, что x^* - слабое индивидуально-оптимальное равновесие следует, что $\neg\exists x \in X : x \overset{NE}{\succ\succ} x^*$, то есть $\exists i \in N, \exists j \in N : u_j(x^*) \ge u_j(x_i, x^*{}_{N\setminus i})$, а значит $\bar{\mu}_i^j u_i(x^*) \ge \bar{\mu}_i^j u_j(x_i, x^*{}_{N\setminus i})$. Поскольку $\bar{\mu}_i^j u_j(x^*) = \lambda_i = 1 \Big/ \sum\limits_{k \in N} \dfrac{1}{u_k(x^*)} = const$, то любого игрока $i \in N$ и для любой стратегии $x_i \in X_i$ имеют место неравенства: $\min\limits_{j \in N} \bar{\mu}_i^j u_j(x_i, x_{N\setminus i}^*) \le \bar{\mu}_i^k u_k(x^*),\ \forall k \in N$. Поэтому *x** - равновесие при Нешем параметрической игре (6). ♦

Проблема выбора индивидуально-оптимальных равновесий

Теорема 2 дает возможность конструктивно описать множества $NE(G), WKNE_Q(G), SO(G), WIOE(G)$ как решения параметрической игры следующим образом.

1) $x^* \in NE(G) \Leftrightarrow x^* \in NE(G(\mu))$ при $\mu_i^j \to 0;\ \mu_i^j \to 1,\ \forall j \ne i;\ i \in N$. Поскольку величина $1/\mu_i^j,\ 1/\mu_i^j \in (1, \infty)$, характеризует приоритет функции выигрыша *j*-го игрока в предпочтении *i*-го на множестве всех игроков, указанные условия означают, что каждого игрока интересует лишь собственный выигрыш, выигрыши других он не учитывает.

2) $x^* \in WKNE_Q(G) \Leftrightarrow x^* \in NE(G(\mu))$ при $\mu_i^k \to 0,\ \forall k \in N(k_i);\ \ \mu_i^j \to 1,\ \forall j \in N \setminus N(k_i);\ i \in N$, где $N(k_i)$ - коалиция, к которой принадлежит *i* -й игрок, $i \in N$. В этом случае каждого игрока интересует лишь выигрыш членов его коалиции, выигрыши других он не учитывает.

3) $x^* \in SO(G) \Leftrightarrow x^* \in NE(G(\mu))$ при $\mu_j^i = \mu_k^i,\ \forall j,k \in N;\ i \in N$. В этом случае каждый игрок учитывает интересы всех других, но приоритеты всех игроков по отношению к *i*-му одинаковы, $i \in N$.

4) $x^* \in WIOE(G) \Leftrightarrow x^* \in NE(G(\mu))$ при произвольных векторах параметров $\mu_i \in M_i^+,\ i \in N$. В этом случае каждый игрок учитывает интересы всех других с произвольными приоритетами.

Особенную роль играют параметры $\mu_i^i,\ i \in N$, которые характеризуют приоритет $1/\mu_i^i \in (1, \infty)$ собственной функции выигрыша игрока по отношению к функциям выигрыша других игроков.

Допустим для простоты исследования, что приоритеты функций выигрыша всех игроков, с которыми считается *i* для него одинаковые и множества $NE(G), WKNE_Q(G), SO(G), WIOE(G)$ непустые. Тогда при $\mu_i^i \to 0,\ \forall i \in N$, получим $NE(G) = NE(G(\mu))$. При большем значении $\mu_i^i = \varepsilon |N(k_i)|,\ \forall i \in N$, где $\varepsilon \to 0$, а k_i - номер коалиции, в состав которой входит *i*-й игрок, получим $WKNE_Q(G) = NE(G(\mu))$, при чем чем больше коалиция, тем больший весовой коэффициент μ_i^i и соответственно меньший приоритет собственного критерия игрока *i*. При еще большем значении $\mu_i^i = \varepsilon |N|,\ \forall i \in N, \varepsilon \to 0$, будут еще

большие весовые коэффициенты $\mu_i^i, i \in N$ и соответственно меньшие приоритеты собственных критериев игроков, тогда мы получим $SO(G) = NE(G(\mu))$.

На основе этой оценки можно выдвинуть гипотезу, что величина приоритета $1/\mu_i^i$ собственной функции выигрыша характеризует степень «индивидуальности» для *i*-го игрока индивидуально-оптимального равновесия, а обратная к приоритету $1/\mu_i^i$ величина μ_i^i характеризует толерантность *i*-го игрока по отношению к другим. Поэтому меньшей толерантности всех игроков отвечает больший уровень «стабильности» ситуации (равновесие Неша), большей толерантности – меньший уровень (Парето-оптимальность).

Эти рассуждения приводят к мысли, что каждую ситуацию $x \in X$ можно оценить минимально возможным уровнем толерантности всех игроков необходимым и достаточным для того, чтобы она была индивидуально-оптимальным равновесием. Эта оценка будет оценкой уровня «стабильности» этого равновесия.

Для произвольной ситуации $x \in X$ сформулируем следующую оптимизационную задачу относительно весовых коэффициентов $\mu_i^j;\ j \in N$, в которой ситуация $x \in X$ выступает как параметр:

$$\mu^{\min}(x) = \inf_{\mu} \max_{i \in N} \mu_i^i \tag{8}$$

$$\min_{j \in N} \mu_i^j u_j(x) \geq \min_{j \in N} \mu_i^j u_j(y_i, x_{N \setminus i}),\ y_i \in X_i,\ i \in N \tag{9}$$

$$\sum_{j \in N} \mu_i^j = 1;\ \mu_i^j \geq 0;\ i, j \in N \tag{10}$$

Ограничение (9) описывают по определению (3) условия существования равновесия Неша для параметрической игры $G(\mu)$. Ограничения (10) обеспечивают условие $\mu_i \in M_i^+, i \in N$. На основании теоремы 2 $x \in WIOE(G)$, тогда и только тогда, когда система ограничений (9),(10) будет совместной. Если ситуация $x \notin WIOE(G)$, будем считать $\mu^{\min}(x) = 1$. Величина $\mu^{\min}(x)$ будет точной нижней границей гарантированного уровня толерантности всех игроков достаточным для того, чтобы ситуация $x \in X$ была индивидуально-оптимальным равновесием, и будет оценкой уровня ее «стабильности».

Отображение $\mu^{\min} : X \to [0,1]$ будем называть критерием толерантности. Индивидуально-оптимальное равновесие x^* игры *G*, которое обеспечивает $\min_{x \in WIOE(G)} \mu^{\min}(x)$, то есть минимизирует критерий толерантности, может быть рекомендовано игрокам как основа для стабильного соглашения.

Этот подход является особенно актуальным, когда в игре *G* не существует равновесий Неша. Тогда x^* будет «наиболее стабильным» индивидуально-оптимальным равновесием (возможно коалиционным равновесием, возможно Парето-оптимальным).

Например, рассмотрим игру двух лиц. Первый игрок хочет продать товар второму игроку, при этом может назначить как высокую цену (ВЦ), так и низкую (НЦ). Второй игрок может купить товар (К) или нет (НК). Будем считать, что продажа товара как по высокой, так и низкой цене безубыточны для продавца, а у покупателя достаточно денег для покупки товара по любой цене. Если продавец предложил высокую цену, а покупатель отказался покупать товар, продавец терпит моральный ущерб от своей жадности. Если покупатель покупает товар по высокой цене, он также получает моральный ущерб от своей расточительности. Игра может описываться следующей таблицей выиграшей.

	К	НК
ВЦ	2 -1	-1 0
НЦ	1 2	0 0

В этой игре нет равновесий Неша, но есть две индивидуально-оптимальные ситуации (ВЦ,К) и (НЦ,К), которые оптимальны по Парето. Наиболее стабильным индивидуально-оптимальным равновесием является ситуация (НЦ,К), которая может быть рекомендована игрокам как основа соглашения. Таким образом продавцу целесообразно не гнаться за сверхприбылями и продавать товар но низкой цене, покупателю рекомендуется не отказываться от покупки.

Заключение

Следует заметить, что понятие стабильности равновесий является достаточно сложным и многоаспектным (более или менее в полной мере оно исследовано в работах лауреатов Нобелевской премии по экономике за 1994 год Дж. Харшаньи и Р. Зельтеном [2]) и связано, в основном, с выбором единственного равновесия из множества равновесий Неша на основе критериев эффективности по выигрышу и эффективности по риску. Описанный выше подход предлагает новую (более широкую) модель выбора равновесий с возможностью использования наряду с критериями эффективности по выигрышу и риску критерия толерантности.

Ссылки

[1] Мулен Э. Теория игр с примерами из математической экономики. –М.: Мир, 1985. [2] Подиновский В.В., Ногин В.Д. Парето-оптимальные решения многокритериальных задач. –М.: Наука, 1982. [3] Харшаньи Дж., Зельтен Р. Общая теория выбора равновесия в играх. –Санкт-Петербург: Экономическая школа, 2001.

[4] Бухтояров С.Е., Емеличев В.А. Конечные коалиционные игры: параметризация принципа оптимальности(«вот Парето до Неша») и устойчивость обобщенно еффективных ситуаций//Докл. НАН Беларуси.-2002.-46, N6.-С.36-38.

Информация об авторе

Мащенко Сергей Олегович *– Киевский национальный университет имени Тараса Шевченко, Доцент; Проспект академика Глушкова, 6, Киев – 207, Украина; e-mail: msomail@yandex.ru*

ФОРМАЛИЗАЦИЯ СТРУКТУРНЫХ ОГРАНИЧЕНИЙ СВЯЗЕЙ В МОДЕЛИ „СУЩНОСТЬ-СВЯЗЬ”

Дмитрий Буй, Людмила Сильвейструк

Резюме*: Рассматриваются и формализируются в терминах теории отношений такие основные понятия модели „сущность-связь”: сущности, связи, структурные ограничения связей (показатель кардинальности, степень участия, структурные ограничения вида (min, max)). Для бинарных отношений введены два оператора min, max, в терминах которых задаются указанные структурные ограничения; приведена основная теорема о совместности значений этих операторов на исходном отношении и отношении, обратном к нему.*

Ключевые слова: *сущность, связь, показатель кардинальности, степень участия, структурное ограничение вида (min, max).*

АСМ классификация ключевых слов*: E.4 Coding and information theory – Formal models of communication*

Введение

Моделирование данных – это процесс создания логического представления структуры базы данных. Существуют разные подходы к моделированию данных, каждый из которых имеет своих поклонников. Одна из таких моделей – ER-модель (Entity-Relationship model, модель „сущность-связь"). Эта модель стала традиционной и наиболее популярной.

Данную модель предложил в 1976 г. Питер Чен [Chen P., 1976]. Нужно отметить, что модель постоянно расширяется и модифицируется. Кроме того, модель „сущность-связь" вошла в состав многих CASE-инструментов, которые также внесли свой вклад в ее эволюцию. Последние исследования в области ER-моделирования приведены, например, в работах К. Батини, С. Кэрри и Б. Зелхема [Batini Carlo, Ceri S., Navathe S.B. and Batini Carol, 1991; Thelheim B., 2000].

На сегодняшний день не существует единственного общепринятого стандарта для ER-модели, но имеется набор общих конструкций, которые лежат в основе большинства её вариантов [Гарсиа-Молина Г., Ульман Дж., Уидом Дж., 2004, гл. 2; Дейт Дж., 1998, часть III, гл. 14; Коннолли Т., Бегг К., Страчан А., 2000, часть II, гл. 5; Крёнке Д., 2003, часть II, гл. 3].

Необходимо отметить, что модель „сущность-связь" не является формальной моделью или, если сказать точнее, является такой не в первую очередь. Фактически она состоит из набора преимущественно неформальных концепций, но в ней присутствуют и формальные аспекты. В данной работе формализуются такие основные понятия модели как сущности, связи и структурные ограничения связей.

Сущности и связи

Ключевыми понятиями модели „сущность-связь" являются *сущности*, *атрибуты* и *связи*.

Следует отметить, что понятия модели не имеют общепринятой и четкой интерпретации; более того, существует существенный терминологический разнобой; поэтому приведем в следующей таблице все варианты, встречающиеся в литературе.

Табл. 1 – Ключевые понятия ER-модели и их названия

Понятие (содержательное объяснение)	1 вариант	2 вариант	3 вариант	4 вариант
Сущность (состоит из своих экземпляров; содержит, порождает свои экземпляры)	Тип сущности	Множество сущности	Тип сущности	Класс сущности
Экземпляр сущности (принадлежат своей сущности, порождаются своей сущностью)	Экземпляр сущности	Экземпляр сущности	Сущность	Экземпляр сущности
Атрибут	Свойство	Атрибут	Атрибут	Атрибут
Связь (состоит из своих экземпляров; содержит, порождает свои экземпляры)	Тип связи	Связь	Тип связи	Класс связи
Экземпляр связи (принадлежат своей связи, порождаются своей связью)	Экземпляр связи	Экземпляр связи	Связь	Экземпляр связи
Источник	[Дейт Дж., 1998, ч. III, гл. 14]	[Гарсиа-Молина Г., Ульман Дж., Уидом Дж., 2004, гл. 2]	[Коннолли Т., Бегг К., Страчан А., 2000, ч. II, гл. 5]	[Крёнке Д., 2003, ч. II, гл. 3]

Далее будем придерживаться 3-го варианта.

Сущности. Тип сущности будем интерпретировать как множество, а сущность – как элемент этого множества.

Связи. Содержательно, связь это ассоциация между n сущностями, которые называются ее участниками. Число участников связи называется *степенью связи* (*relationship degree*) или *арностью* связи. Отдельные связи формируют тип связи, причем арность всех связей одного типа связи одинаковая; таким образом, арность является характеристикой типа связи. Здесь полная аналогия с арностью логико-математических отношений и количеством компонент кортежей, из которых (кортежей) состоит отношение.

Рассмотрим бинарные связи. *Бинарная связь* в общем случае способна соединить любую сущность одного типа сущности с любой сущностью другого типа сущности, в частности, с любой сущностью того же типа сущности (в частности, бинарная связь может соединить сущность саму с собой; здесь полная аналогия с рефлективностью бинарных отношений; отметим также, что для таких связей иногда используется явно неудачный термин „рекурсивная связь").

Типы связей будем уточнять в виде (конечноарных) логико-математических отношений; в частности, типы бинарных связей – в виде бинарных отношений.

Показатель кардинальности. В модели существуют стандартные ограничения, которые накладываются на типы связи. Одно из таких ограничений – показатель кардинальности. *Показатель кардинальности* (*index cardinality*), содержательно говоря, задает количество возможных связей для каждой сущности-участницы связи; если говорить более точно, то этот показатель задает количество сущностей, ассоциируемых с фиксированной сущностью. Заметим, что употребление термина „кардинальность", учитывая финитность модели, в данном контексте явно неудачное.

Среди бинарных типов связи выделяют типы связи с показателями кардинальности „один к одному" (1:1), „многие к одному" (M:1), „один ко многим" (1:M) и „многие ко многим" (M:N).

Допустим, что R – тип связи, которая соединяет типы сущностей E и F. Для адекватной формализации показателя кардинальности типы сущностей E и F будем интерпретировать как множества **E** и **F** соответственно, а тип связи R – как бинарное отношение **R**, причем $\mathbf{R} \subseteq \mathbf{E} \times \mathbf{F}$ (порядок множеств в декартовом произведении существенен). Как обычно $\mathbf{R}^{-1} \subseteq \mathbf{F} \times \mathbf{E}$ – отношение, обратное к **R**.

В табл. 2 показана взаимосвязь между показателями кардинальности типа связи и свойствами функциональности бинарных отношений **R**, $\mathbf{R}^{-1}$.

Табл. 2 – Взаимосвязь между показателем кардинальности и функциональностью бинарных отношений, а также значениями оператора max

		Отношение **R**	
		R функциональное	**R** не функциональное
Отношение $\mathbf{R}^{-1}$	$\mathbf{R}^{-1}$ функциональное	R – „один к одному" (1:1)	R – „многие к одному" (M:1), которая направлена от F к E R – „один ко многим" (1:M), которая направлена от E к F
		$\max(\mathbf{R}) \le 1 \wedge \max(\mathbf{R}^{-1}) \le 1$	$2 \le \max(\mathbf{R}) \le \infty \wedge \max(\mathbf{R}^{-1}) \le 1$
	$\mathbf{R}^{-1}$ не функциональное	R – „многие к одному" (M:1), которая направлена от E к F R – „один ко многим" (1:M), которая направлена от F к E	R – „многие ко многим" (M:N)
		$\max(\mathbf{R}) \le 1 \wedge 2 \le \max(\mathbf{R}^{-1}) \le \infty$	$2 \le \max(\mathbf{R}) \le \infty \wedge 2 \le \max(\mathbf{R}^{-1}) \le \infty$

Как видно, тип связи вида „один к одному” будет, когда оба отношения **R, R**$^{-1}$ функциональные; тип связи вида „один ко многим” (эквивалентно „многие к одному”) – когда в точности одно из этих отношений функциональное; и, наконец, тип связи вида „многие ко многим” – когда оба эти отношения не функциональны.

Таким образом, можно сделать вывод: ограничение „один ко многим” („многие к одному”) связано с функциональностью (эквивалентно: с инъективностью); „один к одному” – с одновременной функциональностью и инъективностью; наконец „многие ко многим” – с бинарными отношениями общего вида. Здесь нужно учесть очевидную логическую связь между функциональностью и инъективностью: бинарное отношение функциональное тогда и только тогда, когда обратное отношение инъективное (см., например [Буй Д. Б., Кахута Н. Д., 2005, утверждение 1]).

Степень участия сущности в связи. Существует еще одно ограничение для типов связи – *степень участия сущности в связи*. Одна из возможных интерпретаций: степень участия определяет зависимость существования некоторого типа сущностей от участия в типе связи другого типа сущностей (по крайней мере, для полного участия в связи; см. далее).

Существует два вида участия типа сущности в типе связи: полное и частичное. Пусть R – тип связи, а тип сущности E – участник типа связи R (отметим, что понятие участника естественно переносится с связей на типы связей). Характеристический признак такой: если каждая сущность типа E находится по крайней мере в одной связи согласно типа связи R, то участие типа сущности E в типе связи R называется *полным* (*total*), в противном случае (то есть существует сущность типа E, которая согласно типа связи R не находится в связи ни с одной сущностью другого типа сущности) – *частичным (partial)*.

Формализировать понятие полного и частичного участия можно посредством проекции отношения (табл. 3). В предыдущих обозначениях имеем нижеследующую таблицу, где $\pi_1^2(\mathbf{R})$, $\pi_2^2(\mathbf{R})$ – проекции бинарного отношения соответственно по первой и второй компонентам.

Табл. 3 – Взаимосвязь между степенью участия и проекцией отношения, а также значениями оператора min

Проекция отношения	Степень участия типа сущности в связи	Значение оператора min
$\pi_1^2(\mathbf{R}) = \mathbf{E}$	участие типа сущности E в типе связи R полное	$0 < \min(\mathbf{R})$
$\pi_1^2(\mathbf{R}) \subset \mathbf{E}$	участие типа сущности E в типе связи R частичное	$\min(\mathbf{R}) = 0$
$\pi_2^2(\mathbf{R}) = \mathbf{F}$	участие типа сущности F в типе связи R полное	$0 < \min(\mathbf{R}^{-1})$
$\pi_2^2(\mathbf{R}) \subset \mathbf{F}$	участие типа сущности F в типе связи R частичное	$\min(\mathbf{R}^{-1}) = 0$

Структурные ограничения вида (min, max). Существует и альтернативный вариант рассмотрения ограничений на типах связи – так называемые структурные ограничения, которые предусматривают задание минимальных и максимальных значений (min, max). Как далее будет показано, эти структурные ограничения позволяют специфицировать больше информации о связи.

Формализировать такие структурные ограничения можно посредством понятия полного образа. Как и ранее типы сущностей E, F будем интерпретировать как непустые множества **E**, **F**; элементы таких множеств будем обозначать как $x, y, \ldots$.

Пусть $x \in \mathbf{E}$, полный образ одноэлементного множества $\{x\}$ относительно отношения **R** обозначим через $\mathbf{R}[x]$; напомним, что согласно стандартному определению $\mathbf{R}[x] \stackrel{def}{=} \{ y \mid y \in \mathbf{F} \wedge < x, y > \in \mathbf{R} \}$ – множество всех элементов множества **F**, которые находятся в отношении **R** с элементом x.

Будем считать, что множества **E**, **F** не более чем счетные, а все полные образы одноэлементных множеств конечные; учитывая финитность всех объектов, такие ограничения являются естественными.

Множество мощностей полных образов всех элементов множества **E** обозначим как $\mathrm{Im}(\mathbf{R}) \stackrel{def}{=} \{ |\mathbf{R}[x]| \mid x \in \mathbf{E} \}$.

Очевидно, что $\mathrm{Im}(\mathbf{R})$ – непустое подмножество натуральных чисел, конечное (то есть ограниченное

сверху) или бесконечное (то есть неограниченное сверху). В любом случае это множество имеет наименьший элемент, который обозначим, как $\min(\mathbf{R})$. Наибольшего же элемента множество $\mathrm{Im}(\mathbf{R})$ может и не иметь, поэтому введем следующее обозначение, в котором ∞ – некоторый элемент, не принадлежащий множеству натуральных чисел:

$$\max(\mathbf{R}) = \begin{cases} \text{наибольший елемент множества } \mathrm{Im}(\mathbf{R}), \text{если } \mathrm{Im}(\mathbf{R}) \text{ конечное множество,} \\ \infty, \quad \text{в противном случае.} \end{cases}$$

По существу множество натуральный чисел N со стандартным порядком $\leq$ дополнили наибольшим элементом ∞, превратив его в полную решетку $\langle N', \leq \rangle$, где $N' \overset{def}{=} N \cup \{\infty\}$, причем $n < \infty$ для всех $n \in N$ (по поводу условной полноты и пополнения условно полных решеток см., например, [Биркгоф Г.,1984, гл. V, § 3, с. 153-154]). Непосредственно из определений вытекают равенства

$$\min(\mathbf{R}) = \prod \mathrm{Im}(\mathbf{R}), \ \max(\mathbf{R}) = \coprod \mathrm{Im}(\mathbf{R}), \qquad (*)$$

где символы $\prod, \coprod$ используются для обозначения инфимумов и супремумов соответственно (в полной решетке N').

Основная задача заключается в исследовании логической связи между значениями введенных операторов на исходном отношении ($\mathbf{R}$) и на отношении, обратном к нему ($\mathbf{R}^{-1}$). Эта задача будет решена в приведенной далее теореме, доказательство которой опирается на следующие леммы о основных свойствах операторов $\min, \max$.

Лемма 1. Для произвольного бинарного отношения $\mathbf{R}$ выполняются следующие утверждения:

1. $\min(\mathbf{R}) \leq \max(\mathbf{R})$, более того $\max(\mathbf{R}) = \infty \Rightarrow \min(\mathbf{R}) < \max(\mathbf{R})$;
2. $\min(\mathbf{R}) = \max(\mathbf{R}) \Leftrightarrow \forall x\, y \left(x, y \in \mathbf{E} \Rightarrow \left|\mathbf{R}[x]\right| = \left|\mathbf{R}[y]\right|\right)$;
3. пусть k – такое натуральное число, что $\min(\mathbf{R}) = \max(\mathbf{R}) = k$; тогда $\forall x \left(x \in \mathbf{E} \Rightarrow \left|\mathbf{R}[x]\right| = k\right)$;
4. пусть k – такое натуральное число, что $\forall x \left(x \in \mathbf{E} \Rightarrow \left|\mathbf{R}[x]\right| = k\right)$; тогда $\min(\mathbf{R}) = \max(\mathbf{R}) = k$;
5. $\mathbf{R} = \varnothing \Leftrightarrow \min(\mathbf{R}) = \max(\mathbf{R}) = 0$; более того $\mathbf{R} = \varnothing \Leftrightarrow \min(\mathbf{R}) = \max(\mathbf{R}) = \min(\mathbf{R}^{-1}) = \max(\mathbf{R}^{-1}) = 0$ (характеристическое свойство пустого бинарного отношения);
6. $\pi_1^2(\mathbf{R}) \subset \mathbf{E} \Leftrightarrow \min(\mathbf{R}) = 0$, $\pi_1^2(\mathbf{R}) = \mathbf{E} \Leftrightarrow \min(\mathbf{R}) > 0$;
7. $0 < \min(\mathbf{R}) < \max(\mathbf{R}) \Rightarrow \left|\pi_1^2(\mathbf{R})\right| \geq 2$;
8. $\max(\mathbf{R}) = \infty \Rightarrow \left|\pi_1^2(\mathbf{R})\right| = \left|\pi_2^2(\mathbf{R})\right| = \omega$, где ω – кардинал счетных множеств;
9. $\mathbf{R}$ – функциональное отношение $\Leftrightarrow$ $\max(\mathbf{R}) \leq 1$. □

Из пунктов 6, 9 этой леммы вытекает заполнение таблиц 2, 3 применительно к операторам $\min, \max$. Таким образом, в терминах введенных операторов очень просто выражаются показатель кардинальности и степень участия.

Лемма 2 (значения операторов $\min, \max$ на конечном универсальном отношении). Пусть $\left|\mathbf{E}\right| = l > 0$, $\left|\mathbf{F}\right| = k > 0$, а $U(l,k) \overset{def}{=} \mathbf{E} \times \mathbf{F}$ – универсальное отношение на множествах $\mathbf{E}$, $\mathbf{F}$; тогда $\min(\mathbf{R}) = \max(\mathbf{R}) = l$ и $\min(\mathbf{R}^{-1}) = \max(\mathbf{R}^{-1}) = k$. □

Значения операторов $\min, \max$ зависят не только от аргумента-отношения (в предыдущих обозначениях $\mathbf{R}$), но и от множества-параметра, которому принадлежат первые компоненты пар отношения (множества $\mathbf{E}$); поэтому точнее было бы писать, например, $\min_{\mathbf{E}}(\mathbf{R})$ вместо $\min(\mathbf{R})$. Следующая лемма уточняет зависимость от множества-параметра.

Лемма 3. Пусть отношение $\mathbf{R}$ и множества $\mathbf{E}$, $\mathbf{F}$, $\mathbf{E}'$, такие, что $\mathbf{R} \subseteq \mathbf{E} \times \mathbf{F}$ и $\mathbf{E} \subset \mathbf{E}'$; тогда $\min_{\mathbf{E}'}(\mathbf{R}) = 0$ и $\max_{\mathbf{E}}(\mathbf{R}) = \max_{\mathbf{E}'}(\mathbf{R})$. □

Следовательно, собственное расширение множества-параметра влияет только на значение оператора $\min$, которое из возможно ненулевого становится нулевым.

Следующая лемма рассматривает случай, когда отношение имеет структуру объединения попарно совместных отношений (совместность в понимании [Редько В. Н., Брона Ю. Й., Буй Д. Б., Поляков С. А., 2001], т.е. $U \approx V \overset{def}{\Leftrightarrow} U \mid \mathbf{X} = V \mid \mathbf{X}$, где $\mathbf{X} \overset{def}{=} \pi_1^2 U \bigcap \pi_1^2 V$ – пересечение проекций отношений по первой компоненте, а $U \mid \mathbf{X}$, $V \mid \mathbf{X}$ – ограничения отношений по множеству $\mathbf{X}$), в лемме выражаются значения операторов $\min, \max$ на исходном множестве через значение тех же операторов на множествах из объединения.

Лемма 4. Пусть отношение $\mathbf{R}$ такое, что $\mathbf{R} = \bigcup_{i \in I} \mathbf{R}_i$, причем все отношения $\mathbf{R}_i$, $i \in I$ попарно совместные. Тогда $\max_{\mathbf{E}}(\mathbf{R}) = \coprod_{i \in I} \max_{\mathbf{E}_i}(\mathbf{R}_i)$, где множества $\mathbf{E}, \mathbf{E}_i$, такие, что $\pi_1^2(\mathbf{R}) \subseteq \mathbf{E}$, $\pi_1^2(\mathbf{R}_i) \subseteq \mathbf{E}_i$ для всех $i \in I$. Кроме того, обозначая проекции по первой компоненте отношений $\mathbf{R}$ и $\mathbf{R}_i$ через $\mathbf{G}$ и $\mathbf{G}_i$ соответственно, имеем равенство $\min_{\mathbf{G}}(\mathbf{R}) = \prod_{i \in I} \min_{\mathbf{G}_i}(\mathbf{R}_i)$. □

Доказательство следует с равенства $\mathrm{Im}_{\mathbf{G}}(\mathbf{R}) = \bigcup_{i \in I} \mathrm{Im}_{\mathbf{G}_i}(\mathbf{R}_i)$, равенств (*) и хорошо известного утверждения о точных гранях объединения множеств (см. например, [Скорняков Л. А., 1982, § 1, теорема 9]). □

Все варианты значений $\min, \max$ для отношений приведены в табл. 4, строки которой отвечают исходному отношению $\mathbf{R}$, а столбцы – обратному отношению $\mathbf{R}^{-1}$.

Следующая теорема отвечает на вопрос совместимости значений операторов $\min, \max$ для отношения и обратного отношения.

Теорема. Для ячеек табл. 4, обозначенных +, существуют отношения с соответствующими значениями $\min, \max$. Для ячеек табл. 4, обозначенных –, не существуют отношения с соответствующими значениями $\min, \max$. □

Доказательство основывается на предыдущих леммах. Так, заполнение первой строки и первого столбца непосредственно вытекает из п. 5 леммы 1 (о характеристическом свойстве пустого отношения). Отметим только, что соответствующие отношения строятся объединением конечных универсальных отношений (лемма 2) с использованием леммы 4 для счетных объединений.

Табл. 4 – Все варианты значений $\min, \max$ и их совместимость для отношения и обратного к нему отношения

		$l' \overset{def}{=} \min(\mathbf{R}^{-1})$, $l \overset{def}{=} \max(\mathbf{R}^{-1})$					
		$l' = l = 0$	$l' = l > 0$	$l' = 0, l \geq 1$	$l' = 0, l = \infty$	$l' \geq 1, l \geq l'$	$l' \geq 1, l = \infty$
$k' \overset{def}{=} \min(\mathbf{R})$ $k \overset{def}{=} \max(\mathbf{R})$	$k' = k = 0$	+	–	–	–	–	–
	$k' = k > 0$	–	+	+	+	+	+
	$k' = 0, k \geq 1$	–	+	+	+	+	+
	$k' = 0, k = \infty$	–	+	+	+	+	+
	$k' \geq 1, k \geq k'$	–	+	+	+	+	+
	$k' \geq 1, k = \infty$	–	+	+	+	+	+

Отметим, что по построению табл. 4 ее заполнение симметрично (относительно главной диагонали), поскольку при смене строк на столбцы (либо наоборот) исходное отношение и обратное к нему просто меняются ролями; таким образом, доказательство требуется только для заполнения ячеек, находящихся на главной диагонали и выше неё. □

Из этой теоремы следует, что, за исключением тривиального случая пустого отношения, для любого распределения значений операторов min, max существует отношение, на котором указанные значения достигаются. В этом узком смысле нет логической связи между значениями операторов min, max на исходном и обратном отношениях. Причина этого состоит в том, что полные образы одноэлементных множеств несут локальную информацию о отношении (например, функциональность выражается, а инъективность уже не выражается.)

Выводы

В работе были рассмотрены и уточнены в терминах теории отношений такие понятия модели „сущность-связь": сущности, связи, структурные ограничения связей (показатель кардинальности, степень участия, структурные ограничения вида (min, max)).

После рассмотрения ограничений на типы связей (табл. 2-3) можно сделать вывод: структурное ограничение вида (min, max) является более выразительным, чем показатель кардинальности и степень участия сущности в связи.

Основное задание последующей работы –формализация таких понятий модели „сущность-связь": атрибуты, многосторонние связи, слабые и сильные сущности.

Литература

[Chen P., 1976]. The entity-relationship model – towards a unified view of data // ACM TODS. – March 1976. – v. 1, № 1.

[Batini Carlo, Ceri S., Navathe S.B. and Batini Carol, 1991]. Conceptual Database Design: an Entity/Relationship Approach, Addison-Wesley, Reading, MA, 1991.

[Thelheim B., 2000]. Fundamentals of Entity-Relationship Modeling, Springer-Verlag, Berlin, 2000.

[Гарсиа-Молина Г., Ульман Дж., Уидом Дж., 2004]. Системы баз данных. Полный курс.: пер. с англ. – Москва: Издательский дом „Вильямс", 2004. – 1088 с.

[Дейт Дж., 1998]. Введение в системы баз данных.: пер. с англ. – Киев: „Диалектика", 1998.– 784 с.

[Коннолли Т., Бегг К., Страчан А., 2000]. Базы данных: проектирование, реализация и сопровождение. Теория и практика, 2-е изд.: пер. с англ. – Москва: Издательский дом „Вильямс", 2000.– 1120 с.

[Крёнке Д., 2003]. Теория и практика построения баз данных. 8-е изд. – Санкт-Петербург: „Питер", 2003. – 800 с.

[Буй Д. Б., Кахута Н. Д., 2005]. Властивості теоретико-множинних конструкцій повного образу та обмеження // Вісник Київського університету. Серія: фіз.-мат. науки. – 2005. – Вип. 2. – С. 232-240.

[Биркгоф Г.,1984]. Теория решеток. – Москва: Наука, 1984. – 564 с.

[Редько В. Н., Брона Ю. Й., Буй Д. Б., Поляков С. А., 2001]. Реляційні бази даних: табличні алгебри та SQL-подібні мови. – Київ: Видавничий дім "Академперіодика", 2001. – 198 с.

[Скорняков Л. А., 1982]. Элементы теории структур. – Москва: Наука, 1982. – 158 с.

Информация об авторах

Дмитрий Буй – *Киевский национальный университет имени Тараса Шевченко, факультет кибернетики: Украина, Киев, 03022, пр. Глушкова 2, корп.6; e-mail: buy@unicyb.kiev.ua*

Людмила Сильвейструк - *Киевский национальный университет имени Тараса Шевченко, факультет кибернетики: Украина, Киев, 03022, пр. Глушкова 2, корп.6; e-mail: slm-klm@rambler.ru*

FORMAL DEFINITION OF ARTIFICIAL INTELLIGENCE AND AN ALGORITHM WHICH SATISFIES THIS DEFINITION [6]

Dimiter Dobrev

***Abstract**: In this paper you can find a formal definition of the notion of Artificial Intelligence. A corollary of this definition is an algorithm which satisfies it. It is suspicious that this is the first paper which gives description of the AI algorithm. Really, this is true but only from the point of view of theoretical computer science because this algorithm is useless for the practice due to the combinatory explosion. In theory, every program which terminates after finite number of steps is a terminating program but for the practice terminating programs are only those which terminate after reasonable number of steps. So, in this paper you will find an algorithm which is AI but which is not sufficiently efficient. The only thing which you have to do in order to obtain real AI is to optimise this algorithm.*

***Keywords**: AI Definition, Artificial Intelligence.*

Introduction

We will start with the formal definition of Artificial Intelligence. After defining this notion the question which is the algorithm which satisfies the definition will be obvious.

A definition of Artificial Intelligence was proposed in [1] but this definition was not absolutely formal at least because the word "Human" was used. In this paper we will formalize this definition.

The definition in [1] first was published in popular form in [2, 3]. It was stated in one sentence but with many assumptions and explanations which were given before and after this sentence. Here is the definition of AI in one sentence:

AI will be such a program which in an arbitrary world will cope no worse than a human.

From this sentence you can see that we assume that AI is a program. Also, we assume that AI is a step device and that on every step it inputs from outside a portion of information (a letter from finite alphabet Σ) and outputs a portion of information (a letter from a finite alphabet Ω). The third assumption is that AI is in some environment which gives it a portion of information on every step and which receives the output of AI. Also, we assume that this environment will be influenced of the information which AI outputs. This environment can be natural or artificial and we will refer to it as "World".

The ***World*** will be: one set ***S***, one element s_0 of ***S*** and two functions ***World(s, d)*** and ***View(s)***. The set ***S*** contains the internal states of the world and it can be finite or infinite. The element s_0 of ***S*** will be the world's starting state. The function ***World*** will take as arguments the current state of the world and the influence that our device exerts on the world at the current step. As a result, this function will return the new state of the world (which it will obtain on the next step). The function ***View*** gives the information what our device sees. An argument of this function will be the world's current state and the returned value will be the information that the device will receive (at a given step).

Life in one world will be any infinite row of the type: $d_1, v_1, d_2, v_2, \ldots$ where v_i are letters from Σ and d_i are letters from Ω. Also, there has to exist infinite row $s_0, s_1, s_2, \ldots$ such that s_0 is the staring state of the world and $\forall i>0\ v_i=View(s_i)$ and $\forall i\ s_{i+1}=World(s_i, d_{i+1})$. It is obvious that if the world is given then the life depends only on the actions of AI (i.e. depends only on the row $d_1, d_2, d_3, \ldots$).

In order to transform the definition in [1] and to make it formal, we have to define what is a program, what is a good world and when one life is better than another.

The first task is easy because this work is done by Turing in the main part. Anyway, the Turing definition of program is for a program which represents function, but here we need a transducer which inputs the row $v_1, v_2, v_3, \ldots$ and outputs the row $d_1, d_2, d_3, \ldots$. So, we will make a modification of the definition of Turing machine [6].

[6] This publication is partially supported by the Bulgarian Ministry of Education (contract 1102/2001)

Our second task is to say what a good world is. It was written in [1] that if you can make a fatal error in one world then this world is not good. What is world without fatal errors needs additional formalization.

The next problem is to say when one life is better than another. This is done in [1] but there are some problems connected with the infinity which have to be fixed.

The last task is to say how intelligent our program should be and this cannot be done by comparison with a human being.

What is a Program

We will define a program as a Turing machine [6]. Let its alphabet Δ consist of the letters from Σ, from Ω, from one blank symbol λ and from some service signs.

Let our Turing machine have finite set of internal states P, one starting state p_0 and a partial function $F: P \times \Delta \to P \times \Delta \times \{Left, Right\}$.

The Turing machine (TM) is a step device and it makes steps in order to do calculations. On the other hand, AI is a step device and its life consists of steps. In order to make distinction between these two types of steps we will call them small and big steps. When we speak about time we will mean the number of big steps.

Of course, our TM will start from the state p_0 with infinite tape filled with the blank symbol λ. How our TM will make one small step. If it is in state p and if its head looks at the letter δ then $F(p, \delta)$ will be a 3-tuple which first element is the new state after the small step, the second element will be the new letter which will replace δ on the tape and the third element will be direction in which the head will move.

How will our TM (which is also our AI) make one big step? This will happen when after a small step the new state of TM is again p_0. At this moment our TM has to output one letter d_i and to input one letter v_i. We will assume that the letter which is outputted is that which is written on the place of δ on this small step. But how after outputting the letter d_i will our TM input the letter v_i? We will put this letter on the place where the head after the small step is. In this way we are intervening in the work of the TM by replacing one symbol from the tape with another. The replaced symbol is lost in some sense because it will not influence the execution of the TM from this small step on.

We will assume that our TM is outputting only letters from Ω (no letters from the rest of Δ). Also, we assume that our TM never hangs. TM hangs if after reading some input $v_1, v_2, \dots, v_n$ it stops because it falls into some state p and its head falls on some letter δ such that $F(p, \delta)$ is not defined. TM also hangs if after reading of some input $v_1, v_2, \dots, v_n$ it makes infinitely many small steps without reaching the state p_0 (without making of big steps anymore).

After this we have a formal definition of a program. We have to mention that there is no restriction on the number of the small steps which TM needs to make for one big step. This number has to be finite but it is not restricted. Maybe it is a good idea to add one parameter ***Max_number_of_small_steps_in_AI*** in order to exclude some decisions for AI which are combinatory explosions. (If we restrict the number of small steps then we have to restrict also the number of service signs in Δ because we can speed up the TM by increasing the size of its alphabet.) If we want to use AI as a real program on a real computer then we have to take into consideration that the memory of the real computers is limited. So, we can restrict also the size of the tape. Anyway, we will not care about the efficiency of AI and we will not make such restrictions.

What is a World without Fatal Errors

It is very difficult to define what a world without fatal errors is. That is why we will do something else. We will restrict our set of worlds is such a way that the new set will contain only worlds without fatal errors.

Let our world look like one infinite sequence of games. Let every game be independent from the previous ones. Let us have three special letters in Σ, which we will call final letters. Let this letters be ***{victory, loss, draw}***. Let every game finish with one of the final letters. Let every game be shorter than 1000 big steps.

Remark 1: Our definition of AI will depend on many parameters. In order to simplify the exposition we will fix these parameters to concrete numbers. Such parameter is the maximum number of steps in a game which will be fixed to 1000. Also, in order to simplify the exposition we will use different numbers for different parameters.

Remark 2: The only parameters in our definition which are not numbers are the alphabets Σ and Ω. We will assume that these alphabets are fixed and that Ω has at least 2 letters and Σ has at least 2 letters which are not final. (If Ω has only one letter then there will be no choice for the action of AI and the world will be independent from this action. If Σ has only one letter, which is not final, then the game will be blind because AI will not receive any information until the end of the game. Therefore, the minimum for $|\Sigma|$ is 5.)

We will assume that the world has three special internal states ***{s_victory, s_loss, s_draw}***, which we will call final states. Let these states be indistinguishable from the state s_0 for the function ***World***. This means that the world will behave in the final states in the same way as if it was in the starting state. Let the function ***View*** distinguish the final states and return from them the letters ***victory***, ***loss*** and ***draw*** respectively. Also, the final states will be the only states on which the function ***View*** will return one of the letters ***{victory, loss, draw}***.

After the restriction of the definition of ***World***, we can be sure that there are no fatal errors in our world because the life in such a world is an infinite sequence of games and if we lose some games (finitely many) then this will not be fatal because every new game is independent from the previous ones. Also, we are sure that a new game will come sooner or later because every game is finite (i.e. previous game is shorter than 1000 steps).

When is One Life Better than Another

In [1] we gave the following definition for the meaning of the life: One life is better than another if it includes more good letters and fewer bad letters. Here good letters will be ***{victory, draw}*** and bad letters will be ***{loss, draw}***. So, here life is good if we win often and lose seldom.

We want to introduce one function ***Success*** which will evaluate with a real number every life in order to say how good it is. For that we will define first the function ***Success*** for the every beginning of life (all beginnings are finite). After that we will calculate the limit of ***Success*** when the size of the beginnings goes to infinity and this limit will be the value of ***Success*** for the entire life.

The function ***Success*** can be defined for the beginnings like the difference between the number of victories and the number of losses. This is not a good idea because then the limit of ***Success*** will possibly converge to infinity (plus or minus infinity). It is a better idea to calculate the percentage of victories. So, we define ***Success*** as ***(2.N_victory +N_draw)/ (2.N_games)***. Here ***N_victory*** is the number of victories (analogically for ***N_draw*** and ***N_games***). Function ***Success*** will give us a number between 0 and 1 for every beginning and its limit will be also between 0 and 1. The only problem is that ***Success*** may not have a limit. In such a case we will use the average between limit inferior and limit superior.

Trivial Decisions

Now we have a really formal definition of AI and this gives us the first trivial decision for AI.

TD1 will be the program which plays at random until the first victory. After that TD1 repeats this victory forever. For this TD1 needs only to remember what it did in the last game. If the last game was victorious then it can repeat this last game because the function World is deterministic and if TD1 is doing the same then the world will do the same too.

TD1 is perfect in all worlds in which the victory is possible. If the victory is not possible then TD1 will play at random forever. That is why we will make TD2 which will be perfect in all worlds.

TD2 will be this program which tries sequentially all possible game's strategies until it finds the first victory and after that repeats this victory forever. If there is no victorious game strategy then TD2 repeats the last draw game forever. If the draw game is not possible too then TD2 plays at random. (It is important that the game's length is not more than 1000. This means that the number of the game's strategies is finite.)

TD2 is perfect in all worlds and this means that it is a trivial decision on our definition for AI. Really, the definition stated that AI has to cope no worse than a human but for the perfect program this is true because it copes no worse than anything even no worse than a human being.

It is suspicious that such simple program like TD2 can satisfy our definition for AI. That is why we will change the definition by accepting more possible worlds. It is too restrictive to assume that the game is deterministic and every time you do the same the same will happen.

Nondeterministic Games

We will assume that the function ***World*** is not deterministic. It is better to say that it is multy-valued function, which chooses at random one of its possible values. Let every possible value correspond to one real number, which is the possibility for this value to be chosen. We will assume also that $\forall s\ \forall \omega$ ***World(s, ω)*** has at least one value and that $\forall s\ \forall \omega$ ***(*** for every two different values of ***World(s, ω)*** the function ***View*** returns different result***)***.

Remark 3: The latter means that if something nondeterministic happens this information will be given to AI immediately by the function ***View***. There is no sense to assume existence of a nondeterministic change which cannot be detected immediately but later or even which cannot be detected never.

Now we will ask the question what is the best strategy in such a world and we will offer a program, which will be almost perfect. Before that we need several definitions:

Definition 1: ***Tree of any game***. It will have two types of vertices. The root and the other vertices which are on even depth will be the vertices of type AI (because they correspond to the moments when AI has to do its choice). The vertices which are on odd depth will be vertices of the type world (because they correspond to the moments when the world will answer at random). From the vertices of type AI go out $|\Omega|$ arcs and to every such arc corresponds one of the letters from Ω. There is one exception. If the arc which is right before this vertex corresponds to a final letter, then this vertex is a leaf. From the vertices of type world go out $|\Sigma|$ arcs and to every such arc corresponds one of the letters from Σ. Here there is an exception again. If this vertex is on depth 1999, then only three arcs go out and these three arcs correspond to the final letters.

You can see that the tree of any game is finite and its maximum depth is 2000 (because games are not longer than 1000 steps). Nevertheless, there are leaves on any even depth between 2 and 2000.

Definition 2: ***Tree of any 100 games***. Let us take the tree of any game. Let us replace all of its leaves with the tree on any game. Let us repeat this operation 99 times. The result will be the tree of any 100 games (which is 100 times deeper than the tree of any game).

From the ***tree of any game*** we will receive ***Strategy for any game***. This will be its subtree which is obtained by choosing one vertex from the successors of every vertex of the type AI and deleting the rest successors (and their subtrees). Analogically we make ***Strategy for any 100 games*** like a subtree from the ***tree of any 100 games***. We have to mention that the ***Strategy for any 100 games*** can be different from repeating one ***Strategy for any game*** 100 times. The reason is because the strategy on the next game can depend on the previous games.

Definition 3: ***Tree of this game.*** For every concrete game (i.e. concrete world) we can construct the tree of this game as a subtree from the tree of any game. We will juxtapose internal states of the world to the vertices of type AI in the time of this construction. First, we will juxtapose the state s_0 to the root. Let k_0, k_1 and k_2 be vertices and let k_1 be successor of k_0 and k_2 be successor of k_1. Let k_0 be vertex of type AI and let the state s be juxtaposed to it. Let the letters ω and ε be juxtaposed to the arcs $< k_0, k_1 >$ and $< k_1, k_2 >$. In this case if $\varepsilon \neq$ ***View(World(s, ω))*** for every value of ***World(s, ω)*** then we delete the vertex k_2 (and its subtree). In the opposite case we juxtapose k_2 to this value of ***World(s, ω)*** for which $\varepsilon =$ ***View(World(s, ω))***. This value is only (look at remark 3). Also, we will juxtapose the possibility ε to be the value of ***View(World(s, ω))*** to the arc $< k_1, k_2 >$. So, one letter and one possibility will be juxtaposed to the arc $< k_1, k_2 >$.

Analogically to the strategy for any game we can make strategy for this game. We have to say that if the World is deterministic (i.e. every vertex of type world has only one successor) then the strategy for this game is a path (a tree without branches). In this case the paths in the tree of this game are exactly the strategies for this game. This was used from TD2 in order to try all strategies.

Max-Sum Algorithm

For every vertex of the tree of this game we can calculate the best possible success (this is our expectation for success, if we play with the best strategy from that vertex on).

1. The best possible success for the leaves will be 1, 0 and 1/2 for the states ***s_victory***, ***s_loss*** and ***s_draw*** respectively.

2. If the vertex is of type AI, then its best possible success will be the maximum from the best possible successes of its successors.

3. If the vertex is of type world, then its best possible success will be the sum Σ ***Possibility(i). BestPossibleSuccess(i)***. Here ***i*** runs through all successors of this vertex.

The algorithm for calculating the best possible success can also be used to calculate the best strategy in this game (the best strategy can be more than one). This algorithm looks like the Min-Max algorithm, which we use in chess. Anyway, this is different algorithm, to which we will refer as Max-Sum algorithm. The difference is essential because in the chess we assume that we play against someone who will do the worst thing to us (remark 4). Anyway, in the arbitrary world we cannot assume that the world is against us. For example, when you go to work you go first to the parking lot in order to take your car. If your car is stolen, then you go to the bus stop in order to take the bus. If every time you were presumed the worst case, then you would go directly to the bus stop.

New Trivial Decisions

Now we can calculate the best possible success for any game and we will give the next trivial decision (TD3), which will do the best in every game. This means that the success of TD3 for one world will be equal to its best possible success.

TD3 will be the program which plays at random for long time enough. In this time TD3 collects statistical information for the ***tree of this game*** and builds inside its memory this tree together with the values of all possibilities. After that time TD3 starts playing by the use of Max-Sum algorithm.

TD3 gives the perfect decision in any world but TD3 is impossible because we cannot say when enough statistical information is selected. Anyway, possible is something which is a little bit worse. For every $\varepsilon > 0$ we will make TD4, which for every world will make success on a distance no more than ε from the best possible.

TD4 will be this program which simultaneously collects statistical information for the ***tree of this game*** and in the same time plays by the use of Max-Sum algorithm on the base of statistics, which is collected up to the current moment. In order to collect statistics TD4 makes experiments which contradict to the recommendations of Max-Sum algorithm. Such experiments are made rarely enough to have success on a distance not bigger than ε from the best possible success.

We can choose the value of ε to be as small as we want. Anyway, the price for the small value of ε is the longer time for education (because of rare experiments). We will call the parameter ε "courage". Here we receive a surprising conclusion that if AI is more cowardly it is closer to perfection (this is true only in the case of infinite life).

TD4 is a decision for our definition of AI because it is only on ε distance to perfection unlike the people who are much farther from perfection. We have to mention that in some sense TD4 is not as trivial as TD2, because TD4 represents awful combinatory explosion in the execution time (number of small steps) and in the memory size. Anyway, we said that we will not care about the efficiency of AI for the moment. On the other hand, there is one additional problem, which is present in both TD2 and TD4, and which makes them both useless. This is the problem of the combinatory explosion of the educational time. Imagine that you are playing chess at random against deterministic partner. How long will you need to make accidental victory? Or in case your partner is not deterministic. How long will you need to play all possible game's strategies and try each one several times in order to collect statistical information on how your partner reacts in every case?

Finite Life

In some sense TD2 and TD4 are extremely stupid because they need extremely long time for education. Really, educational time and level of intelligence are two different parameters of the mind and if one of them is better then this can be at the expense of the other. For example, a human being needs about a year to learn to walk, which is much worse in comparison to most animals. Some of the greatest scientists had bad results in school, which can be interpreted as a fact that they advanced slower than the ordinary people.

Therefore, the educational time is important and it has to be limited in order to make our definition useful. This will be done by changing the life length from infinite to finite. We will assume that the length of the life is 100 games. Each game has maximum 1000 steps, which means that the life length is not bigger than 10,000 steps. Now the

success of the life will not be the limit of the ***Success*** function but the value of this function for the first 100 games.

After this we can look for program which makes a good success in an arbitrary world, but this is not a good idea because the arbitrary world is too unpredictable. Human beings use the assumption that the world is simple and that is why they cope very well in a more simple environment and they are totally confused if the environment is too complicated. Therefore, we have to restrict the complexity of the world and give bigger importance to the more simple worlds. For this restriction we will use Kolmogorov Complexity [5]. The parameter which restricts the complexity of the world will be the level of intelligence of AI.

Kolmogorov Complexity

First we need a definition of program which calculates the functions World and View. For this we will use the same definition of TM as for the program which was our AI. There will be some small differences: The alphabet of the Turing Machine of the world (TM_W) will be $\Sigma \cup \Omega \cup \{\lambda\}$ (the only service symbol will be λ). Also, TM_W will input the row $d_1, d_2, d_3, \ldots$ and output the row $v_1, v_2, v_3, \ldots$. At the beginning TM_W will start with tape on which d_1 is at the head position and the rest is λ. At the end of the first big step TM_W will output v_1 and input d_2. F will be set of 5-tuples which is a subset to $P \times \Delta \times P \times \Delta \times \{Left, Right\}$. This means that F is not a function but a relation (because it will represent multy-valued function). We will assume that $\forall s \ \forall \delta \ \exists 5\text{-tuple} \in F$ whose first two elements are s and δ (this makes the reasons for hanging with one less). The 5-tuples in F whose third element is p_0 will be called output 5-tuples. The fourth element of output 5-tuples has to be letter from Σ (this is not necessary but sufficient condition in order TM_W to output only letters from Σ). We will allow nondeterministic behavior only for output 5-tuples. This means that if two different 5-tuples have the same first and second elements then they both have to be output 5-tuples. There will be no two 5-tuples which differ only at the fifth element (we cannot have a choice between two nondeterministic 5-tuples which output the same letter - look again at remark 3). It will be more interesting if we assume that nondeterministic 5-tuples have additional parameter which shows the possibility for each of them to be chosen. Nevertheless, we will assume that this possibility is distributed equally and that we do not have such additional parameter.

According to this definition, the internal states of the world will be the states of the tape of the TM_W. If we want to have worlds without fatal errors we have to clean the tape of TM_W after each game (after printing any final letter). Nevertheless, we will not do this because the absence of fatal errors was important when we had infinite life and when we counted on that sooner or later all errors will be compensated. For real AI is better to assume some connection between games. Otherwise AI will not remember what was done in the last game or it will remember it but it will not know whether this was in the last game or in some other game from the previous ones.

Another question is what we will do with TM_W which hangs. We do not want to exclude these programs from our definition (at least because we cannot check this characteristic). That is why we will assume that if one TM_W makes more than 800 small steps without making a big step then we will interrupt it with output "draw". This means that it will do the next small step in the same way as if the 5-tuple executed at this moment had third element p_0 and fourth element "draw". Also, if one TM_W makes 1000 big steps without outputting any final symbol then the output of the next big step will be "draw". We need this in order to keep the games finite, which is important in order to keep the life finite (the life is 100 games).

We will define the size of TM_W as the number of internal states plus the level of indefiniteness (this is the minimal number of nondeterministic 5-tuples, which have to be deleted from F in order to make it deterministic or this is the number of all nondeterministic 5-tuples minus the number of different groups of nondeterministic 5-tuples).

So, we will restrict the set of possible worlds to those generated by Turing Machine whose size is not bigger than 20. The maximum size of the TM_W will be the level of intelligence of AI. The simpler worlds will be more important because they are generated from more than one TM_W and that is why we will count their result more than once.

Remark 4: It looks like that two Turing machines (the world and AI) play against each other. Anyway, this is wrong because the world does not care about AI and it dose not play against AI.

Final Definition of AI

Now everything is fixed. We have finite lives, which are exactly 100 games. We had selected the success function that will evaluate these lives. Also, we made a finite set of worlds which consist of the worlds generated from the TM_W with size not bigger than 20. Now we can define AI as this program which will make the best average success in the selected worlds. Such program exists and it will be the next trivial decision (TD5).

The number of all strategies for playing 100 consecutive games is finite. The number of selected worlds is also finite. We can calculate the expected success of any strategy in any world. The average success of a strategy will be the arithmetical mean from its expected success in any world. (The calculation of the expected success of a strategy in a world is easy if the world is deterministic. In this case, we will have simply to play 100 games with this strategy in this world. In the opposite case, if the world is nondeterministic then we have to use Max-Sum algorithm, which is theoretically possible, but in practice it is a combinatory explosion. Nevertheless, even if the worlds were only deterministic, we would have combinatory explosion again from the number of worlds and from the number of strategies.)

Hence, TD5 will be this program which calculates and plays the best strategy (this which average success is biggest). Such program is easy to be written but it is very difficult to wait until it makes its first step. The situation with the perfect chess playing program is analogical. (It plays chess by calculating all possible moves.) This program can be written very easy but the time until the end of the universe will be not enough for it to make its first move.

It will be too restrictive if we define AI as the best program (such as TD5 or such as any other program equivalent to TD5). It will be better if we say that AI is a program whose average success is not more than 10% from the best (from TD5). Such definition is theoretically possible, but practically inconvenient. The reason for this is the fact that the value of the average success of TD5 can be theoretically calculated, but in practice this is absolutely impossible. So, if we select such definition we will not be able to check it for a concrete program.

Final definition: **AI will be a program which makes more than 70% average success in the selected set of worlds.**

Assumption 1: Here we assume that the average success of TD5 is about 80%. If this conjecture is true then there exists a program which satisfies the definition (at least TD5 do). If the average success of TD5 is smaller than 70% then there is no such a program (of course, in such case we can change this parameter and make it smaller than 70%).

The advantage of this definition of AI is that we can check it for a concrete program. Of course, we cannot calculate the average success for any program due to the combinatory explosion, but we can calculate it approximately by the methods of the statistics. For this, we will select at random N worlds (world is TM_W with size not bigger than 20) and we will play 100 consecutive games in every world. If the world is deterministic then this will give us the expected success of our program in this world. If it is not deterministic then we will play at random in this world. This will give us statistically good evaluation of the expected success because the possibility to be extremely lucky in 100 games is very small (so is the possibility to be extremely unlucky). If N (the number of the tested worlds) is big then the statistical result will be close to the average success of our program.

If $|\Sigma \cup \Omega \cup \{\lambda\}| = 5$ (which is the minimum - remark 2) then the number of deterministic TM_W with 20 states is 200 on power of 100. If we take the number of nondeterministic TM_W with 19 states and level of indefiniteness one (which means with two nondeterministic 5-tuples) then this number is many times smaller than 200 on power of 100. In order to use the method of statistics we have to calculate how many times this number is smaller. Otherwise we will use wrong correlation between deterministic and nondeterministic TM_W. Anyway, such wrong correlation will make an unessential change in the definition of AI.

Conclusion

Now we have definition of AI and at least one program (TD5) which satisfies it (with assumption 1). The first question is: Does this definition satisfy our intuitive idea that AI is a program which is more intelligent than a human being. Yes, but for some values of the parameters educational time and level of intelligence. In this paper the educational time was fixed on 100 games each of them no longer than 1000 steps (educational time is equal to the life length because we learn all our life). The level of intelligence here was fixed on 20. Which means that

we assume that we can find a model of the world which is TM_W with size not bigger than 20. We cannot say what is the exact level of intelligence of the human being.

The second question is: Is TD5 which satisfies the definition the program which we are looking for. The answer is definitely no. We are looking for a program which can work in real time.

TD5 is like the perfect chess playing program. All other chess playing programs play worse than the prefect one. Anyway, the perfect chess program is useless because it cannot play in real time. In the same way TD5 is perfect but useless. We need AI which is not perfect but which can play in real time.

Bibliography

[1] Dobrev D. D. A Definition of Artificial Intelligence. In: Mathematica Balkanica, New Series, Vol. 19, 2005, Fasc. 1-2, pp.67-74.

[2] Dobrev D. D. AI - What is this. In: PC Magazine - Bulgaria, November'2000, pp.12-13 (in Bulgarian, also in [4] in English).

[3] Dobrev D. D. AI - How does it cope in an arbitrary world. In: PC Magazine - Bulgaria, February'2001, pp.12-13 (in Bulgarian, also in [4] in English).

[4] Dobrev D. D. AI Project, http://www.dobrev.com/AI

[5] Kolmogorov A. N. and Uspensky V. A. Algorithms and randomness. - SIAM J. Theory of Probability and Its Applications, vol. 32 (1987), pp.389-412.

[6] Turing, A. M. On Computable Numbers, with an Application to the Entscheidungsproblem. Proceedings of the London Mathematical Society, Series 2, 42, 1936-37, pp.230-265.

Author's Information

Dimiter Dobrev - *Institute of Mathematics and Informatics, BAS, Acad.G.Bonthev St., bl.8, Sofia-1113, Bulgaria; P.O.Box: 1274, Sofia-1000, Bulgaria; e-mail: d@dobrev.com*

СИСТЕМНЫЙ АНАЛИЗ МОДЕЛИ ЛЕОНТЬЕВА ПРИ НЕЧЁТКО ЗАДАННЫХ ПАРАМЕТРАХ МЕТОДОМ БАЗИСНЫХ МАТРИЦ

Владимир Кудин, Григорий Кудин

Аннотация. *Предложено применение метода базисных матриц для анализа модели Леонтьева (МЛ). МЛ можно интерпретировать как задачу прогноза затрат-выпуска продукции на основе известной статистической информации о значениях элементов технологической матрицы при нечётко заданном векторе ограничений и границах переменных. Вектор правых частей ограничений МЛ может испытывать изменения и в таком случае получается динамический аналог задачи. Существенным осложнением МЛ является включение ограничений на переменные и вектор целевой функции.*

Ключевые слова: *модель Леонтьева, количественный и качественный анализ, нечёткое множество, базисная матрица, функция принадлежности.*

Введение

Модель Леонтьева (МЛ) была сформулирована, как задача прогноза затрат-выпуска продукции на основе известной статистической информации о значениях элементов технологической матрицы при нечётко заданном векторе ограничений и границах переменных [Леонтьев, 1972]. Нечёткость значений параметров модели предопределяет наличие в контуре принятия решения экспертов (ЛПР), которые призваны качественно определить структуру модели, указать механизм устранения неопределённостей и разногласий при ее формировании [Гасс., 1961].

Существенным осложнением МЛ есть включение ограничений границ на переменные [Орловский, 1981]. При исследованиях, наряду с классической МЛ типа системы линейных алгебраических уравнений с квадратной невырожденной матрицей ограничений (СЛАР), рассматривается и более обобщённая модель, которая математически интерпретируется как система линейных алгебраических неравенств с соответствующей матрицей ограничений (СЛАН), а также и как задача линейного программирования (ЗЛП) [Волошин, 1993], [Войналович, 1987], [Войналович, 1988], [Кудин, 2002]. При построении МЛ, после проведения качественного наполнения и анализа модели [Орловский, 1981]), нужно провести количественный анализ непротиворечивости структурных элементов [Волошин,1993], [Войналович, 1987], [Войналович, 1988], [Кудин, 2002], что математически включает следующие стадии:

- проверки невырожденности матрицы ограничений и установление ранга матрицы системы;
- направленной коррекции ранга матрицы ограничений изменением отдельных её элементов;
- установление совместных зависимостей МЛ и ограничений на переменные – разрешимость (неразрешимость);
- анализ статуса ограничений МЛ для многогранного множества ограничений на переменные, проведения, при необходимости, направленных изменений;
- нахождение решений при разрешимости за совместностью;
- установление свойства единственности или неединственности решений.

Постановка задачи

Пусть имеем МЛ вида

$$u = A^T\, u^T + y^T, \tag{1}$$

где $A^T = \left\{a_{ij}\right\}_{\substack{i=\overline{1,m} \\ j=\overline{1,m}}}$ невырожденная квадратная матрица размерности $(m \times m)$,

$u = (u_1, u_2, ..., u_m)$ - вектор переменных, для которого выполняется: $u \in U_{\approx},\ U_{\approx} = \bigcup\limits_{\lambda \in [0,1]} U_\lambda$,

$$U_\lambda = \left\{ u \,/\, \prod_{i=1}^{m} U_\lambda^{(i)} = \prod_{i=1}^{m} [U_\lambda^{(-)}, U_\lambda^{(+)}],\ 1 \geq \mu_{x_i}(u) \geq \lambda,\ u \in [U_i^{(-)}, U_i^{(+)}] \subseteq (-\infty, +\infty) \right\},$$

$Y^T = (y_1, y_2, ..., y_m)$ - вектор ограничений со свойством $Y \in Y_{\approx},\ Y_{\cong} = \bigcup\limits_{\lambda \in [0,1]} Y_\lambda$,

$$Y_\lambda = \left\{ y \,/\, \prod_{i=1}^{m} Y_\lambda^{(i)} = \prod_{i=1}^{m} [Y_\lambda^{(-)}, Y_\lambda^{(+)}],\ 1 \geq \mu_{x_i}(y) \geq \lambda,\ y \in [Y_i^{(-)}, Y_i^{(+)}] \subseteq (-\infty, +\infty) \right\},$$

$\mu_{x_i}(u),\ \mu_{y_i}(y),\ i \in I$ - заданные кусочно-линейные функции принадлежности [Орловский, 1981] с областью значений $[0,1]$, областью определения $(-\infty, +\infty)$, $i \in I = (1, 2, ...m)$ - индексы строк и столбцов матрицы ограничений, Т - знак транспонирования. Модель (1) исследуется в эвклидовом пространстве E^m. При наличии в контуре принятия решения экспертов (ЛПР) фаза качественного анализа (1) определяет следующую задачу количественного анализа при указанных уровнях $\lambda^{(p)},\ p = \{1, 2, ...p\}$ последовательности задач (1) при дополнительных ограничениях вида

$$U \in \{ \prod_{i=1}^{m} U_з^{(i)} = \prod_{i-1}^{m} [U_{i(p)}^{(-)}, U_{i(p)}^{(+)}], p \in P \},\ Y \in \left\{ \prod_{i=1}^{m} Y_з^{(i)} = \prod_{i-1}^{m} [Y_{i(p)}^{(-)}, Y_{i(p)}^{(+)}], p \in P \right\} \tag{2}$$

Предложена схема применения методологии последовательного анализа [Волошин, 1987] и метода базисных матриц (МБМ) [Кудин, 2002] для проведения количественного анализа разрешимости за совместимостью и свойств ограничений (активности - пассивности) задачи (1)-(2) в зависимости от уточнения значений параметров на стадии качественного анализа модели.

Основные положения метода базисных матриц (МБМ)

Модель вида (1) можно преобразовать к эквивалентной СЛАР вида $A^T u^T = C^T$.

Введем соответственно системе (1), изменением знака (= на $\le$), СЛАН

$$A^T u^T \le C^T . \qquad (3)$$

Модели СЛАН (3) и СЛАР типа (1) могут исследоваться при наличии целевой функций вида

$$\max_{u \in R^m} Bu , \qquad (4)$$

как задача анализа модели линейного программирования, где в (1) - (3) $B = (b_1, b_2, ..., b_m),\ C^T = (c_1, c_2, .., c_n),\ u^T = (u_1, u_2, .., u_m),\ a_j = (a_{j1}, a_{j2}, ..., a_{jm}),\ j = (1, 2, .., n)$ - строки матрицы A^T.

Предметом исследования будет:

- установление ранга системы (1) и разрешимости задачи (1), (2);
- анализ статуса ограничений (1) для многогранного множества (2);
- нахождение решения (1), (2) при разрешимости за совместимостью;
- установление условий разрешимости (неразрешимости) за совместимостью системы (1)-(2),
- (2) - (3);
- установление условий существования, единственности и неединственности решения СЛАН (3);
- исследование свойств, построенных на основании (1), (2), оптимизацонных задач (3),(4).

В предлагаемом МБМ используются строчные базисные матрицы [Войналович, 1987], [Войналович, 1988], [Кудін, 2002]. Базисные матрицы в ходе итераций решения задачи последовательно изменяются вводом-выводом из нее строк-нормалей ограничений.

В общем случае количество ограничений превышает количество переменных вида (4) при ограничениях:

$$A^T u^T \le C^T,\ \ u \in R^m, \qquad u \in R^m . \qquad (5)$$

Определение 1. Подматрицу $A_б$ матрицы A^T, составленную из m линейно независимых нормалей ограничений (5), будем называть базисной, а решение соответствующей ей системы уравнений $A_б u_0^T = C^0$ базисным. Две базисные матрицы отличающимися одной строкой будем называть сопредельными.

Пусть $\beta_{ij},\ \ i, j = 1, 2, ..., m,$ элементы базисной подматрицы $A_б$, e_{ri} - элементы матрицы $A_б^{-1}$, обратной к $A_б$; $e_k = (A_б^{-1})_k$.- столбец обратной матрицы. Решение $u_0 = (u_{01}, u_{02}, ..., u_{0m})$ системы уравнений $A_б u^T = c^0$, где c^0 - подвектор С, компоненты которого состоят из правых частей ограничений (5), нормали которых образуют базисную матрицу $A_б$; $\alpha_r = (\alpha_{r1}, \alpha_{r2}, ..., \alpha_{rm})$ - вектор разложения нормали ограничения $a_r u_1 \le c_r$ за строками базисной матрицы $A_б$, $\alpha_0 = (\alpha_{01}, \alpha_{02}, ..., \alpha_{0m})$ вектор разложения нормали целевой функции (4) за строками базисной матрицы $A_б$, $\Delta_r = a_r u_0^T - c_r$ - невязка r-го ограничения (5) в вершине u_0; $J_б, J_H,\ J = J_б \cup J_H$ - множества индексов базисных и небазисных ограничений (5). Установим формулы связи базисного решения, коэффициентов разложения нормалей ограничений и целевой функции (4), коэффициентов обратной матрицы, невязок ограничений и значений целевой функции при переходе к базисной матрице $\overline{A}_б$, которая образуется из матрицы $\overline{A}_б$ заменой ее строки a_k на a_l, которая не входит в базисную матрицу $A_б$. В новой базисной матрице $\overline{A}_б$ введённые величины будем называть элементами или компонентами метода базисных матриц и будем обозначать черточкой сверху, т.е. $\overline{\beta}_{ij},\ \ \overline{\alpha}_r,\ \ \overline{\Delta}_k,\ \ \overline{e}_{ri},\ \ \overline{\alpha}_0$. Вершины (0-грани), ребра (1-грани), к-грани, нормали, гиперплоскости и полупространства, многообразия будем называть структурными элементами модели.

При нахождении формул и основных соотношений между элементами метода при переходе от одной базисной матрицы к следующей считаем, что $a_{i1}, a_{i2}, ..., a_{im}$ - нормали ограничений,

$a_j u^T \le c_j, \quad j \in J_б$, где $J_б = \{i_1, i_2, ..., i_m\}$ - индексы ограничений, нормали которых образовывают строки базисной матрицы $A_{б,}$ a_l - нормаль ограничения $a_l u \le c_l, \quad \alpha_l = (\alpha_{l1}, \alpha_{l2}, ..., \alpha_{lm})$ - коэффициенты разложения вектора a_l за строками базисной матрицы $A_{б,}$.

Лемма 1. (Критерий линейной независимости системы векторов). Необходимым и достаточным условием линейной независимости системы векторов $a_{i_1}, a_{i_2}, ..., a_{i_{k-1}}, a_l, a_{i_{k+1}}, ..., a_{i_m}$, образованной заменой вектора a_{i_k}, который занимает k-ю строку в базисной матрице $A_б$, вектором a_l, есть выполнение условия $\alpha_{lk} \neq 0$.

Теорема 1. (О связи между смежными базисными матрицами). Между коэффициентами разложения нормалей ограничений (5) и целевой функции (4) за строками базисной матрицы, элементами обратных матриц, базисными решениями, невязками ограничений (5) и значениями целевой функции для двух смежных базисных матриц имеют место соотношения

$$\bar{\alpha}_{rk} = \frac{\alpha_{rk}}{\alpha_{lk}}, \quad \bar{\alpha}_{ri} = \alpha_{ri} - \frac{\alpha_{rk}}{\alpha_{lk}}\alpha_{li,} \quad r = \overline{0,n}; \quad i = \overline{1,m}; \quad i \neq k; \tag{6}$$

$$\bar{e}_{rk} = \frac{e_{rk}}{\alpha_{lk}}, \quad \bar{e}_{ri} = e_{ri} - \frac{e_{rk}}{\alpha_{lk}}\alpha_{li,} \quad r = \overline{1,m}; \quad i = \overline{1,m}; \quad i \neq k; \tag{7}$$

$$\bar{u}_{0j} = u_{0j} - \frac{e_{jk}}{\alpha_{1k}}\Delta_l, \quad j = \overline{1,m}, \tag{8}$$

$$\bar{\Delta}_k = -\frac{\Delta_l}{\alpha_{lk}}, \quad \bar{\Delta}_r = \Delta_r - \frac{\alpha_{rk}}{\alpha_{lk}}\Delta_l, \quad r = \overline{1,n}; \quad r \neq k; \tag{9}$$

$$\bar{Bu}_0^T = Bu_0^T - \frac{\alpha_{0k}}{\alpha_{lk}}\Delta_l, \tag{10}$$

причем условием того, что матрица остаётся базисной при замещении вектором нормали a_l k-ой строки базисной матрицы $A_б$, есть выполнение условия $\alpha_{lk} \neq 0$, условием допустимости опорного базисного решения есть $\alpha_{lk} < 0$, роста значений целевой функции $\alpha_{0k} < 0$.

Доказательство леммы 1 и теоремы 1 основывается на теоретических положениях, изложенных в [Войналович, 1987], [Войналович, 1988], [Кудин, 2002].

С учётом соотношений (6)-(10) будет строиться схема типа метода поиска не только оптимального решения последовательными переходами согласно допустимым базисным вершинами, а и проведение анализа свойств линейных моделей - метод базисных матриц.

Определение 2. Допустимое базисное решение u_0 оптимальное, если $Bu_0^T \ge Bu^T$ для всех u, которые удовлетворяют (5).

Теорема 2. (Критерий оптимальности решения). Для оптимальности базисного решения u_0 необходимо и достаточно неотрицательности коэффициентов разложения вектора нормали целевой функции (4) по строкам базисной матрицы $A_б$, т.е. $\alpha_{ok} \ge 0$ для всех $k = \overline{1,m}$.

Справедливость критерия оптимальности вытекает из формулы (10) теоремы 1.

Теорема 3. (О неограниченности значений целевой функции). Если существует индекс k такой, что $\alpha_{0k} < 0$ и $\alpha_{rk} \ge 0$ для всех небазисных r, то целевая функция задачи приобретает неограниченные значения на множестве допустимых решений.

Полное доказательство теоремы приведено в [Войналович, 1987].

Теорема 4. Для существования единственного решения (1) необходимо и достаточно, чтобы $\alpha_{lk}^{(i)} \neq 0, \ i = \overline{1,m}$, где $\alpha_{lk}^{(i)}$ ведущие элементы симплексной итерации МБМ по замещению строк базисной матрицы (2) нормалями ограничений (1).

Следствие 1. Матрица А основной системы (1) невырожденная, если $\alpha_{lk}^{(i)} \neq 0,\ i=\overline{1,m}$.

Следствие 2. Ранг системы (1) определяется количеством корректных замещений строк матрицы ограничений вспомогательной системы векторами нормалями (1), согласно формулам (6)-(10).
Справедливость теоремы 2 и следствий 1,2 является непосредственным следствием леммы 1.

Теорема 5. Оптимизационная задача (3),(4) с квадратной невырожденной матрицей ограничений имеет единственное решение тогда и только тогда, если $\alpha_{0i} > 0,\ i=\overline{1,m}$.

Теорема 6. Необходимым и достаточным условием неединственности решения задачи (3),(4) есть $\exists\, i$ таких, что $\alpha_{0i} = 0$, причём множество решений имеет ребра неограниченности.

Пусть для a_l ограничения $a_l u^T \le c_l$ относительно $A_{\bar{б}}$, выполняется $\alpha_{lk} = 0$, а для возмущения $(a_l + a_l^{'})u^T \le c_l + c_l^{'}$, $\bar{\Delta}_l = \Delta_l + \Delta_l^{"}$ $\bar{\alpha}_l = (\alpha_l + \alpha_l^{'}) = (a_l + a_l^{'})A_{\bar{б}}^{-1}$.

Теорема 7. Необходимым условием невирожденности новой $\bar{A}_{\bar{б}}$, образованной замещением нормали a_l, которая занимает k- ю строку в базисной матрице нормалью $(\epsilon_l + \epsilon_l)$ является выполнения условия $\exists\ i\ \ e_{ik} \neq 0$, где e_{ik} элемент $A_{\ddagger}^{-1}$ такой, что $\alpha_{li}^{'} \neq 0$. Решение будет определяться соотношением $\bar{u}_{oj} = u_{oj} - \frac{e_{jk}}{\tilde{\alpha}_{lk}}\bar{\Delta}_l$, причем если $\bar{\Delta}_l = 0$, то $\bar{u}_{oj} = u_{oj}$, $j=1,m$.

Теорема 7 дает условия направленного восстановления свойства невырожденности базисной матрицы (1).

Следствие 3. Если ведущий элемент симплексной итерации $\alpha_{lk} \neq 0$, то при возмущении, сохраняющего невырожденность, должно выполняться $\alpha_{lk}^{'} + \alpha_{lk} \neq 0$.

Пусть $U = \{\ u / a_j u^T \le c_j,\quad j \in J\ \},\ U_r = \{u / a_j u^T \le c_j,\quad j \in J,\quad J \neq r\ \}$

Определение 3. Ограничение $a_r u^T \le c_r$ пассивное, если $U = U_r$.

Определение 4. Ограничение $a_r u^T \le c_r$ порождующее неразрешимость за несовместностью, если $U = \varnothing,\ U_r \neq \varnothing$.

Теорема 8. (Критерий пассивности). Для того, чтобы ограничение $a_r u^T \le c_r$ было пассивным для (5), необходимо и достаточно существование базисной матрицы $A_б$, относительно которой разложение $\alpha_{rk} \ge 0$ для всех k.

Введем в рассмотрение

$\Pi_r^{(0)} = \{\ u / a_r u^T = c_r\ \},\qquad \Pi_r^{(+)} = \{\ u / a_r u^T \ge c_r\ \}$

$\Pi_r^{(-)} = \{\ u / a_r u^T \le c_r\ \}$

образованные с использованием гиперплоскости $a_r u^T = c_r$.

Теорема 9. (Критерий неразрешимости за несовместностью). Для несовместности множества (5) $(U = \varnothing,\ U_r \neq \varnothing,\ \ U = U_r \cap \Pi_r^{(-)})$ необходимо и достаточно существование $A_{\ddagger}$ и u_0 таких, что $\alpha_{ri} \le 0,\quad i=\overline{1,m}$ и $\Delta_r = a_r u_0^T - c_r > 0$.

Доказательство теорем 8, 9 приведено в [Войналович, 1987].

Построим с использованием ограничения $a_r u^T \le c_r$ следующие модели линейного программирования

$$\max\{a_r u^T / u \in U_p\}, \quad (11)$$

$$\min\{a_r u^T / u \in U_p\}, \quad (12)$$

где $c_{rH(p)}$, $c_{rB(p)}$, $u_{rH(p)}, u_{rB(p)}$, $A^{-1}_{rH(p)}$, $A^{-1}_{rB(p)}$ значение целевых функций, оптимальных решений, оптимальные базисные матрицы (13), (14), а $\alpha_{(H)rk} \ge 0,\ k=\overline{1,m},\ \alpha_{(H)rk} \le 0,\ k=\overline{1,m}$ - коэффициенты разложения нормалей целевых функций по строкам соответствующих базисных матриц для (11) и (12), $U_p,\ p=\overline{1,k}$ множество, соответствующее ограничениям на переменные (2).

Следует отметить, что условия пассивности и неразрешимости за несовместностью существенно упрощаются для множества (2) - гиперпараллелепипеда переменных, поскольку решения задач (11) и (12) формируются непосредственно - без решения оптимизационной задачи.

Пусть $\delta = \min\limits_{j\in J} \sqrt{\sum\limits_{j=1}^{m} (\frac{u_{j(B)} - u_{j(H)}}{2})^2}$ -минимальный радиус сферы, вписанной в П, $R = \sqrt{\sum\limits_{j=1}^{m} (\frac{u_{j(B)} - u_{j(H)}}{2})^2}$ - радиус сферы, описанной вокруг П с центром в точке $u_{0(sr)} = (\frac{u_{1(B)} + u_{1(H)}}{2}, \frac{u_{2(B)} + u_{2(H)}}{2}, \ldots \frac{u_{m(B)} + u_{m(H)}}{2})$, расстояние от заданной точки $u_{0(sr)}$ к r-ой гиперплоскости $d_r = \frac{\left| a_r u^T_{0(sr)} - c_r \right|}{\left\| a_r \right\|}$.

Следствие 4. Достаточным условием существования единственного решения (1), (2) есть выполнение $\left\| u_0^{(i)} - u_{0(sr)} \right\| < \delta$, $\alpha_{lk}^{(i)} \ne 0,\ i=\overline{1,m}$, на итерациях замещения строк (2) строками (1)

Следствие 5. Необходимым и достаточным условием неразрешимости (1), (2) есть выполнение $c_r \notin \left[c^0_{r(H)}, c^0_{r(B)} \right]$ хотя бы для одного $r \in I$.

Следствие 6. Достаточным условием существования единственного решения (1), (2) есть выполнение $\left\| u_0^{(i)} - u_{0(sr)} \right\| < \delta$, $\alpha_{lk}^{(i)} \ne 0,\ i=\overline{1,m}$, на итерациях замещения строк (2) строками (1)

Следствие 7. Необходимым и достаточным условием неразрешимости (1), (2) есть выполнение $c_r \notin \left[c^0_{r(H)}, c^0_{r(B)} \right]$ хотя бы для одного $r \in I$.

Следствие 8. Необходимым и достаточным условием неразрешимости (1), (2) есть выполнения $R_r < d_r$ хотя бы для одного $r \in I$.

Следствие 9. Необходимым условием разрешимости (1), (2) есть выполнение $c^0_{r(H)} < c_r < c^0_{r(B)}\ \ \forall r \in I$.

Следствие 10. Необходимым условием разрешимости (1), (2) есть выполнения $R_r > d_r\ \ \forall r \in I$.

Справедливость теоремы 10 и следствий 4-10 вытекает из критериев пассивности и неразрешимости (теоремы 8, 9).

Выводы

Применение симплексной идеологии на основе МБМ даёт возможность:

- исследовать свойства решений СЛАР и СЛАН (1), (2) при изменениях в векторах ограничений;
- проводить анализ свойств системы при изменении значений отдельных элементов и ее компонент;
- использовать решение начальной системы при анализе возмущенной системы
- контролировать или направлено изменять величину ранга системы;

- разрабатывать механизм постадийного преобразования вырожденной матрицы ограничений в невырожденную направленной коррекцией элементов;
- находить решение квадратной системы неравенств за фиксированное количество шагов
- строить начальные решения задач на основе тривиальных базисных матриц, которые исключает трудоёмкие начальные вычисления;
- применять схему анализа для задач, которые предусматривают многошаговость или многократность расчетов на моделях при изменениях в компонентах модели;.
- анализировать корректность построения модели - на наличие пассивных и порождающих неразрешимость за несовместностью компонент на стадии анализа так и на стадии решения при уточнении функции принадлежности или интервалов принадлежности параметров.

Литература

[Леонтьев, 1972] Леонтьев В.В., Форд Д. Межотраслевой анализ воздействия структуры экономики на окружающую среду// Экономика и математические методы.- 1972.-Т.VIII,-Вып.3.-с.370-400

[Гасс, 1961] Гасс С. Линейное программирование. Физматгиз,-1961

[Волошин, 1987] Волошин А.Ф. Метод локализации области оптимума в задачах математического программирования // Докл. АН СССР. - 1987. -293, N 3.- С. 549-553.

[Орловский, 1981] Орловский С.А Принятие решения при нечёткой исходной информации.- М:, Наука,-1981,- 206с.

[Волошин,1993] Волошин А.Ф. Войналович В.М., Кудин В.И. Предоптимизационые и оптимизационные схемы сокращения размерности задачи линейного програмирования // Автоматика,N4, 1993.

[Войналович, 1987] Волкович В.Л., Войналович В.М., Кудин В.И. Релаксационная схема строчного симплекс метод // Автоматика. - 1987. -N4.-С. 79-86.

[Войналович, 1988] Волкович В.Л., Войналович В.М., Кудин В.И. Релаксационная схема двойственного строчного симплекс метода // Автоматика.-1988. -N 1,с.39-46.

[Кудин, 2002] Кудин В.И. Применение метода базисных матриц при исследовании свойств линейной системы // Вестник Киевского университета. Серия физ.-мат. науки. - 2002.-2., С. 56-61.

Информация об авторах

Владимир И. Кудин *– Киев, Киевский национальный университет имени Тараса Шевченко, факультет кибернетики, Украина, к.т. н., с. н.с., e-mail: V_I_Kudin@mail.ru*

Григорий И. Кудин *– Киев, Киевский национальный университет имени Тараса Шевченко, факультет кибернетики, Украина, к.ф. -г.н., доц., e-mail: kuding@mail.univ.kiev.ua*

ЭВОЛЮЦИОННЫЙ МЕТОД ОПРЕДЕЛЕНИЯ КРАТЧАЙШЕГО ПУТИ ПРОЕЗДА ПОЖАРНОГО РАСЧЕТА К МЕСТУ ПОЖАРА С ОПТИМИЗИРОВАННЫМ ПРОСТРАНСТВОМ ПОИСКА

Виталий Снитюк, Александр Джулай

Аннотация*: В статье предложен метод определения кратчайшего пути проезда пожарного автомобиля к месту пожара по критерию минимизации времени с использованием эволюционного моделирования. Исследован алгоритм его реализации на базе полного и оптимизированного пространства поиска возможных решений. Рассмотрены аспекты формирования моделей целевой функции и программной реализации метода. Выполнена экспериментальная верификация и приведены результаты сравнительного анализа с экспертными заключениями.*

Ключевые слова*: Неопределенность, эволюционное моделирование*

Введение

Поиск кратчайшего пути является задачей дискретной оптимизации. При этом определение оптимального пути проезда пожарного автомобиля к месту пожара имеет аспекты, которые выделяют его из общего ряда таких задач. Так, практически, это единственная задача, которая решается в критических условиях, от правильности ее решения зависят человеческие жизни. Верно выбранный маршрут - необходимое условие предотвращения техногенных и экологических катастроф. В условиях дефицита материальных и кадровых ресурсов минимизация времени проезда пожарного расчета является решающим фактором предотвращения негативных последствий пожара. Значительное количество научных исследований посвящено решению этой задачи.

Традиционно предлагают обоснования маршрута выезда пожарного автомобиля, исходя из критерия минимизации времени прибытия личного состава и пожарно-технического вооружения на место пожара. В статье [Пряничников, 1988] выполнен анализ факторов, влияющих на аварийную безопасность дорог: ширины проезжей части, обочины, количества полос движения, радиуса кривизны, видимости, интенсивности транспортных потоков. Предложено определять коэффициент дорожных условий по формуле:

$$D = \left[\sum_{i=1}^{n} \left(\prod_{j=1}^{m} k_{ij} \right) L_i \right] / L, \qquad (1)$$

где n – количество участков маршрута, m – число факторов, определяющих дорожные условия, k_{ij} – коэффициент важности j-го фактора дорожных условий на i-м участке маршрута, L_i – длина i-го участка, L – общая длина маршрута следования. Выезд пожарного расчета предполагается по маршруту, имеющему наибольшее значение D.

Во многих научных публикациях рассматриваются аналогичные подходы. Их недостатком является необходимость рассмотрения фиксированного, желательного полного набора возможных маршрутов, что практически трудно реализуемо. Не предусмотрена возможность варьирования значений важности факторов дорожных условий, что в условиях изменения дорожной обстановки, ремонта дорожного полотна, погодных условий приводит к искажению предполагаемого времени проезда. Необходима разработка адекватной модели времени проезда, как зависимости от значимых факторов, с возможностью ее уточнения и адаптации к изменяющимся внешним условиям.

Важно заметить, что разработка модели времени проезда является необходимым условием определения кратчайшего пути следования к месту пожара. Достаточным условием является метод, который конструктивно позволит определить оптимальный маршрут. Поскольку рассматриваемая задача имеет комбинаторный характер и, как следствие, неизбежной является проблема вычислительной сложности алгоритма, то необходимо предусмотреть реализацию технологии, которая позволит сократить количество анализируемых маршрутов и оптимизировать процесс вычислений.

Постановка задачи определения оптимального пути проезда пожарного автомобиля

Без ограничения общности будем считать, что структура дорог является прямоугольной (рис. 1). Пронумеруем каждый перекресток в соответствие с центрально-радиальной схемой. Местонахождение пожарного подразделения имеет нулевой номер, наиболее отдаленному “северо-восточному” перекрестку отвечает наибольший номер. Количество перекрестков – N. Рассмотренной структуре дорог отвечает матрица расстояний между перекрестками $S = (s_{ij})_{i,j=0}^{N-1}$, где s_{ij} – расстояние от i-го к j-у перекрестку. Зная среднюю скорость движения пожарного расчета, матрице расстояний можно поставить в соответствие матрицу времени проезда между перекрестками $T = (t_{ij})_{i,j=0}^{N-1}$.

Факторы, влияющие на время проезда, по форме представления их значений можно разделить на три группы: детерминированные, вероятностно-статистические и субъективные.

Минимальное количество перекрестков K на пути прохождения – детерминированный фактор, его возможные значения – натуральные числа, равные номеру квазиконцентрической окружности (см. рис. 1) и увеличивающиеся в меру отдаления перекрестка назначения от местоположения пожарного подраздела.

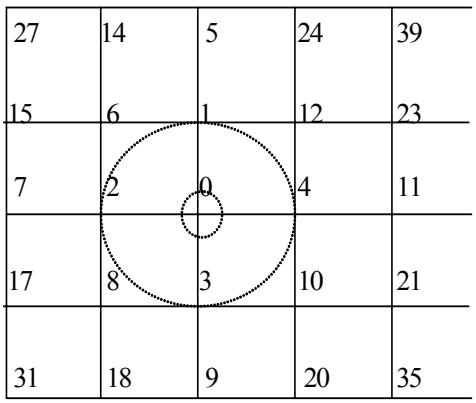


Рис. 1. Центрально-радиальная нумерация перекрестков

Загруженность дорог U – вероятностно-статистический фактор, который характеризуется статистическим рядом распределения (табл. 1), где в верхней части таблицы находятся временные интервалы, в нижней – относительные частоты количества автомобилей на дороге в этих временных интервалах. Качество дорожного покрытия V является субъективным фактором и определяется функцией принадлежности, которая может быть как непрерывной, так и дискретной. Ее построение осуществляется одним из двух способов, первый из которых базируется на парных сравнениях, выполненных одним экспертом [Ротштейн, 2002], второй – на статистической обработке мнений группы экспертов [Zadeh, 1965].

Таблица 1

Статистический ряд

Интервалы	$[t_0, t_1]$	$[t_1, t_2]$	...	$[t_{n-1}, t_n]$
Относительные частоты	f_1	f_2	...	f_n

Предположим, что место пожара H находится между двумя перекрестками n_1 и n_2. Тогда необходимо определить оптимальный маршрут, что отвечает решению задачи [Снитюк, 2004]:

$$\min_t \{L_{on_1} + L_{n_1 H}; L_{on_2} + L_{n_2 H}\} \quad (2)$$

где L_{ij} – маршрут от i-го пункта к j-у. Исходными данными для решения задачи (2) являются значения элементов матриц S; T; $K = (k_{ij})_{i=1,\ j=1}^{N\ \ 2}$, где k_{i1} – номер перекрестка назначения, k_{i2} – минимальное количество перекрестков, которое необходимо проехать при прохождении к k_{i1}; $G = (g_{ij})_{i=1,\ j=1}^{24\ \ 2}$, где g_{i1} – номер временного интервала (сутки разбиты на 24 промежутка: с 0 часов до 1-го часа (1), с 1-го до 2-го часа (2),...), g_{i2} – относительные частоты количества автомобилей в g_{i1}-м временном интервале, $\sum_{i=1}^{24} g_{i2} = 1$; $\sum_{i=1}^{24} g_{i2} = 1$; $Q = (q_{ij})_{i,j=1}^{N}$, где $q_{ij} \in (0,1)$ – коэффициенты, которые определяют качество дорожного покрытия на участке от i-го перекрестка к j-у. Заметим, что матрица G может иметь не статистическую, а субъективную природу. Если движение в одно и то же время на разных участках дороги является неравномерным, то матрица будет трехмерной, одно из измерений которой будет отвечать номеру участка дороги. В зависимости от особенностей конкретного города или ситуации, количество матриц значений факторов, влияющих на скорость движения пожарного расчета, может быть увеличено. Отметим, что содержательно сущность учета других факторов не будет отличаться от уже рассмотренных.

Предпосылки решения задачи определения оптимального пути с помощью эволюционного моделирования

Особенности социально-экономического развития стран являются непосредственной причиной роста количества пожаров и, как следствие, гибели людей и нанесения имущественных убытков. Кадровый и материальный дефицит есть, с одной стороны, причиной неэффективного тушения пожаров, а с другой – стимулом к внедрению новых информационно-аналитических технологий, что позволяет повысить эффективность работы пожарных подразделений.

Одной из задач, требующих применения интеллектуальных моделей и методов, является минимизация времени проезда пожарного расчета к месту пожара.

Определим исходные предпосылки ее решения. Заметим, что такая задача имеет некоторые общие аспекты с известной задачей коммивояжера. Известно, что точного метода решения данной задачи любой размерности, кроме полного перебора всех вариантов, не существует. Удовлетворительные результаты дают метод ветвей и границ [Luger, 2002; Зайченко, 2000], метод последовательного анализа вариантов [Волкович, 1993], поиск оптимального пути с использованием нейронной сети Хопфилда [Уоссермен, 1992]. Однако посредством последнего метода точный результат получают, примерно, в 50% вычислений, точность первых методов зависит от размерности задачи, высоковероятным также является попадание в локальные оптимумы.

Особенностью задачи поиска оптимального пути пожарного расчета заключаются в том, что наилучшее решение ищется по критерию минимума времени. При этом необходимо учитывать количество перекрестков по пути следования, загруженность дорог (среднее количество автомобилей на дороге в единицу времени), их качество. Учет других факторов также является возможным при их особой значимости и необходимости. Отметим, что технология определения оптимального пути проезда пожарного расчета к месту пожара реализуется с учетом субъективных и статистических факторов. Базовым ее элементом является эволюционный метод определения кратчайшего пути, который заключается в следующем.

Без ограничения общности представим (2) как задачу нахождения

$$\min_{t} L_{0n_1}. \qquad (3)$$

Очевидно, что для решения задачи (2) необходимо дважды решить (3) и выполнить некоторые уточнения результата. Поиск оптимального пути будем осуществлять посредством эволюционного алгоритма (ЕА) специального вида, который позволяет находить глобальные оптимумы, в общем случае, недифференциируемых функций. Определим его основные принципы и базовые элементы.

Основным понятием ЕА является *генеральная совокупность* – все множество возможных решений. В нашем случае определим генеральную совокупность как множество векторов $X = (x_0, x_1, x_2, ..., x_k, x_n)$, где x_0 – место дислокации пожарного подразделения, x_n – номер перекрестка, ближайшего к месту пожара. Таким образом, значениями элементов вектора X является последовательность номеров перекрестков, которые необходимо проехать для того, чтобы прибыть в x_n. Заметим, что количество перекрестков, в общем случае, является переменным. Минимальное значение k определяется номером квазиокружности (см. рис. 1), на котором лежит перекресток x_n, максимальное значение может быть достаточно большим. Все x_i, $i = \overline{0,k}$ являются разными и ни одно из них не совпадает с x_n. На первый взгляд, оптимальнее будут те варианты, у которых $x_i < x_j$ для всех $i < j$, но выполнение такого условия не является обязательным.

Модель целевой функции

Адекватное применение ЕА связано с превращениями числовых значений из двоичной системы исчисления в десятичную и наоборот. При этом возникает информационная избыточность, поскольку не все двоичные представления имеют свои аналоги в десятичной системе. В общем случае, это приводит к необходимости привлечения дополнительных вычислительных ресурсов и увеличения времени решения задачи [Кисляков 2000, 2001].

Вышеизложенные факты указывают на значительную трудоемкость и нецелесообразность формирования генеральной совокупности. О принадлежности к ней будут свидетельствовать результаты проверки. Важной процедурой является определение выборочной последовательности, которая должна иметь свойство репрезентативности [Goldberg, 1989; Werbos, 1974; Исаев, 2000; Jensen, 2001]. Векторы выборочной последовательности могут иметь разное количество элементов, что связано с количеством перекрестков на пути проезда. Их генерация происходит с учетом содержания матрицы S. Первый и последний элементы векторов одинаковы (перекресток, где находится пожарное депо и ближайший перекресток к месту пожара). Другие элементы определяются случайным образом, но с учетом

выполнения условия, что из места дислокации пожарного подраздела можно попасть на один из 4-х перекрестков, а из каждого из них – уже на один из трех. Обозначим P – количество элементов в выборочной совокупности.

Для формирования целевой функции (fitness-function) можно применить два подхода. В первом случае необходимо иметь достаточное множество статистических данных, сгруппированных в табл. 2, и осуществить идентификацию зависимости

Таблица 2

Структура исходных данных для идентификации fitness-function

Длина пути, L	Количество перекрестков, K	№ временного интервала, g	Качество дорожного покрытия, q	Время проезда, T

$$T = F(L, K, g, q), \tag{4}$$

где T – время следования пожарного расчета к месту пожара, K – количество перекрестков, которые он проехал, g – номер временного интервала, q – показатель качества дорожного покрытия, который интегрирует в себе и погодные условия. При правильной формализации задачи осуществить идентификацию (4) несложно. Достаточно предварительно выполнить нормализацию данных и применить метод наименьших квадратов для построения уравнения линейной регрессии [Наконечный, 1997], метод Брандона – для нелинейной регрессии [Чавкин, 2001], методы самоорганизации моделей – для полиномиальных зависимостей (метод группового учета аргументов [Ивахненко, 1975] или метод последовательных упрощений [Васильев, 2001]).

Во втором случае формирования целевой функции происходит эмпирически с использованием взвешивающих и поправочных коэффициентов. При этом используются данные матрицы T. Среднее время проезда из x_0 в x_n определяется по формуле (по одному из маршрутов):

$$T_{cp.} = \sum_{i=0}\sum_{j\neq i}^{n} t_{ij} \cdot \chi(s_{ij} \neq 0), \tag{5}$$

где $\chi(*)$ – функция-индикатор. Ввиду того, что, в среднем, время прохождения пожарного расчета увеличивается с увеличением количества перекрестков, уточним (5):

$$T = w_1 \cdot k_{n2} \cdot T_{cp}, \tag{6}$$

где w_1 – весовой коэффициент, который определяет значимость параметра количества перекрестков. С учетом качества дорожного покрытия целевая функция (5)–(6) является такой:

$$T = w_1 \cdot w_2 \cdot k_{n2} \cdot \sum_{i=0}\sum_{j\neq i}^{n} t_{ij} \cdot q_{ij} \cdot \chi(s_{ij} \neq 0), \tag{7}$$

где w_2 – весовой коэффициент, указывающий на важность параметра качества дорожного покрытия. Поскольку в разное время суток длительность прохождения пожарного расчета к месту пожара будет разной, то модель (7) необходимо уточнить:

$$T_v = \frac{\prod_{i=1}^{3} w_i}{g_{t2}} \cdot k_{n2} \cdot \chi(v = g_{t1}) \cdot \sum_{i=0}\sum_{j\neq i}^{n} t_{ij} \cdot q_{ij} \cdot \chi(s_{ij} \neq 0), \tag{8}$$

где w_3 – весовой коэффициент важности временных интервалов, v – номер временного интервала.

Сделаем ряд замечаний. Значение функции (8) необходимо рассчитывать в зависимости от времени пожара. Весовые коэффициенты определяются эмпирически экспертами. Таким образом, использование предложенного подхода субъективизировано. Построение функции (4) осуществляется аналитически и, в большинстве случаев, может быть теоретически обосновано. Зависимость (8) получают, исходя из эмпирических умозаключений, и процедура ее верификации является достаточно длительной. Второй подход к получению модели рационально использовать при малой ретроспективе априорных данных.

Эволюционный метод определения оптимального пути следования пожарного расчета

Учитывая то, что каждая вершина (перекресток) инцидентна только четырем другим вершинам, а их общее количество является достаточно большим (используется при построении матриц S и T), применять традиционное бинарное представление элементов вектора совокупности (хромосомы) в классическом ЭА нерационально. Пусть $X_1, X_2, ..., X_p$ – векторы выборочной совокупности (содержат множество перекрестков–маршрутов), упорядоченные по количеству элементов, т.е. $|X_i| \le |X_j|, i < j$. Для каждого из них, рассчитав значение функции (4), получим $T_1, T_2, ..., T_p$.

Используя принцип последовательного преодоления неопределенности, *кроссовер* будем проводить по принципу последовательного отбора [Витковски, 2003; Алгулиев, 2004], в соответствии с которым большую вероятность участия в рекомбинациях имеют векторы с меньшим значением fitness-function. Предположим, что необходимо определить оптимальный маршрут к перекрестку № 39 (см. рис. 1). Для кроссовера выбраны векторы (0, 1, 5, 24, 12, 23, 39) и (0, 1, 12, 4, 11, 23, 39). Определяем, имеются ли одинаковые элементы в этих векторах, кроме первых двух и последнего элемента. Такой элемент – 12, он и является точкой рекомбинации. Осуществив кроссовер, получим два вектора-потомка: (0, 1, 12, 23, 39) и (0, 1, 5, 24, 12, 4, 11, 23, 39). Если одинаковых элементов нет, то один из векторов (с минимальным значением fitness-function) оставляем и случайным образом (с использованием принципа пропорциональности) выбираем другой вектор из выборочной совокупности. Результатом кроссовера будет ноль, один или два вектора. Ноль, если $\exists x_i, x_j : x_i = x_j, i \ne j$ в каждом из векторов; один – если в одном; два, если указанные условия не выполнены ни для одного из векторов-потомков.

Получив P потомков, среди них и среди P родителей выбираем P наилучших векторов. Такой отбор называется элитным. Кроме него, существуют и другие методы отбора: селективный, панмиксия, отбор с вытеснением [Исаев, 2000]. Практическое моделирование засвидетельствовало преимущество именно элитного отбора, поскольку при нем не теряются оптимальные векторы-решения. Из всех видов отбора только для элитного теоретически доказано [Harti, 1990], что итерационный процесс поиска оптимального решения сходится.

Для предотвращения попадания fitness-function в локальный оптимум предусмотрена процедура мутации. Происходит она с вероятностью 0,01 по такой схеме. Разыгрываем случайное равномерно распределенное на множестве $\{1, 2, ..., P\}$ число. Если $\xi = k$, то мутации подлежит k-й вектор выборочной совокупности. Если количество элементов в нем равняется d, то разыгрывается случайное число η на множестве $\{2, 3, ..., d-1\}$. Мутации осуществляются у $\eta = L$ элементов, для чего осуществляется случайный выбор из двух вариантов $(L+1)$-го элемента. Критерием окончания процесса поиска оптимального решения является выполнение одного из следующих условий:

- достижение необходимого значения fitness-function;
- выборочная популяция состоит из одинаковых элементов;
- для любого значения $\varepsilon > 0 : |T_i - T_j| < \varepsilon, \forall i, j, i \ne j$.

Если выполняются первое или третье условие, то решением задачи будет вектор, значение fitness-function которого является наименьшим.

Такой метод имеет свои преимущества перед классическим ЭА и недостатки, связанные с особенностями задачи. Преимуществом является значительное сокращение количества операций, что объясняется неприменением процедуры преобразования чисел в ЭА из десятичной системы счисления в двоичную и наоборот. Десятичное представление оптимизирует процедуру кроссовера за счет уменьшения времени формирования векторов-потомков. В пользу предложенного метода свидетельствует также то, что он не “привязан” к прямоугольной структуре улиц. Если на некоторых из них выполняется ремонт, то в матрицах

S и T достаточно на соответствующих местах поставить нули. К недостаткам отнесем проблему формирования выборочной совокупности, что связано с разным количеством элементов у векторов-представителей. Кроме того, процедура определения каждого следующего элемента вектора требует пересмотра строки матрицы расстояний или времени, что при большом количестве перекрестков значительно увеличивает время работы алгоритма.

Предложенная технология ориентирована на то, что соответствующий программный модуль будет работать как в активном, так и в пассивном режимах. В пассивном режиме для каждого временного интервала по известным матрицам количества перекрестков на пути следования пожарного подразделения и качества дорожного покрытия рассчитывается оптимальный маршрут и записывается в базу данных. При пожаре расчету будет выдаваться распоряжение с двумя вариантами маршрутов к смежным перекресткам. При изменении параметров в одной из определяющих матриц или возникновении ситуации, при которой возникает потребность в экстренной выдаче информации о маршруте, которого нет в базе данных, система переводится в активный режим работы и экстренно решает задачу.

В общем случае, решение (8) является локальным оптимумом, поскольку процесс его поиска определяется выбором начальной точки и величины шага поиска. Поэтому возникает необходимость использования эволюционных методов, которые являются инвариантными к такому выбору.

Технология оптимизации пространства поиска решения задачи

В процессе моделирования выявлено две проблемы. Первая из них заключалась в том, что из каждого перекрестка, обычно, имеются маршруты к четырем других. В то же время, в матрице расстояний, как минимум, на порядок больше вариантов, поэтому возникает значительная вычислительная избыточность. Другая проблема заключается в рациональном представлении хромосом-решений. В частности, априорно невозможно определить, какую длину должна иметь хромосома, количество элементов которой отвечает количеству перекрестков, которые должен проехать пожарный расчет при следовании к месту пожара.

Для решения указанных проблем предлагается такая процедура. В соответствие с рис. 1 и матрицей расстояний строим матрицу $N = (n_{ij})_{i,j=1}^{4\ m}$ (таблицу направлений, в которой показаны перекрестки, смежные фиксированному) и матрицу $L = (l_{ij})_{i,j=1}^{4\ m}$ (таблица расстояний от фиксированного перекрестка к смежным) (табл. 3).

Таблица 3

Таблица направлений

Перекресток	0	1	2	3	4	5	6	7	8	9	10	11	12
Налево	2	6	7	8	0	14	15	*	17	18	3	4	1
Прямо	1	5	6	0	12	*	14	15	2	3	4	23	24
Направо	4	12	0	10	11	24	1	6	3	20	21	*	23
Назад	3	0	8	9	10	1	2	17	18	*	20	21	4
Таблица расстояний между перекрестками													
Перекресток	0	1	2	3	4	5	6	7	8	9	10	11	12
Налево	1	3	3	5	6	3	2	*	3	2	4	4	2
Прямо	9	1	1	3	7	*	2	3	1	2	2	3	3
Направо	6	2	1	4	4	2	3	3	5	1	2	*	3
Назад	3	9	1	2	2	1	1	2	1	*	2	3	7

Очевидно, что до фиксированного перекрестка существует большое количество путей, каждый из которых проходит через разное количество промежуточных перекрестков. Минимальное количество таких

перекрестков определяется номером квазиконцентрической окружности, проходящей через финальный перекресток. Максимальное количество перекрестков определяется экспертным путем и, чаще всего, не превышает тройного количества минимальных перекрестков в критических случаях, и двойного – в штатных ситуациях.

Определим в качестве конечного перекресток № 39 (см. рис. 1). Он принадлежит четвертой окружности, поэтому наименьшая длина хромосомы равняется четырем и она будет такой:

$x(1)$	$x(2)$	$x(3)$	$x(4)$

В хромосоме $x(1)=0$ стартовая точка (депо), $x(4)=39-$ конечная точка. Максимальную длину хромосомы положим равной восьми. Параллельно с выполнением традиционных операций ЭА, в предложенной процедуре необходимо придерживаться таких шагов. При инициализации выборочной популяции обеспечить равномерное представительство хромосом разной длины. Для этого разыгрываем случайное равномерно распределенное целое число из множества {4, 5, 6, 7, 8}, которое отвечает длине хромосомы. Если это число 4, то первый и последний ее фрагмент уже известны. Вспомогательная хромосома состоит из четырех генов. Первые два гена кодируют направление движения из $x(1)$ (соответственно: 00 – налево, 01 – прямо, 10 – направо, 11 – назад), другие два – из $x(4)$. На допустимость решения указывает выполнение ограничения, которое определяет то, что перекрестки $x(2)$ и $x(3)$ являются соседними. Для хромосом с большей длиной такая процедура выполняется рекурсивно.

Анализ результатов моделирования

Время экспериментального моделирования без выполнения процедуры сужения пространства поиска на компьютере Pentium 2,0 GHz составило, в среднем, 12-16 минут. Если же в алгоритме поиска выполняется вспомогательная процедура, то время поиска оптимального решения за счет сокращения неверных шагов уменьшилось до 0,8-1,1 минут. Если целевой функцией является зависимость (8) с предварительно установленными экспертным путем весовыми коэффициентами, то время проезда к месту пожара по маршруту, определенному посредством моделирования, на 7-10% является меньшим, чем время, которое отвечает маршруту, предложенному экспертами (начальниками боевых расчетов) или совпадает. Верификация этого факта достигается вычислением целевой функции по двум предложенным маршрутам при постоянных значениях весовых коэффициентов, определяющих особенность проезда.

Заключение

Метод определения кратчайшего пути следования пожарного расчета к месту пожара с оптимизацией пространства поиска является технологией, позволяющей избежать человеческих жертв и сократить материальный ущерб. Его эффективное применение предполагает наличие информационной базы, содержащей данные о количестве перекрестков, состоянии дорог и дорожной обстановке, а также ее обновление в режиме реального времени. Увеличивающееся количество “пробок” на дорогах подчеркивает актуальность предложенного метода. Изменение информации предполагает пересчет оптимального маршрута.

Вычислительная сложность эволюционных алгоритмов обосновывает необходимость разработки методов, направленных на увеличение скорости расчетов при неизменной точности. Потому перспективным представляется разработка оптимизированных моделей целевых функций, процедур уменьшения информационной избыточности начальных данных. Важно заметить, что предложенные модели обладают свойством открытости, т.е. допускают учет и других значимых факторов, а весовые коэффициенты целесообразно разделить на локальные (характеризующие участки дорог) и глобальные, являющиеся атрибутами дорожной ситуации в целом.

Библиография

[Пряничников, 1988] В.А. Пряничников, В.В. Роенко. Критерий выбора маршрутов следования пожарных автомобилей // Организация работ по профилактике и тушению пожаров: Сб. научн. тр. – Москва: ВНИИПО, 1988. – С. 89-92.

[Ротштейн, 2002] А.П. Ротштейн. Влияние методов дефаззификации на скорость настройки нечеткой модели // Кибернетика и системный анализ. – 2002. – № 1. – с. 34-45.

[Zadeh, 1965] L. Zadeh. Fuzzy sets // Information and control. – 1965. – № 8. – P. 338-353.

[Снитюк, 2004] В.Е. Снитюк, А.Н. Джулай. Интеллектуальная технология оптимизации пути следования пожарного расчета к месту пожара // АСУ и приборы автоматики. – 2004. – Вып. 129. – С. 41-46.

[Luger, 2002] G.F. Luger. Artificial intelligence. Structures and strategies for complex problem solving. – Addison Wesley: Boston, 2002. – 864 p.

[Зайченко, 2000] Ю.П. Зайченко. Исследование операций. – Киев: Випол, 2000. – 688 с.

[Волкович, 1993] В.Л. Волкович, А.Ф.Волошин, В.А. Заславский, И.А. Ушаков. Модели и методы оптимизации надежности сложных систем. – Киев: Наукова думка, 1993. – 312 с.

[Уоссермен, 1992] Ф. Уоссермен. – Нейрокомпьютерная техника: теория и практика. – Москва: Юнити, 1992. – 240 с.

[Кисляков, 2000] А.В. Кисляков. Генетические алгоритмы: математический анализ некоторых схем репродукции // Информационные технологии. – 2000. – № 12. – С. 9-14.

[Кисляков, 2001] А.В. Кисляков. Генетические алгоритмы: операторы скрещивания и мутации репродукции // Информационные технологии. – 2001. – № 1. – С. 29-34.

[Goldberg, 1989] D.E. Goldberg. Genetic algorithms in search, optimization and machine learning. – Addison wesley, 1989.– 196 p.

[Werbos, 1974] P. Werbos. Beyond regression: new tools for prediction and analysis in the behavioral sciences. – PhD thesis: Harvard university, 1974. – 240 p.

[Исаев, 2000] С.А. Исаев. Разработка и исследование генетических алгоритмов для принятия решений на основе многокритериальных нелинейных моделей: Автореф. дисс. канд. техн. наук: 05.13.17 / Нижегородск. гос. унив. – Нижний Новгород. – 2000. – 18 с.

[Jensen, 2001] Mikkel. T. Jensen. Robust and flexible scheduling with evolutionary computation // phd thesis. – University of Aarhus, Denmark. – 2001. – 299 pp.

[Наконечный, 1997] С.И. Наконечный, Т.О. Терещенко, Т.П. Романюк. Эконометрия. – Киев: КНЭУ, 1997. – 352 с.

[Чавкин, 2001] А.М. Чавкин. Методы и модели рационального управления в рыночной экономике. – Москва: Финансы и статистика, 2001. – 320 с.

[Ивахненко, 1975] А.Г. Ивахненко. Долгосрочное прогнозирование и управление сложными системами. – Киев: Техника, 1975. – 312 с.

[Васильев, 2001] В.И. Васильев. Взаимозаменяемость метода группового учета аргументов (МГУА) и метода предельных упрощений (МПУ) // Искусственный интеллект. – 2001. – № 1. – С. 29–42.

[Витковски, 2003] Т. Витковски, С. Эльзвай, А. Антчак. Проектирование основных операций генетических алгоритмов для планирования производства // Проблемы управления и информатики. – 2003. – № 6. – С. 129-138.

[Алгулиев, 2004] Р.М. Алгулиев, Р.М. Алыгулиев. Генетический подход к оптимальному назначению заданий в распределенной системе // Искусственный интеллект. – 2004. – № 4. – С. 79-88.

[Harti, 1990] R.E. Harti. A global convergence proof for class of genetic algorithms. – Wien: Technische Universitaet, 1990. – 136 p.

Информация об авторах

Виталий Снитюк – *Киевский национальный университет имени Тараса Шевченко, докторант факультета кибернетики; пр. Акад. Глушкова 2, стр. 6, Киев, Украина; e-mail: svit@majar.com*

Александр Джулай *– Черкасский институт пожарной безопасности имени Героев Чернобыля, старший преподаватель; ул. Оноприенко, 8, Черкассы, Украина; e-mail: djulaj@ukr.net*

К ОПРЕДЕЛЕНИЮ КОНКУРИРУЮЩИХ РИСКОВ И ИХ ВЕРОЯТНОСТНОМУ МОДЕЛИРОВАНИЮ

Май Корнийчук, Инна Совтус

***Аннотация**. Предлагается оригинальный взгляд на изучение теории риска, где рассматривается взаимодействие нескольких рисков, которые действуют на один, и один и тот же объект. Особенность этого взаимодействия определяется в конкуренции рисков. Можно спорить относительно удачности этого названия, но подход к его оцениванию удается сделать содержательным. Окончание работы не дает формализованных результатов, что требует дальнейших шагов изучения конкурирующих рисков.*

***Ключевые слова:** риск, конкурирующие риски, вероятностная модель.*

Постановка проблемы

В данном исследовании мы преследуем двойную цель: дать примитивное в смысле утилитарного определения риска на основе теоретико-вероятностного обоснования и определиться с соображениями и подходами применения элементов теории риска с его специфической трудностью в трактовании отдельных параметров в моделях экономико-математического анализа вообще и в частности в моделях вероятностного экономико-математического (технологического) прогнозирования.

Состояние проблемы в современных исследованиях

Укажем, что, на наш взгляд, примитивизм в дефинициях риска не предусматривает вероятностного его упрощенного понятия или вульгаризации структуры. Под рассматриваемым углом зрения это такой же инструмент с разной (меньшей) мощностью и эффективностью, как, для примера – примитивного сравнения, соха и современный плуг для пахоты земли. В нашем исследовании соотносятся определения риска примитивное и возможное определения на базе современного содержания теоретико-вероятностной теории, хотя такого именно содержательного определения в известной нам литературе довольно найти. Можно прийти в изумление, ведь столько литературы посвящено этой теме, экономической, технической, и всюду есть определения риска, и на любой вкус, более того – это определение зафиксировано нормативным изданием: риск – это математическое ожидание убытков [1]. Определяется технический риск, экологический риск, экономический риск, а также его вариации и дифференциации: учетный риск, риск маркетинга, финансовый риск и большое количество других рисков, которые не дискутируются, а уже навязываются в массовых учебниках и пособиях, в особенности для экономических специальностей. Стало даже модным в любых исследованиях да и популярных публикациях говорить о риске в положительном или бранном смысле, поскольку и исследователи, и прикладники нередко смешивают понятия риска и понятия его следствий.

Мы говорим об аспектах риска, но не о самом риске, который за аспектами и характеристиками (что часто безосновательно к ним причисленные) еще не определяет четко, что это такое – риск. Хотя за логикой построения науки это определение должно предшествовать его аспектам [2]. Ведь не все аспекты и характеристики, которые ему приписывают, есть ему присущими, что вытекают с его сущностной структуры, ибо зная объект, далее легко познать и определить его модификации и их определяющие характеристики.

Для упрощения и наглядности восприятия изложенного приведем аналогию: владея понятием стол, довольно содержательно воспринимаем его модификации: обеденный стол, письменный стол, слесарный, стол дантиста, хирургический. Несмотря на чрезвычайно колебательную содержательную и прикладную вариацию этих понятий-слов, мы владеем четко очерченными понятиями, так как они базируются и нанизанные на его первичном базовом определении и понятии стола. Не имея такого первичного базового определения и понятия риска как такого, довольно и говорить о какой-либо четкой и содержательной цепи следственных определений частичных рисков, тем более, что при наличии такого

базового определения львиная часть отдельных определений отпадут, так как не являются таковыми и не могут попасть в эту следственную цепь.

Ход наших мыслей может вызвать удивление и даже негодование некоторых специалистов, авторов развесистых статей и даже книг с рискологии, которые предлагают десятки разнообразных определений риска на любой вкус, даже некоторую формализацию этих определений, которая, как правило, сводится к формализации не его, а лишь отдельной характеристики риска.

Проблема именно и состоит в том, чтобы сформировать четко очерченное понятие риска, формализованное в определенном пространстве непротиворечивой системы понятий, определений и утверждений, и дать его базовое, логически законченное формальное определение, применимые аспекты которого согласовывались бы с теми частыми аспектами, которые непротиворечиво описывают суть вещей. В аспекте расширения разноплановости исследований этой проблемы отдельный исследовательский интерес имеет так называемая конкуренция рисков [3, 4]. На первый взгляд имеем странный подход. Ведь о какой конкуренции может идти речь, когда не договорились с определением риска. Но с целью прозрачности и облегченности восприятия мы также сначала пойдем путем от прикладного аспекта риска, от частного к общему. В повседневной жизни мы часто рискуем... Когда мы рискуем? Тогда, когда ждем после осуществления определенного комплекса условий (надеемся) желательного наступления определенного события (надежность), и одновременно не ждем (хотя осознаем возможность) нежелательного наступления. То есть риск связан с желательным (нежелательным) наступлением в будущем (завтра, через час, время) определенного события, когда раньше (до сего момента, может сегодня) накопился комплекс определенных условий для такого наступления/ненаступления. То есть риск связан с возможным следствием, когда реализуется определенный комплекс условий, причем, нежелательному возможному наступлению следствия обязательно предшествует этот комплекс условий. Мы говорим о возможном наступлении, а это значит, что оно может и не наступить. То есть здесь мы действуем в пространстве случайности, так как реализация комплекса условий обязательно ведет к альтернативам нежелательного наступления следствия (желательного ненаступления) или наоборот. Имеем не что иное, как классическую схему теории вероятностей, где в рамках этой теории комплекс условий – это стохастичный эксперимент, предшествующий наступлению события со своим следствием. Поскольку событие может и не наступить, то оно есть случайно, причем альтернативность следствий есть не что иное, как несовместимость событий, а обязательность альтернативы наступления-ненаступления событий после реализации комплекса условий обуславливает их полную группу. Таким образом имеем пространство элементарных (альтернативных) событий: нежелательного наступления следствия (риск, рискуем в житейском смысле на нежелательное), и ненаступления его (надежность, надеемся на желательное). Следовательно, в аспекте наших выяснений и исследований в центре внимания риска четко обрисовывается неопределенность и его величество – случай. И лишь сквозь него, через неопределенность лежит путь познания сущности истины риска, его толкования и определения как и альтернативы этой истины – надежности. А это уже сфера действия теории вероятностей вообще, экономико-математического теоретико-вероятностного моделирования в частности.

Постановка задачи

Итак, перед нами задача применения вероятностных понятий и утверждений в экономико-математическом моделировании и преодолении специфической трудности, которые вытекают из привязки абсолютно абстрактных объектов теории вероятностей и реально определенных объектов экономического анализа. Огульно эти трудности можно идентифицировать двумя категориями.

Сущность первой состоит в том, что никакая реальная задача практически непосредственно не связана с таким математическим понятием, фундаментальным для теории вероятностей, как основное вероятностное множество. Итак, прежде чем делать попытку решения практической задачи с помощью экономико-вероятностного моделирования, надо сформулировать ее у вероятностных терминах. На самом деле перевод реальных экономических объектов у вероятностные термины фактически сводится к построению экономико-математической модели этой задачи. Вообще говоря, есть много способов построения математической модели, так что возникает вопрос, который из них избрать, учитывая при этом уровень адекватности. Есть случаи, когда адекватность конкретной экономико-вероятностной модели может быть проверена эмпирическим путем. Но в других случаях проверка такого рода может

оказаться весьма трудной, и мы должны полагаться на наше интуитивное ощущение, которое подсказывает нам, что причинно-следственный механизм связных факторов, постулированный в модели, удовлетворительным образом отвечает явлению, которое изучается. В этих условиях естественно, что решение задачи, основанное на данной модели, применимо к самой модели (точнее, к постулированных ею данных), и совсем не обязательно к явлению, для которого она была предназначена. Мера расхождения соответствия между математическим решением и характером явления зависит от адекватности модели.

Сущность второй категории трудности во время рассмотрения задач практики состоит в том, что большинство наблюдаемых явлений есть весьма сложные. Поэтому, когда мы подобающим образом анализируем их и строим содержательную экономико-математическую модель, которая выдается нам адекватной, часто случается, что и эта построенная модель самая по себе структурно настолько сложная, что возникают чисто математические трудности при получении необходимого решения. Во многих таких и подобных случаях мы терпим неудачу и вынуждены пересматривать структуру модели, жертвуя ее содержательностью и адекватностью для достижения определенной аналитической (математической) простоты с целью принадлежности ее к классу математически разрешимых.

Среди задач экономической практики, которые иллюстрируют указанные выше обстоятельства, являются задача вычисления так называемых конкурирующих рисков. Подобная задача кроме экономики может быть сформулирована во многих других областях, к примеру биологии, медицины, физических исследований, где она имеет целую цепь разных модификаций. Мы же попробуем исследовать ее на простом, элементарном, но актуальном примере из экономики. Однако соображения в этом примере покажут, как можно совершать и переносить их на более общие задачи. Здесь мы отойдем от задекларированного фундаментального определения риска, а подойдем традиционно как у многих публикациях из этой тематики [1–6], исследуя вместо риска его отдельные характеристики.

Решения проблемы

Итак, пример. Рассмотрим формализовано эффективность реформ в сфере производственной деятельности маленьких и средних предприятий, процесс функционирования которых во время рецессии отягощенный многими экономическими болезнями, такими, как отсутствие оборотных средств, отсутствие посильных и адекватных кредитов, недостаточность финансирования, недостаточность сырья и энергоресурсов, проблемы стоимости и сбыта продукции. Под содержанием реформ будем понимать формализованную реализацию определенного алгоритма на процесс функционирования отдельных предприятий. Комплекс определенных алгоритмических мероприятий реформы можно трактовать (удобно рассматривать) как способ лечения фирмы от этих болезней.

При исследовании эффективности действия некоторой реформы A^* как способа лечения хронических заболеваний фирмы, которые рассмотренные выше, важно учесть следующее: а) как часто при условии действия этой реформы выздоровления фирмы обрывается на протяжении некоторого промежутка времени T рецидивом этого же самого заболевания; б) как часто этот рецидив заканчивается банкротством фирмы (смертью) за время T от начала исследования.

На первый взгляд эмпирическое решение этих двух задач может показаться совсем простым. Для примера, чтобы ответить на первый вопрос, можно вообразить себе выделенную большую однородную группу фирм, скажем, тысячу предприятий (N = 1000), которые испытывали положительное влияние реформы A^* (некоторого лечения), и, кажется, что выздоровели, или почти выздоровели. Далее нужно подсчитать количество n_1 тех из них, которые имели рецидив болезни за это же время T (после предыдущего выздоровления). Тогда для характеризации эффективности реформы A^* (способа лечения) можно было бы использовать относительную частоту рецидива $q_1 = n_1/N$. Если рассматриваются два альтернативных способа реформирования предприятий A_1^* и A_2^* (лечения A_1^* и A_2^*), тогда лучшей можно было бы считать тот способ реформ, для которого частота q есть меньшей. Но, к сожалению, существуют серьезная практическая трудность при применении нами описанного пути. Первая трудность состоит в том, что невозможно четко выделить реальную группу предприятий, которые выздоровели от болезни и возвратились к нормальному функционированию. Уже через полгода, год–два, или около этого много предприятий растворяются между другими так, что тяжело “собрать” их для того, чтобы убедительно убедиться, живые они, или нет, нормально функционируют, испытали ли рецидив. Вторая

трудность (которая на самом деле есть скрытой вариацией того же типа, что и первая) в особенности очевидная тогда, когда исследуется проблема сравнения двух способов реформирования A_1 и A_2 предприятий, который применяется до двух разных групп N_1 и N_2, и которые функционируют в разных условиях. Например, вообразим себе, что группа N_1 находится в прекрасных условиях, так что на протяжении года после их нормального функционирования, то есть мысленного выздоровления от начала кризисного заболевания только несколько предприятий этой группы исчезли, причем от причин, не связанных с кризисной болезнью, которая исследуется, или эти предприятия объединились в корпорацию, или переместились в другой район (исчезли из поля зрения, как варианты исследуемой выборки). Предположим дальше, что группа N_2 состоит преимущественно из предприятий, которые испытывают повседневный риск всякого рода бед и влияния разных естественных (например, погодных) условий, связанного с условиями их функционирования (тенизация, региональная коррупция, рэкет). Не исключено, что до конца определенного периода наблюдения T группа N_1 может не утратить вообще ни одного члена, и, таким образом, на протяжении второго периода все N_1 из них будут подвергнутые риска рецидива болезни, рецессии, которая исследуется. С другой стороны, на протяжении первого периода наблюдения от причин, не связанных с той болезнью рецессии, может разориться и исчезнуть существенное количество членов из группы N_2, которые, таким образом, будут ограждены от рецидива этой болезни во втором и в последующих периодах наблюдений. Таким образом, устанавливаем, что *риск* разорения от других причин *“конкурирует” с риском рецидива*. В результате, если эффективность способов лечения A_1 и A_2 имеет в точности объективно одинаковые показатели, но группа предприятий N_1 испытывает незначительный риск разорения и уничтожения от других причин, тогда как группа N_2 испытывает сильный риск, то относительная частота q_1 рецидива, вычисленная для группы N_1, очевидно будет большей, чем в группе N_2. Если принимать во внимание сравнения способов эффективности лечения, то объективно отсюда вытекает, что частота q_1 не характеризует риск лишь одного чистого рецидива. Она зависит также от конкурирующего риска исчезновения (смерти) предприятий. Нетрудно убедиться, что частота q_1 зависит также от интенсивности “риска” убывающих предприятий, которые “утерянные” за время T, когда в конце этого периода невозможно установить их место нахождения (реформирования, слияния, изменение места пребывания) и получить четкую информацию о их судьбе.

Как следствие рассмотренных причин дезинформационного содержания частного типа q_1 не могут быть благоприятными оценками характеристики соответствующих рисков. По аналогии с [2] мы будем считать (называть) их приближенными нормами этих рисков. Теперь проследим, каким образом можно построить число, которое было бы приемлемое за характеристику интенсивности риска [5]. Наиболее естественным и логическим, что прежде всего напрашивается в канву рассуждений, может быть понятие относительной частоты данного события (риска) в длинной цепи наблюдений, когда этот риск наблюдался в искусственных условиях, где были исключены все другие риски. Тогда бы эту относительную частоту в длинном ряде испытаний можно было бы назвать чистой нормой риска за время T и трактовать (соответственно обозначать) через P (оценка вероятности соответствующего события). Итак, из определения вытекает, что чистая норма заданного риска с необходимостью есть чисто формальной (абстракцией), так как реально она едва ли может наблюдаться, поскольку невозможно исключить риск (вероятность) исчезновения предприятий. Правда, при сокращении времени наблюдений мы можем уменьшить риск исчезновения. Однако при этом риск рецидива также будет уменьшаться, и при внимательном исследовании оказывается, что задачи для случая, когда отрезок времени T большой, или когда время T – маленький, по сути теоретически одинаковые, хотя практическая реализация не всегда с этим согласовывается.

Заключение

Введением понятий приближенной и чистой норм рассматриваемого риска мы сделали шаг в направлении построения математической модели явления конкурирующих рисков. Однако нужно признать, что этот шаг есть недостаточно четким, его можно считать лишь первым шагом, и сам по себе он есть лишь прелюдией к формулированию задачи вероятностного моделирования конкурирующих рисков и построения лишь этапа модели ее решения. Дальнейшие шаги в этом направлении могут быть содержанием дальнейших исследований. Здесь мы лишь сосредоточили внимание на проблеме реального существования и возможной формализации модели конкурирующих рисков.

Библиография

1. Вероятность и математическая статистика. – М.: Изд-во «Российская энциклопедия», 1999. – 360 с.
2. *Корнійчук М.Т.*, *Совтус І.К.* Ризик і надійність. Економіко-стохастичні методи й алгоритми побудови та оптимізації систем. Монографія. – К.: КНЕУ, 2000.–212 с.
3. *Нейман Ю*. Вводный курс теории вероятностей и математической статистики. – М.: Наука, 1968. – 448 с.
4. *Хенлі Е.Дж., Кумамото Х.* Надійність проектування технічних систем і оцінка ризику. Пер. з англ. – К.: Вища школа, 1987. – 406 с.
5. *Корнійчук М.Т., Совтус І.К*. Квазіекспоненціальна модель функції вартості ризику як базовий критерій оптимізації системи // Моделювання та інформаційні системи в економіці. – К.: КНЕУ, 2000. – Вип.63. – С. 60–76.
6. *Корнійчук М.Т.*, *Совтус І.К.* Складні системи з випадковою зв'язністю: ймовірнісне моделювання та оптимізація. Монографія. – К.: КНЕУ, 2003.– 374 с.
7. *Цай Т.Н., Грабовый П.В., Марашада Б.С*. Конкуренция и управление рисками на предприятиях в условиях рынка. – М.: Аланс, 1997. – 220 с.
8. *Черкасов В.В*. Проблемы риска в управленческой деятельности. Монография. – М.: Рефа-бук; К.: Ваклер, 1999. – 288 с.

Информация об авторах

Май Корнийчук – *д.т.н., проф. Киевский национальный экономический университет, пр. Победы, 54/1, Киев-03680, Украина, e-mail: XSWEDCTRAVKA_@bigmir.net.*

Инна Совтус *– доктор технических наук, профессор.*

ИННА СОВТУС

После тяжкой непродолжительной болезни 9 октября 2005 года ушла из жизни Инна Кузьминична Совтус (девичья фамилия Жук), талантливый ученый и педагог, доктор технических наук, профессор, действительный член (академик) Аэрокосмической академии Украины.

Детство ее было трудным – безотцовщина с большими материальными лишениями. Родилась она 4 сентября 1955 года в семье служащих в провинциальном историческом городишке Остёр на Черниговщине. Мать ее (из остарбайтеров, что в СССР приравнивалось к предательству) – средний медработник войсковой части, отец – свободный музыкант – семью оставил. Еще в раннем детстве Инна познала лишения, несправедливости от взрослых, безнаказанность обидчиков. Поэтому рано пришло осознание необходимости к самоутверждению. Единственным средством к этому была отличная учеба в школе по всем предметам, успехи по физкультуре, а затем (после 1966 года, когда с появлением отчима перешла в киевскую школу) и в спорте, где она выступала в соревнованиях за класс, школу, район на соревнованиях по г. Киеву. Одновременно училась в музыкальной школе по классу баяна, имела абсолютный слух и чистый, хотя и несильный голос. Успехи в музыке были настолько значительны, что после окончания школы в 1972 году стояла проблема выбора продолжения учебы в консерватории, ибо победы и призовые места на общегородских школьных и молодежных фестивалях и конкурсах обеспечивали достаточные условия такого выбора. Но победил выбор в пользу точных наук, по которым успехи занятий и олимпиад были отмечены отличиями и грамотами; а также прагматизм учителей, советовавших развивать способности к естественным наукам и особенно к математике.

И она выбрала математику. Поступила в Киевский государственный педагогический институт на физмат факультет, где успешно училась и блестяще закончила в 1976 году, получив диплом с отличием. Но, к сожалению, в институте не обратили внимания на ее исследовательские способности, на ее незаурядное аналитическое мышление. И поэтому она после окончания института 10 лет работала в школе учителем математики. И лишь в 1985 году случайно попала на кафедру высшей математики военного института

(КВИРТУ), где и были замечены ее потенциальные способности к исследовательской работе. Это было время перед распадом Союза, и экономика страны, ранее удовлетворявшая любые милитаристские требования военных ястребов, теперь задыхалась от неспособности обеспечения вооруженных сил качественными надежными системами; если ранее качество достигалось любой ценой, то теперь стояла проблема его достижения при минимальной, или хотя бы рационально-ограниченной стоимости. Исследованиями по этой проблеме вплотную занялась И.Совтус. Уже через два года получила свои оригинальные результаты в области разработки и построения методов и моделей проектов конструирования сложных систем с заданными тактико-техническими характеристиками (ТТХ) и обеспечения этих ТТХ оптимальным набором требований к ТТХ элементов этих систем. В подобном ракурсе проблема исследовалась и ранее, но все исследования сводились к использованию методов линейного программирования, т.е. линеаризации целевой функции и ограничений. Подобные модели обладали весьма слабой адекватностью и давали весьма приближенное решение, да и то в условиях равно одинакового вклада качества (надежности) каждого элемента системы в качество самой системы. Совтус И.К. удалось преодолеть эту проблему и выйти на построение методов и моделей нелинейного программирования. В ее работах построена нелинейная целевая функция в таком виде (квазиэкспонента с гиперболическим показателем), что решение задачи математического программирования лежит внутри некоторых ограничений (например, надежность: $0<p<1$) за счет спецификации модели целевой функции и информационной жесткости других ограничений. Этот подход в оптимизации, назван в работах И. Совтус модифицированным методом неопределенных множителей.

Далее исследовательские интересы сосредоточились на распространение этого метода на различные сложные структуры соединения элементов в системе, с различным характером резервирования элементов. Причем везде окончательный результат обосновывался аналитически строго доказанными утверждениями, с обязательным построением алгоритма сходящегося двупараметрического процесса итераций к реальному решению, существование и единственность которого обосновывается, как правило, предварительно.

Позже в работах И. Совтус впервые рассматривается новый взгляд на структуру сложной системы (технической, экономической и др.) со стохастической связью ее элементов; в отличие от классического взгляда и общепринятого подхода моделирования такой стохастичности исходными стохастическими дифференциальными уравнениями, здесь стохастичность интерпретировалась как событие связи соседних элементов, что позволило существенно повысить адекватность решений-моделей, ибо это позволяло легко оценивать вероятности этих событий эмпирической частотой (по структуре потока требований, обслуживаемых системой), что, в свою очередь, дает содержательную и прозрачную интерпретацию параметров модели – результата решения задачи.

В последнее время исследовательские интересы привлекали проблемы оптимизации организационных структур, которые находятся на стыке технических и экономических наук. Полученные графо-аналитичиские результаты с прикладной интерпретацией: «Центр, периферия и оптимальное место менеджера в организационной структуре», «Топология экономических структур: графо-аналитичиский анализ и моделирование » и др. она считала весьма перспективным в экономико-математических исследованиях, работала над совершенствованием их аналитической обозримости и прикладной направленности.

Особая исследовательская любовь последних нескольких лет – это «вероятностная теория экономического риска», элементы которой планировались содержанием разделов второй докторской диссертации, которую она планировала защитить на экономические науки. Она была искренней в науке, содержанием ее выступлений на конференциях и в статьях была позиция отрицания марксистких определений теории риска, которые нормативно Минвузом Украины реализованы в учебниках и пособиях для экономических вузов. Гражданственность и патриотизм были всегда присущи Совтус И.К. Всю сознательную жизнь она боролась за утверждение украинского языка, была отмечена наградами, весной 2005 года была приглашена и принимала участие в парламентских слушаниях (Верховный Совет Украины).

И вот ее не стало. На взлете творческих сил и порывов наступил обрыв и ушла из жизни Она, оставив нас наедине с невыносимыми страданиями о неповторимости духовной личности и невозвратимости щемящей мечты новых книг о структурных многообразиях, о совершенстве неопределенных множителей, о топологии экономических структур, о графоаналитических моделях, о жизненных рисках, и еще о многом, о чем знала и мечтала Инна Кузьминична. Лишь она одна! И не всем этим успела с нами поделиться. Талант своеобразия общения, его уникальность, неповторимость личности педагога заставляли сквозь умело построенные парадоксы к активизации мышления слушателей, неповторимость интеллектуального и духовного обаяния, неповторимость взаимной искренности и доброжелательности, и еще многие душевные и другие качества, которые гармонично и по-своему уникально сочетались в ее нетривиальной натуре. Возможно и наверняка со временем будут разрешены проблемы, волновавшие ее творческую натуру. Но никогда не появится в их разрешении присущая ей творческая манера, ее изысканность в неожиданных тонкостях аналитических преобразований при обосновании утверждений, ее эмоциональное наполнение содержания результатов, что граничило с сопереживанием творений искусства.

Публикации И. Совтус (книги, статьи, авторские свидетельства на изобретения и патенты в открытой и закрытой печати) составляют более 150 наименований. Наиболее характерны для ее творческой личности:

1. Высшая математика. Методические рекомендации. К.: КВИРТУ. 1990 – 260 с. (соавтор.)
2. Метод повышения адекватности и достоверности моделей вероятностной оценки надежноси системы. // Статистические методы обработки сигналов – Сб. научн. стат. К.: КИИГА, 1992. – С. 54-65.
3. Сборник задач: спецглавы высшей математики. К.: КВИРТУ, - 1992. – 506 с. (соавтор.)
4. Построение вероятностной модели состояний системы с управляемым восстановлением. // Проблемы моделирования и цифровой обработки сигналов. – Сб. научн. труд. – К.: КМУГА, 1995.
5. Стохастические модели последействия отказов систем обработки информации. – Зб. наук. праць. К.: КВІУЗ, №2, - 1998. – С. 120-132.
6. Метод производящих функций в моделях оценивания входящей способности информационных систем. // Защита информации. – Сб. науч. труд. К.: КМУЦА, - 1999. – С. 180-186.
7. Марківська однорідна по другій компоненті стохастична модель надійності. // Актуальні проблеми автоматизації та інформаційних технологій. Т.1. Зб. наук. праць.–Дніпропетровськ,:Навч.Книга. - 1999.
8. Метод оптимізації розробки послідовно зв'язаних резервованих систем. // Захист інформації. №4(5). К.: КМУЦА, - 2000. – С. 34-42
9. Стохастичні моделі інформаційних технологій оптимізації надійності складних систем. Монографія. К.: КВІУЗ, - 2000. – 316 с. (соавтор)
10. Ризик і надійність. Економіко-стохастичні методи і моделі побудови систем. Монографія. К.: КНЕУ, 2000 – 212 с. (соавтор)
11. Ризик і надійність. Альтернатива категорій та проблеми їхньої формалізації. // Вісник КМУЦА. - № 3(4). К.: КМУЦА, 2000. – С.306-310
12. Нелінійний метод оптимізації неідентично резервованих систем обробки інформації на етапі їх розробки. // Захист інформації, № 3(8). К.: КМУЦА, 2001. – С.31-44
13. Математическая модель реконструкции сложных технико-экономических структур.-УСіМ.–2002.- № 2.
14. Технічні системи обробки економічної інформації: сучасні структури неідентичного резервування. // Вчені записки. Вип.4. К.: КНЕУ, 2002. – С.280-291
15. Складні системи з випадковою зв'язністю: ймовірнісне моделювання та оптимізація. Монографія. К.: КНЕУ, 2003. – 374 с. (соавтор)

М.Т. Корнийчук

CONSECUTIVE ALGORITHM OF DEFINITION OF THE ORDER NEAREST TO THE SET ANY RELATION BETWEEN OBJECTS

Grigory Gnatienko

Abstract: *The problem of a finding of ranging of the objects nearest to the cyclic relation set by the expert between objects is considered. Formalization of the problem arising at it is resulted. The algorithm based on a method of the consecutive analysis of variants and the analysis of conditions of acyclicity is offered.*

Keywords*: ranking, the binary relation, acyclicity, basic variant, consecutive analysis of variants*

Introduction

Many practical problems cannot be solved without application of expert estimations. One of the most widespread approaches at an expert estimation of objects is their ordering.

The problem of ordering of set of objects in degrees of display of some properties is one of the primary goals of expert reception of estimations [Литвак, 1983]. The essence of a problem will consist in definition of the full order on set of compared objects under the set partial order.

Among problems of decision-making the problem of linear ordering of objects is allocated with a plenty of concrete applications and a unconditional urgency of a theme. This problem traditionally is in the center of attention of researchers and the quantity of the works devoted to questions of construction optimum in this or that sense of linear orders on set of compared objects is very great [Миркин, 1976].

Practical application of problems of ranging is very various [Левин, 1987]. Such problems arise, for example, at the decision of

- a problem of definition of sequence of loading and unloading of a transport spacecraft;
- a finding of sequence of elimination of malfunctions of various systems;
- the complex analysis of quality of production;
- the analysis of characteristics of production and allocation of the main parameters of quality;
- a finding of bottlenecks in some complex systems possessing such properties as stability, controllability, self-organizing;
- designing of liaison channels between units in information networks;
- expert reception of estimations of projects of development of branches of a national economy or scientific researches;
- planning of building of residential areas, etc.

Problems of qualitative and quantitative ranging are considered. At the decision of such problems wide application was received with a method of pair comparisons. The set of works [Гнатиенко, 1993] is devoted to the analysis of the specified problems.

The problem of definition of ranging of objects is the widespread problem of the theory of decision-making. This problem is solved various methods. At application of expert estimations for ranging objects the information both from one expert and from expert group can be used.

As the person frequently supposes infringement of a condition of transitivity of relations at estimations of objects even at the decision of a problem of ranging one expert in relations between objects can appear cycles. In such cases there is a problem of definition of the ranking nearest to the relation set by the expert.

Problem Definition

Let it is necessary to find ranking (the linear order) n objects of set A, the nearest to the cyclic relation set by a matrix of pair comparisons

$$P = \left(p_{ij}\right), \quad i, j \in I = \{1,\ldots,n\}. \tag{1}$$

Elements of a matrix P express result of comparison of objects $a_i \in A,\ \ i \in I,$ with indexes $i, j \in I,$ and are defined thus:

$$p_{ij} = \begin{cases} 1, & if\ a_i \succ a_j, \\ 0, & if\ i = j, \\ -1, & if\ a_i \prec a_j, \end{cases}$$

$$p_{ij} + p_{ji} = 0,\ \ \forall i, j \in I,\ \ i \neq j,$$

where "$\succ$" - a symbol of the relation of preference between objects.

Construction of the linear order generally demands entering enough the big changes in initial structure of preferences of a kind (1) on set of objects A. The problem of a finding of the order is a complex combinatory problem, NP - difficult in strong sense [Миркин, 1976]. Therefore algorithms of local optimization, heuristic algorithms or the algorithms basing a method of branches and borders are applied to construction of ranking R*. Methods of the consecutive analysis of variants in the offered interpretation for this class of problems were not applied, though their use in this area of researches is perspective.

The linear order nearest to the set relation of a kind (1), we shall search as

$$R^* = Arg \min_{R \in \Re} d(P, R),$$

where $\Re$ - set of matrixes which correspond to linear orders n objects, d(R,P) – distance between ranking $R \in \Re$, which is under construction, and set cyclic relation P of a kind (1).

For measurement of distance between set relation P and ranking R, we shall use the most widespread in this class of problems metrics Hamming

$$d(P, R) = 0{,}5 \sum_{i \in I} \sum_{j \in I} \left| p_{ij} - r_{ij} \right|,$$

where p_{ij}, r_{ij} – accordingly elements of matrixes P and R.

Formalization of a Problem

As matrixes of relation P and R are slanting symmetric they can be written down as vectors C and X with elements

$$\begin{aligned} & c_t = p_{ij},\ \ x_t = r_{ij}, \\ & t = (i-1)n + j - (i+1)i/2,\ \ 1 \le i < j \le n. \end{aligned} \tag{2}$$

Then the distance between relations P and R will be written down as

$$d(P, R) = \sum_{j \in J} \left| c_j - x_j \right|,\ \ j = 1, \ldots, n(n-1)/2 = N,\ \ J = \{1, \ldots, N\}.$$

The problem of a finding of the linear order nearest to set on set of objects to the relation (1), is formalized as

$$\sum_{j \in J} \left| c_{ij} - x_{ij} \right| \ \rightarrow \ \min, \tag{3}$$

$$x_j \in X_j^0 = \{-1, 1\},\ \ j \in J, \tag{4}$$

$$x \in D^A \subset X^0,\ \ X^0 = \prod_{j \in J} X_j^0, \tag{5}$$

where D^A – set of vectors of a kind (2) which correspond{meet} to acyclic relations between objects.

Specificity of a problem (3) - (5) will be, that its decision $x \in X$ should satisfy to a condition of acyclicity as the relation which is set by matrix R, should belong to a class of linear orders.

We shall consider a chain of objects a_{i1}, a_{i2}, a_{i3} with the set relations of preference which we shall designate symbols $a_{i1}\ \pi\ a_{i2}\ \pi\ a_{i3}$, $i_1,i_2,i_3 \in L$, $\pi \in \{\succ,\prec\}$.

Basic sub-variant b, which is generated by the three of objects (a_{i1}, a_{i2}, a_{i3}), we shall name elements of a vector of a kind (2) with components

$$b = (c_{j1}, c_{j2}, c_{j3}),\ 1 \le j_1 < j_2 < j_3 \le N,\ \ c_{j1}, c_{j2}, c_{j3} \in c,\ \ c_{j1}, c_{j2}, c_{j3} \in \{-1,1\}, \tag{6}$$

which values answer relations of a kind

$$\left(a_{i1}\ \pi\ a_{i2},\ \ a_{i2}\ \pi\ a_{i3},\ \ a_{i3}\ \pi\ a_{i1}\right),\ \ a_{i1}, a_{i2}, a_{i3} \in A,\ \ \pi \in \{\succ,\prec\}.$$

The basic sub-variant is a minimal subset of objects from set A, on which it is possible to reveal a cycle.

Allowable basic sub-variant we shall name a basic sub-variant which is generated by the three of objects, relations between which satisfy to a condition of acyclicity.

Full variant (variant of length N) we shall name a vector which answers the full binary relation on set of objects.

Allowable variant x^D we shall name a full variant which answers the acyclic relation on set of all n objects, that is $x^D \in D^A$.

At check of an admissibility of basic variants of a kind (6) which are formed by objects a_{i1}, a_{i2}, a_{i3}, $1 \le i_1 < i_2 < i_3 \le n$, it is necessary to consider relations as ($a_{i1}\pi\ a_{i2}$, $a_{i2}\pi\ a_{i3}$, $a_{i1}\underline{\pi}\ a_{i3}$), where $\underline{\pi}$ - inversion of the relation π: $a_{i1}\underline{\pi}\ a_{i3} \Leftrightarrow a_{i3}\pi\ a_{i1}$, $\pi = \{\succ,\prec\}$.

Set $X^s = \{\cup X^s_j,\ j=1,...,N\}$, $X^s \subseteq X^0$, s=1,2,..., we shall name reduced (concerning initial set X^0).

For the decision of a problem (3)-(5) procedure of reduction of set of allowable decisions Z on a condition $X^s = (X^s_1 \times X^s_2 \times \ldots \times X^s_N)$ of acyclicity of the relation which corresponds to the decision of a problem of a finding of strict resulting ranging of objects of set A is used. The analysis of variants in view of a condition of acyclicity of the decision is carried out with use of the procedure described in [Гнатиенко, 2005].

Let's designate set of all possible values which can get elements of a basic variant, through B^0. Sets of a kind B^0 are formed of set X^s by association of three various columns of a matrix X^s: $X^0_{j1} \cup X^0_{j2} \cup X^0_{j3}$, $1 \le j_1 < j_2 < j_3 \le N$, $X^0_{j1}, X^0_{j2}, X^0_{j3} \in X^s$. Capacity of this set is equal $|B^0| = 6$.

Basic set $B^0 = B^0_1 \times B^0_2 \times B^0_3$, $B^0_i = (-1,1)^T$, i=1,...,3, we shall name set of elements of a matrix of which values of basic vectors get out.

Columns of basic set $B^0_i = (-1,1)^T$, i=1,2,3, we shall name subsets of basic set.

The reduced basic set B^s, $B^s \subset B^0$, s=1,2,..., we shall name a matrix which is formed of a matrix B^0 by removal from it separate elements.

It is known [Макаров, 1982], that for matrixes of pair comparisons with elements of a kind (2) requirement of absence of cycles is equivalent to the requirement of absence of cycles of length three (T=3).

As the top triangular matrix of a matrix P^i, $i \in I$, contains the full information on all matrix indexes of objects need to be considered only on increase, that is $1 \le i_1 < i_2 < i_3 \le n$. Indexes of elements of a vector of relations between objects also satisfy to conditions $1 \le j_1 < j_2 < j_3 \le N$.

Let's designate through ψ function of two arguments which values are calculated under the formula

$$j = \psi(i_1, i_2) = (i_1 - 1)n + i_2 - (i_1 + 1)i_1 / 2,\ 1 \le i_1 < i_2 < n.$$

Algorithm

In a problem of definition of the linear order nearest to the set cyclic relation, it is possible to present algorithm of the consecutive analysis and elimination of inadmissible elements in the following kind.

Step 1. Let's put initial values of the decision of a problem equal $x(j) = c(j),\ \ j \in J$.

Step 2. The organization of three enclosed cycles: i:=1 до n-2; i_1:=i+1 до n-1; i_2:=i_1+1 до n. Variables of cycles i, i_1, i_2 are indexes of objects. In a body of these cycles the following steps are executed.

Step 3. Definition of indexes of elements *j, j_1, j_2* the current basic set B^0_j on indexes of objects *i, i_1, i_2*: j=ψ(i,i_1), j_1=ψ(i,i_2), j_2=ψ(i_1,i_2).

Step 4. Definition of three of objects, relations between which form cycles. Quantity of all three n objects equally: k3=n*(n-1)*(n-2)/6.

Quantity of cycles on set n objects it is equal d=(n^3-4n)/24 for even and d=(n^3-n)/24 for odd values n.

Step 5. Generation of a vector of indexes of participation of relations between objects in cycles: $vc(j),\ \ j \in J.$

That is, value $vc(j),\ \ j \in J,$ is equal to quantity of ocurrences of the relation with an index $j,\ \ j \in J,$ in cyclic three.

Step 6. Definition of values of vectors of indexes $vic(j),\ \ j \in J,$ participations of the inverted relations $x(j) = c(j) - 2,\ \ j \in J.$

The choice of an index of the relation between objects is carried out in view of three criteria:

K_1 – inversion of the relation does not generate new three;

K_2 – the total quantity of cycles for the inverted relation is minimal;

K_3 – the difference of quantity of cycles for the set relation and inverted is minimal.

Step 7. Definition of relations, which replacement on inverted, as much as possible reduces quantity of cycles.

Step 8. Choice of an index of the relation for decision-making on its final inverting.

Step 9. The termination of cycles on *i, i_1, i_2*.

Step 10. Recurrence of points 1-9 of the resulted algorithm until in the decision $x(j),\ \ j \in J,$ problems (3)-(5) exist cycles.

Conclusion

The resulted algorithm allows finding consistently for final quantity of steps the ranking of the objects nearest to the cyclic relation set by the expert between objects.

Computing experiments have confirmed efficiency of the resulted algorithm. Received with the help of algorithm of the decision are one of the rankings, the nearest to the cyclic relation set by the expert on set of objects.

Bibliography

[Литвак, 1983] Литвак Б.Г. Меры близости и результирующие ранжирования // Кибернетика. 1983. №1. С.57-63.

[Миркин, 1976] Миркин Б.Г. Анализ качественных признаков: Математические модели и методы. М.: Статистика. 1976. 166 с.

[Левин, 1987] Левин М.Ш. Современные подходы к оценке эффективности плановых и проектных решений в машиностроении. Обзорная информация. Сер.С-9. Автоматизированные системы проектирования и управления. М.: ВНИИТЭМР. 1987. Вып.3.-56с.

[Гнатиенко, 1993] Гнатиенко Г.Н., Микулич А.Ю. Методы метризации качественных ранжировок объектов. Киев.ун-т.-Киев.1993. Библиогр.: 6 назв. Рус. Деп.в УкрНИИНТИ 10.03.93. №432-Ук93.-10с.

[Гнатиенко, 2005] Гнатієнко Г.М. Процедура послідовного аналізу та відсіювання варіантів з урахуванням ациклічності розв'язку//Наукові праці Кіровоградського державного університету. Технічні науки.-Вип.16.-Кіровоград. 2005.-С.294-299.

[Макаров, 1982] Макаров И.М., Виноградская Т.М., Рубчинский А.А. и др. Теория выбора и принятия решений: Учебное пособие. М.:Наука, 1982.-328с.

Author's Information

Grigoriy M. Gnatienko *– T. Shevchenko Kiev National University, Faculty of Cybernetics, Dr.Ph., Kiev, Ukraine; e-mail: G.Gnatienko@veres.com.ua*

СПОСОБЫ ПРЕДСТАВЛЕНИЯ МАТРИЦ ОТНОШЕНИЙ МЕЖДУ ПАРАМИ ОБЪЕКТОВ И ПРЕВРАЩЕНИЯ МЕЖДУ НИМИ

Григорий Н.Гнатиенко

Аннотация: *Описаны способы представления отношений между парами объектов при целостном выборе. Рассматриваются методы выявления и виды отношений между объектами. Приводится таблица соответствий между различными формами представления отношений.*

Ключевые слова: *парные сравнения, ранжирование, тождественные преобразования, балльная оценка, эксперт.*

Введение

На практике часто возникают задачи принятия решения, в которых некоторые свойства объектов удобнее выражать не в терминах параметров и их значений, а в терминах отношений между объектами по некоторому свойству [Миркин, 1980]. Поэтому распространенной проблемой при обработке экспертной информации является проблема определения отношений на заданном множестве объектов. При этом существуют разнообразные способы представления указанных отношений и вычисление соответствий между ними также имеет существенное значение при решении задач принятия решений.

Известны многочисленные результаты систематических исследований задачи сравнения друх объектов и выделения «лучшего» из них. Эти результаты свидетельствуют о том, что такая операция является сложной для эксперта, если объект характеризуется большим количеством параметров. Уже при наличии трех характеристик объектов эксперты используют упрощающие задачу эвристики, которые могут приводить к противоречиям. Эти ограничения свойственны человеку в силу специфических характеристик его оперативной памяти [Ларичев, 1980]. При этом, согласно [Larichev, 1980], в задачах целостного выбора возможности эксперта очень велики, поскольку он использует гештальт (целостный образ) объекта как одну структурную единицу информации. Гештальт, как правило, богаче соответствующего набора параметровю В связи с этим решения, принятые на основании целостного представления, часто не совпадают с решениями тех же задач формальными методами.

Постановка задачи

Пусть рассматривается множество n объектов

$$a_i \in A,\ \ i \in I = \{1,...,n\}, \tag{1}$$

параметры которых не выделяются. С учетом природы практической задачи целостный выбор осуществляется в связи с тем, что параметры объектов невозможно измерить, они неизвестны по некоторым причинам или являются несущественными для принятия решения. Эксперту предлагается определить отношения (предпочтения, сходства-различия, близости или другие) между объектами, используя личный опыт или некоторые иные косвенные свидетельства.

Одним из основных способов представления отношений между объектами множества (1) являются матрицы парных сравнений (МПС):

$$P = \left(p_{ij}\right),\ \ i, j \in I = \{1,...,n\}. \tag{2}$$

Элементами $p_{ij}, i, j \in I,$ матриц вида (2) являются действительные числа, отражающие в некоторой шкале результаты сравнения экспертом объектов с индексами $i, i \in I,$ и $j, j \in I.$

Симметрические элементы матриц p_{ij} и p_{ji} выбираются равными, если соответствующие им объекты равноценны с точки зрения эксперта. Если же объект с индексом $i, i \in I,$ по мнению эксперта, является «лучшим», чем объект с индексом $j, j \in I,$ то отношение между симметричными элементами матрицы устанавливается $p_{ij} > p_{ji},\ \ p_{ij}, p_{ji} \in P,\ \ i, j \in I.$ Кроме этих очевидных условий на элементы

матрицы вида (2), как правило, накладываются дополнительные (калибровочные) ограничения, которые однозначно связывают попарно симметрические элементы p_{ij} и p_{ji}.

В зависимости от условий задачи, значения элементов $p_{ij}, i, j \in I,$ матриц вида (2) могут иметь различный смысл. Матрица P может характеризовать относительный «вес» объектов, если определяется вектор предпочтений на множестве объектов (1), может указывать на относительную важность параметров объектов при принятии решений или свидетельствовать об относительной компетентности экспертов в паре $(i, j),\ i, j \in I.$

Метод парных сравнение и виды МПС

Одним из наиболее распространенных и наиболее надежных ([Миркин, 1974]) методов выявления отношений на заданном множестве объектов (1) является метод парных (иногда употребляется термин – попарных [Паниотто, 1986], [Юшманов, 1990]) сравнений ([Дэвид, 1978], [Литвак, 1982]). При использовании этого метода результаты экспертизы заносятся в МПС вида (2) или представляются в виде ориентированного графа парных сравнений, вершинами которого являются объекты, а дуги характеризуют отношения между ними.

Отношения на заданном множестве объектов выявляются также путем использования методов множественного сравнения ([Паниотто, 1982]), ранжирования ([Миркин, 1974]), приписывания баллов ([Кини, 1981]) и других методов. При этом МПС являются наиболее общим способом представления отношений на множестве объектов ([Миркин, 1980]).

МПС вида (2) могут быть полными (когда все элементы матрицы P полностью определены) или неполными, то есть такими, в которых не все элементы $p_{ij} \in P,\ i, j \in I,$ известны. В последнем случае может возникать задача восстановления неизвестных элементов МПС ([Загоруйко, 1999]), а также определения «веса» объектов по неполной МПС [Чеботарев, 1989].

В реальных экспертизах специалисты не всегда последовательны в своих предпочтениях вследствие сложности задачи, неопределенности отношений между объектами, недостаточной компетентности, личной предубежденности и прочее. Закономерным следствием субъективности экспертов является неточность, размытость и противоречивость экспертных суждений. Поэтому елементы матрицы вида (2) иногда представляются в интервальном виде или в виде функций принадлежности нечеткому множеству. Однако в этой работе будем рассматривать только точечные значения элементов МПС.

Способы представления отношений между объектами

Согласно [Миркин, 1980], при сравнении объектов множества (1) существует четыре основных способа представления результатов такого сравнения в виде элементов МПС вида (2). Оценка экспертом отношения между объектами может выражать:

П1) просто факт предпочтения эксперта одного объекта другому или равноценности между объектами (простая структура) ([Кемени, 1972], [Кендэлл, 1975], [Литвак, 1982]);

П2) долю суммарной интенсивности предпочтения сравниваемых объектов, которая приходится на каждый из них ([Литвак, 1982]), так что $p_{ij} + p_{ji} = T,\ i, j \in I,$ где $T \geq 0$ – некоторое действительное число, одинаковое для всех $p_{ij} \in P,\ i, j \in I;$ чаще всего $T = 1$ и тогда говорят, что применяется вероятностная калибровка; при $T = 0$ имеет место кососимметрическая калибровка, а при $T > 0$ – турнирная калибровка;

П3) балльную оценку отношения ([Кини, 1981]) $p_{ij} \in R,\ p_{ji} \in R,\ i, j \in I,$ где R – множество действительных чисел; иногда устанавливаются односторонние или двухсторонние границы допустимого приписывания баллов;

П4) во сколько раз один объект превосходит другой, то есть $p_{ij} = 1 / p_{ji},\ i, j \in I,$ – говорят, что имеет место степенная калибровка ([Миркин, 1980], [Белкин, 1990]).

Важной характеристикой метризованных отношений является их сверхтранзитивность или кардинальная согласованность в силе предпочтения, которая состоит в выполнении условий: $p_{ij} > 0$ и $p_{ij} p_{jk} = p_{ik}$, $i, j, k \in I.$

Если отношения между парами объектов задаются в формах П1) или П2), то матрица вида (2) является кососимметрической (антисимметрической): $p_{ij} = -p_{ji},\ i, j \in I,$ или легко сводится к такой. Если же отношения заданы в форме П3) или П4), то матрица (2) является обратносимметрической: $p_{ij} = 1/p_{ji},\ i, j \in I,$ или сводится к ней.

Систематизированные и разработанные автором формулы превращения между различными способами представления попарных отношений между объектами приводятся в таблице 1. В таблице 1 через $p_{ij}, i, j \in I,$ обозначены исходные значения элементов матрицы вида (2). Через $r_{ij}, i, j \in I,$ – результирующие значения этих элементов при решении задачи превращения их в требуемую форму представления.

			Простая структура ПС {0,1/2,1)	Простая структура ПС {-1,0,1)	Простая структура ПС {0,1,2)	Балльная структура (Б)	Степенная калибровка (С)
Качественные отношения	Простая структура (ПС)	Простая структура ПС {0,1/2,1)	—	$r_{ij}=2(p_{ij}-1/2)$	$r_{ij}=2p_{ij}$	МЕТОДЫ МЕТРИЗАЦИИ	
		Простая структура ПС {-1,0,1)	$r_{ij}=(p_{ij}+1)/2$	—	$r_{ij}=2p_{ij}+1$		
		Простая структура ПС {0,1,2)	$r_{ij}=p_{ij}/2$	$r_{ij}=p_{ij}-1$	—		
Количественные метризованные отношения	Балльная оценка (взвешенная структура) $p_{ij}\geq 0, \forall i,j\in I,$ (Б)		$r_{ij}=[\text{sign}(p_{ij}-p_{ji})+1]/2$	$r_{ij}=\text{sign}(p_{ij}-p_{ji})$	$r_{ij}=\text{sign}(p_{ij}-p_{ji})+1$	—	$r_{ij} = p_{ij} / p_{ij}$
	Мультипликативные матрицы	Степенная калибровка $p_{ij}*p_{ji}=1,\ p_{ij}>0,\ \forall i,j\in I,$ (С)	$r_{ij}=[\text{sign}(p_{ij}-p_{ji})+1]/2$	$r_{ij}=\text{sign}(\log_T p_{ij}),\ T>1$	$r_{ij}=\text{sign}(p_{ij}-p_{ji})+1$	$r_{ij}=T_1 p_{ij}/(T_2+p_{ij}),$ $T_1>0,\ T_2\geq 1$	—
	Аддитивные матрицы	Турнирная калибровка $p_{ij}+p_{ji}=T,\ \forall i,j\in I,$ (Т)	$r_{ij}=[\text{sign}(p_{ij}-p_{ji})+1]/2$	$r_{ij}=\text{sign}(p_{ij}-p_{ji})$	$r_{ij}=\text{sign}(p_{ij}-p_{ji})+1$	$r_{ij}=p_{ij}$	$r_{ij} = p_{ij} / p_{ij}$
		Вероятностная калибровка (В) $p_{ij}+p_{ji}=1,\ 0\leq p_{ij}\leq 1,\ \forall i,j\in I,$	$r_{ij}=[\text{sign}(p_{ij}-p_{ji})+1]/2$	$r_{ij}=\text{sign}(p_{ij}-p_{ji})$	$r_{ij}=\text{sign}(p_{ij}-p_{ji})+1$	$r_{ij}=Tp_{ij},$ $T>0$	$r_{ij} = p_{ij} / p_{ij}$

			Турнирная калибровка (Т)	Вероятностная калибровка (В)	Кососимметрическая калибровка (К)
Качественные отношения	Простая структура (ПС)	Простая структура ПС {0,1/2,1)	МЕТОДЫ МЕТРИЗАЦИИ		
		Простая структура ПС {-1,0,1)			
		Простая структура ПС {0,1,2)			
Количественные метризованные отношения	Балльная оценка (взвешенная структура) $p_{ij}\geq 0, \forall i,j\in I,$ (Б)		$r_{ij} = p_{ij} + (\max_{i,j} S_{ij} - S_{ij})/2;$ $r_{ij} = p_{ij} + (p_{ij} * \max_{i,j} S_{ij})/S_{ij};$ $S_{ij} = p_{ij} + p_{ji}.$	$r_{ij} = p_{ij} + (\max_{i,j} S_{ij} - S_{ij})/2;$ $r_{ij} = p_{ij} + (p_{ij} * \max_{i,j} S_{ij})/S_{ij};$ $S_{ij} = p_{ij} + p_{ji}.$ $r_{ij} = p_{ij}/(p_{ij} + p_{ji}).$	$r_{ij} = p_{ij} - S_{ij};$ $S_{ij} = p_{ij} + p_{ji};$ $r_{ij} = (p_{ij} - p_{ji})/2.$
	Мультипликативные матрицы	Степенная калибровка $p_{ij}*p_{ji}=1,\ p_{ij}>0,\ \forall i,j\in I,$ (С)	$r_{ij} = p_{ij}/(1+p_{ij});$ $r_{ij} = T\cdot p_{ij}/(1+p_{ij});$ $T>0.$	$r_{ij} = p_{ij}/(1+p_{ij}).$	$r_{ij} = \log_T p_{ij};$ $T>1.$
	Аддитивные матрицы	Турнирная калибровка $p_{ij}+p_{ji}=T,\ \forall i,j\in I,$ (Т)	—	$r_{ij} = p_{ij}/(p_{ij}+p_{ji}).$	$r_{ij} = p_{ij} - p_{ji};$ $r_{ij} = (p_{ij} - p_{ji})/2.$
		Вероятностная калибровка (В) $p_{ij}+p_{ji}=1,\ 0\leq p_{ij}\leq 1,\ \forall i,j\in I,$	$r_{ij} = Tp_{ij},$ $T>0.$	—	$r_{ij} = T(p_{ij} - p_{ji});$ $T>0.$
		Кососимметрическая калибровка $p_{ij}+p_{ji}=0,\ \forall i,j\in I,$ (К)	$r_{ij} = p_{ij}(sign\ p_{ij}+1)/2 +$ $+(\max_{i,j} p_{ij} - \lvert p_{ij}\rvert)/2;$ $r_{ij} = T(p_{ij} + \max_{i,j} p_{ij});$ $T>0.$	$r_{ij} = (p_{ij} + \max_{i,j} p_{ij})/$ $/(2\max_{i,j} p_{ij})$	—

Таблица1. Тождественные преобразования между способами представления отношений между объектами.

Соответствие между формами представления отношений между объектами

Отношения, представленные в форме П1), являются заданными в качественной (квалиметрической) шкале. Последние три формы представления отношений между объектами, которые выражают количественную меру отношений, называют метризованными и говорят, что они отображают интенсивность отношений. Форма П2) называется еще аддитивным, а форма П4) – мультипликативным отношением. Между формами П2), П3), П4) существует взаимооднозначное соответствие ([Миркин, 1980], [Хованов, 1982]).

Для обработки результатов измерения количественных величин используется аппарат математической статистики. Для обработки статистическими методами результатов измерений в качественных шкалах, необходимо нечисловую информацию метризовать ([Литвак, 1982], [Бевз, 1989]), то есть погрузить в систему, производную от действительных чисел. Метризацией (оцифровкой ([Бевз, 1989], [Хованов, 1986]), арифметизацией ([Хованов, 1982])) квалиметрической шкалы называется построение соответствия между формой П1) и остальными формами. Не каждая реляционная система может быть изоморфно метризованной [Гильбурд, 1988]. С другой стороны, некоторые квалиметрические шкалы могут быть метризованы различными способами. Методы метризации квалиметрических отношений приводятся, например, в работах [Бевз, 1989], [Гнатиенко, 1993].

Библиография

[Бевз, 1989] Бевз С.Н. Непротиворечивая метризация качественных признаков//Автоматика, 1989,№3,С.17-23.

[Гильбурд, 1988] Гильбурд М.М. Об эвристических методах построения медианы в задачах группового выбора // Автоматика и телемеханика. 1988. №7. С.131-136.

[Гнатиенко, 1993] Гнатиенко Г.Н., Микулич А.Ю. Методы метризации качественных ранжировок объектов //Киев.ун-т.-Киев,1993.-10с.-Библиогр.: 6 назв. - Рус. - Деп. в УкрНИИНТИ 10.03.93, №432-Ук93.

[Дэвид, 1978] Дэвид Г. Метод парных сравнений. - М.: Статистика.1978.144 с.

[Загоруйко, 1999] Загоруйко Н. Г. Прикладные методы анализа данных и знаний. Новосибирск, издательство института математики, 1999.

[Кемени, 1972] Кемени Дж.Г., Снелл Дж.Л. Кибернетическое моделирование. М.: Советское радио. 1972. 192 с.

[Кендэлл, 1975] Кендэлл М.Дж. Ранговые корреляции. М.: Статистика.1975.214 с.

[Кини, 1981] Кини Р.Л., Райфа Х. Принятие решений при многих критериях: предпочтения и зимещения. М., 1981. 560 с.

[Ларичев, 1980] Ларичев О.И., Мошкович Ш.М. О возможностях получения от человека непротиворечивых оценок многомерных альтернатив/ Дескриптивный подход к изучению процессов принятия решений при многих критериях//Сб. трудов ВНИИСИ.1980.№9.С.58-66.

[Литвак, 1982] Литвак Б.Г. Экспертная информация: Методы получения и анализа. М.: Радио и связь. 1982. 184 с.

[Миркин, 1974] Миркин Б.Г. Проблема группового выбора. М.: Наука, 1974.-256с.

[Миркин, 1980] Миркин Б.Г. Анализ качественных признаков и структур. М.: Статистика. 1980. 319 с.

[Паниотто, 1982] Паниотто В.И., Максименко В.С. Количественные методы в социологических исследованиях. К.: Наукова думка. 1982. 272 с.

[Паниотто, 1986] Паниотто В.И. Качество социологической информации. Методы оценки и процедуры обеспечения.-К.:Наук.думка,1986.-207с.

[Хованов, 1982] Хованов Н.В. Математические основы теории шкал измерения качества/Изд-во ЛГУ. 1982.

[Хованов, 1986] Хованов Н.В. Стохастические модели теории квалиметрических хкал: Учебное пособие. Л.1986.80 с.

[Чеботарев, 1989] Чеботарев П.Ю. Обобщение метода строчных сумм для неполных парных сравнений//Автоматика и телемеханика. 1989. №8. С.125-137.

[Юшманов, 1990] Юшманов С.В. Метод нахождения весов, не требующий полной матрицы парных сравнений//Автоматика и телемеханика. 1990. №2. С.187-189.

[Larichev, 1989] Larichev O., Boichenko V., Moshkovich H., Sheptalova L. Modolling multiattribute information processing strategies a binary decision task/Org. Behav. and human perf. V.26. 1980.

Информация об авторе

Григорий Н. Гнатиенко *– Киевский университет им.Т.Шевченко, факультет кибернетики, докторант. Киев, Украина; e-mail: G.Gnatienko@veres.com.ua*

CLUSTER SUPERCOMPUTER ARCHITECTURE

Andrey Golovinskiy, Sergey Ryabchun, Anatoliy Yakuba

***Abstract**: The paper describes the architecture of supercomputer system of cluster type SCIT and the base architecture features used during this research project. This supercomputer system is put into research operation in Glushkov Institute of Cybernetics NAS of Ukraine from the early 2006 year. The paper may be useful for those scientists and engineers that are practically engaged in a cluster supercomputer systems design, integration and services.*

***Keywords**: supercomputer, cluster, computer system management, computer architecture.*

***ACM Classification Keywords**: C.1.4 Parallel Architectures. C.2.4 Distributed systems, D.4.7 Organization and Design*

1. Introduction

In 2004-2005 years small developer team from the Glushkov Institute of Cybernetics NAS of Ukraine built and put into research operation two high-performance supercomputer systems with cluster architecture SCIT-1 and SCIT-2 on the basis of modern unicore Intel microprocessors. The developed supercomputers allow to solve essentially new challenges of the big dimension in the field of a science, economy, ecologies, an agriculture, in space branch and other branches.

Dynamics of managerial processes during task flow computing inside a supercomputer system, adaptation of existing and creation of new architectural means for maximization of global characteristics of supercomputer productivity is the investigation objects in the current work of research team.

2. Architecture Components of Cluster Supercomputer

The architecture of the multiserver, cluster system is a multi-plane combination of hardware-software means, particularly at the level of interaction of the server operating systems, distributions of processes of computation on processors and synchronization of these processes, effective maintenance of queries to the centralized or distributed files systems.

Specific of tasks for the cluster computing. The supercomputer of cluster type is the computer system with asymmetrical multiprocessing and strongly connected nodes and task, intended for execution in a such computation environment, have features:

- each task consists of great number of interactive processes having an *identical* code, they are started on the different cluster nodes and each of them executes some part of *common* work;
- during the task execution processes can exchange intensively between themselves;
- interprocess data exchange results in smoothing of productivity of each process on speed of the *slowest*;
- each process, as a rule, occupies the large volume of main memory for the period of execution.

A cluster task is located in the user domestic catalogue and for implementation is got on competition basis two cluster resources - processor resource and timing resource. The long continuous execution of task is connected to an opportunity to not receive results in time (for example, from failures or at the large load by other tasks), therefore the double timing resources is given tasks – both full time of execution of task as the session time i.e.

time of noninterapted execution, after a session a check point keeping received intermediate result should be formed. The technology of programing with check points is one of basic components of organization of task execution, as allows repeatedly to interrupt execution and to proceed in it on intermediate results.

Simplified structure of supercomputer. The supercomputer of cluster type is the array of computing nodes, each of which is a multiprocessor server with symmetric mulitprocessing in the field of common main memory (SMP-architecture), incorporated by a few local areas networks of different purpose and productivity; from the array of computing nodes can be abstracted frontend servers (managing nodes) for the centralized process for handling of task executions. Besides this, there are the servers, specialized on the management by shareable files resources (file server) and external access of users to the cluster (access server) – see fig.1.

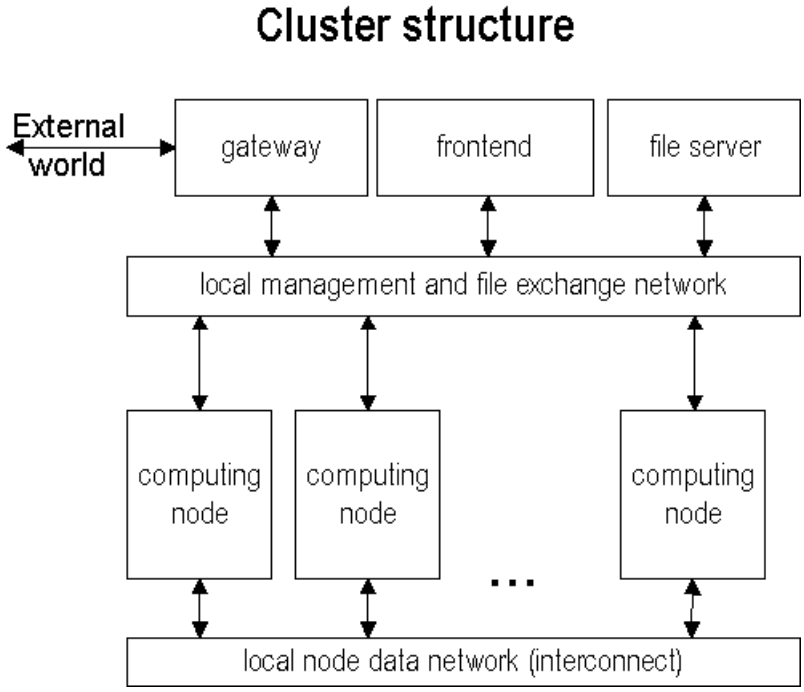


Fig. 1. Simplified cluster structure

Some hardware and software features of SCIT supercomputers are resulted in table 1.

Table 1. SCIT clusters hardware characteristics

	SCIT-1	SCIT-2
Computing nodes quantity	**24**	**32**
Node processor quantity	**48 (Xeon 2,67 GHz)**	**64 (Itanium2 1,4 GHz)**
Frontend quantity	**1**	**1**
Processor cache (Mbyte)	**1**	**3**
All main memory (GByte)	**48 (DDR SDRAM PC-2100 ECC)**	**64 (DDR SDRAM PC-2100 ECC)**
Interconnect network	**Infiniband**	**SCI (Scalable Coherent Interface)**
File managing network	**Gigabit Ethernet**	**Gigabit Ethernet**
Storehouse (TByte)	**1.6 (Common to both clusters)**	
Operating system (Linux)	**Fedora Core 4**	**CentOS 4.2**
Linux kernel	**2.6.12**	**2.6.12**
Global File system	**Lustre 1.4.5**	**Lustre 1.4.5**
Parallel programming system	**Open MPI**	**Open MPI, Scali**
Programmming language	**C, C++, Fortran-77**	**C, C++, Fortran-77**

The cluster operating system are chosen Linux FedoraCore4 and CENTOS 4.2, they are established both on frontend and computing nodes of clusters. As the root file system for computing nodes is used NFS, and as the distributed file system are chosen **Lustre** [1].

The cluster software accessible to the user includes for programming languages C/C++, Fortran compilers of the GNU and Intel different versions, for the parallel calculations - optimized libraries of ATLAS[2], BLACS[3], SCALAPACK[4], Intel MKL[5], application packages of GROMACS[6], WIEN2K[7], GAMESS[8] et al. As a parallel interface various realization of MPI interface - SCAMPI[9], OPENMPI[10] are used.

Task management and file exchange networks. In the environment of tasks management and file management two logical networks are allocated - management network (MN) and file exchange network (FEN). FEN service generally should give opportunities:

- remote management of computing node through protocol of WakeOnLan;
- access of node to the data on a network configuration (protocols of DHCP);
- loadings of the operating system in a computing node (protocols of TFTP);
- access of node to root file system (protocols of NFS);
- deliveries at the computing node of the task data (protocols of NFS).

MN service generally case provides an opportunity of access to the node from outside for:

- operative management by a node;
- receptions of statistical information on loading of processors, employment of memory, the indication of gauges of temperature, speed of rotation of fans;
- start and further control of task processes.

Let's consider in detail use of a network of data exchange for the process to start the cluster computing node. Each node is configured on inclusion at reception by the network interface of the special package *wake-on-lan* and on the load through a network interface by PXE-protocol. A frontend sends the formed package through FEN and a node initiates the load process.

A node sends the broadcast inquiry and from the server DHCP which is established on a frontend, receives all data necessary for loading system, downloads a kernel and minimum root file system from the TFTP server, which is also established on the frontend, unpacks a kernel and starts its implementation.

Farther the process of initializing of the system, being based on received on DHCP data, mounts on NFS file system located at storehouse, does it by a root and completes initializing, having transferred management to the starting scripts located on the new root file system. Since this moment loading of system on a network or from a local disk does not differ practically. Further additional sections NFS with working data, by users directories and etc necessarily are mounted.

Such scheme supposed that root file systems at all cluster nodes are the same, essentially facilitates administration, update, installation of the new software, as works with all cluster entirely, and on orders reduces an opportunity to make a mistake. Root file systems of all nodes are identical, except for a few catalogues which really should be unique at everyone, but also they are located in main memory of node. Processes of each activated task, working as everyone on a separate node, all the same work in the same catalogue located at the storehouse, read and write from/to the same files.

As we see, practically all on the file input-output work is done on the data exchange network, therefore the requirements to throughput of this network very high. It is necessary to specify, that a «bottle neck» in this chart is the network interface of server, throughput of the network interface of node suffices much.

The start of task execution on the cluster nodes can be carried out by various ways depending on a task, i.e. a MPI-task is started by the command of *mpirun*, and the ordinary not parallel program can be activated by the command of *ssh* or *rexec*. For the operative control after the started task state, for its forced completion and liberation of resources busy at a task access is also used to the node on protocol of SSH, it non-obvious implies, that the node should be accessible.

Necessity for the base remote management in each cluster node separately, an opportunity of implementation of such operations as startup-shutdown of node, the console with the output of load of node have demanded installation of **ServNET** [11]. Further it is planned to use nodes only with support of the IPMI interface [12] of

version above 1.5, node providing the remote startup-shutdown at presence only of the Ethernet cable and feed connected to the node, and the function of SERIAL-OVER-LAN in IPMI 2.0+ allows even remotely to adjust the node's BIOS.

IP-network. As supporting in a cluster systems an IP-network is used with a few ranges of private IP-addresses: 10.0.0.0 – 10.255.255.255, 172.16.0.0 – 172.31.255.255, 192.168.0.0 – 192.168.255.255, thus for the cluster nodes used private range is 10.0.0.0 – 10.254.254.254 as most spacious. The following chart of distributing of subnet of IP-addresses is applied in the last:

- A computing node has an IP-address 10.N.M.X, where N is number of cluster, M is number of switch, X is number of port in the switch. Thus, 10.1.1.1 is the first node of the cluster #1, and 10.3.1.24 is twenty fourth node of the cluster #3.
- Mask of IP-address 255.0.0.0, i.e. all network infrastructure is fully attainable from any point, here differentiating of different clusters is executed by VLANs. The nodes of different clusters are mutually invisible as a result, but the systems of storehouse will be accessible even in case if the functions of storehouse and managing cluster node are laid on one device. The switch of networks of cluster management has the fixed IP-address: 10.N.M.250, where N is number of cluster, M is the number of switch.
- The frontend (managind node of cluster) has the fixed IP-address: 10.N.M.254, where N is number of cluster, M is the number of switch.
- The subnet 10.0.0.0/16 is given for services, so 10.0.0.254 is an access server address, (10.0.0.11 – 10.0.0.15) are devices of UPS and etc

File service. As a rule, parallel tasks are focused on the computings connected to huge files of the initial, intermediate or final data. *So, the analysis of results of nuclear researches with terabyte size of the initial data, and a tasks in a package of quantum chemistry Gamess* [8] *can use hundreds creates time files in the size in some gigabytes on process with usual read - write the intermediate results by small, fine packages.* Therefore the extremely important problem is to give to nodes high-speed access to storehouse systems of the huge sizes.

For increase in throughput of system of a storehouse it is used PORT TRUNKING - aggregation of 2-4 network interfaces in one for increasing the common new throughput though a linear gain is not possible.

As the distributed file system of cluster system SCIT still recently it was used NFS. As cluster nodes have no own disks each node during initial loading mounts root file system by NFS. Besides by NFS operating system sections with the working data of tasks were mounted also. Choice of NFS has been caused by the several reasons is a standard network file system, NFS is present in any UNIX-system, NFS is very easily adjusted and configured.

Operating experience NFS within one year as the basic file system has shown, that NFS is an excellent choice only for small (on 4-8 nodes) clusters, for clusters a level the SCIT, on 16-32 nodes, NFS can be a quite good choice under condition of use for the account of tasks with a small amount of operations of input-output with disk files. However NFS becomes a unacceptable choice at use for the task execution with an intensive input - output. Therefore file service has been modified, and the primary goals of modification were:

- a choice of the optimal distributed file system with an opportunity of scaling as on volume about an opportunity to node existing storehouse various clusters in one common data storage and on the maximal throughput;
- transition from use NFS on partial or full use of the chosen file system.

The following candidates for a role of the distributed file system were examined:

GFS manufactures RedHat (earlier SISTINA.COM) [13], for today last version 6.1. GFS uses as the distributed storehouse mounted simultaneously in all units GNBD (global network block device) atop of which GFS works actually with the manager of blocking.

GFS has many advantages - free-of-charge decisions very much, development by the largest manufacturer RedHat Linux, work «it is direct from a box» at use RedHat Enterprise Linux 4 and is higher or Fedora Core 4 and is higher, quite good scalability on volume, ease in installation and configuration.

During too time there are also lacks - bad scalability on the general throughput, and it means, that for escalating capacity it is necessary to use expensive hardware decisions such as FiberChannel as all nodes are shared with one block device refusal of one of nodes can lead to to some damages of file system.

GFS is a quite good choice for the finished decision when it is not planned to increase computing cluster capacity and volume of system of a storehouse, i.e. delivery on a turn-key basis. For use in our case supposing the further escalating of capacity on computing resources and volumes of disk space, approaches a little.

OCFS2 manufactures ORACLE [14], the successor with open codes OCFS. While the stable version of system still is not present, but it already is in kernel Linux, is supported by various distribution kits Linux. Actually represents distributed on all cluster nodes a RAID5-file that gives both high speed of read - write, and some fault tolerance of all file. However, if some units have broken down, the file can collapse down to loss of all data, i.e. the system demands a highly reliable disk subsystem on each cluster node, that very strongly increases a total price of the decision. OCFS2 it is optimum for processing the big databases for what actually and it was created.

Lustre manufactures CLUSTERFS.COM [1], the commercial, free-of-charge version, leaves with some backlog, is maximum for one year. It is very hard in installation, but it is very simple in configuration. It is perfectly scaled both on volume, and on throughput.

The system uses a set of patches to a kernel and consequently there can be problems of its construction, is especial in case of use of a various sort of the non-standard equipment. Problems basically are connected to a binding of interconnect drivers and driver **Lustre** to determined and not always to the same versions of kernel Linux. especially big such discrepancies arise at simultaneous use of architecture IA64, proprietary drivers SCI from firm SCALI and file system **Lustre**.

Thus it is very simple in configuration, it is enough to tell, that after big amount of works on construction **Lustre** and starting adjustment of cluster startup of node demands literally some minutes.

From the point of view of system **Lustre** looks as usual local file system with all pluses as aggressive caching *inodes* and *dentry*. Realization of full compatibility with POSIX is expected at third quarter of 2006 (the call *flock/lockf* doesn't realized now). Escalating of volume and throughput is made by simple addition in system of one or several nodes with disks (OSS). As each file can "be stripped" on several OSS and thus access to it made be parallel throughput grows practically in an arithmetic progression, i.e. the more at us is established OSS, the above speed of read-write. Thus very high degree of recycling of devices of an input-output so File I/O exceeds 90 % raw bandwidth disks is reached{achieved}, and single GigE end-to-end throughput reaches 118 MB/s at a physical maximum of the interface 125 MB/s. Additional plus is that as the network interface in **Lustre** any interface supporting report IP can practically act, and in some cases and more low level protocol (for example Infiniband).

High enough requirements on reliability are showed to storehouse, that is clear, as now integrity of file system depends on serviceability of all components entirely. And, though at what refusal or OSS-node cluster nodes can continue work if their data have not been located on the damaged node, but parts of the data all the same can be lost (in following versions **Lustre**, except for a mode of storage of files stripe or RAID0, modes RAID1 for small and RAID5 for the big files will be realized, therefore the probability of full loss of the data will be sharply reduced). Therefore use on OSS-units of RAID-files with redundancy is not the recommendation but the requirement.

Requirements to performance of OSS-nodes are very low. As the node is occupied with one task execution - maintenance of an input-output, any modern processor fulfils this task with success at low final cost. Moreover, idle computing powers are well enough to realize the programmed RAID-array function, i.e. they are saved money for expensive controllers (modern RAID-controllers lose to program RAID-arrays in speed for the banal reason - controllers uses weak enough CPU, actually for last years 5 RAID-controllers have found support RAID6,10,50,60, trunks PCI-X and PCI-E, but their computing capacities have remained at the same level of five years' prescription and in competition of frequencies wins more high-speed processor).

As there is some blank in **Lustre** performance – it's a search of files in the catalogue, 5000 op/s is very low figure and in some cases results in falling productivity (for example one of programs of the user created in the working catalogue about hundred thousand files and degradation of speed was appreciable). However this feature easily manages accommodation of working files not in one but in tens or hundreds catalogues.

After the analysis existing parallel (as above-named, and some other) file systems and an estimation of our technical opportunities we were defined that our specifications quite corresponded to requirements to **Lustre** and this file system has been chosen as the major candidate for a role of the distributed file system for ours supercomputers.

Early 2006 clusters SCIT have been transferred to use of distributed file system **Lustre**. It has allowed to unit all storehouses in one common file system in volume 1.7 Tbyte physically general file system settles down on three servers of the data (OSS) with four disk files (OSD) and one server of the metadata (MDS). As at configuration **Lustre** we have specified to distribute a file on all OSD (actually it is classical RAID-0 in application to a file) thus we could distribute loading on a file input - output simultaneously on all servers.

Results of testing of two file systems, **Lustre** and NFS, on a file in the size in 8 Gbyte (in testing are measured: throughput - Kb / c, use of the processor, frequency of search) are resulted in table 2.

Table 2.

Operation	Sequential Output						Sequential Input					
	Per Char		Block		Rewrite		Per Char		Block		Random seek	
	Kbps	%CPU	Kbps	%CPU	Kbps	%CPU	Kbps	%CPU	Kbps	%CPU	k/sec	%CPU
NFS	**26665**	**88.1**	**27907**	**6.3**	**3134**	**95.2**	**29215**	**91.1**	**84975**	**15.3**	**460.7**	**2.8**
Lustre	**27791**	**99.3**	**69991**	**41.3**	**39668**	**58.0**	**28066**	**98.9**	**98254**	**86.1**	**121.6**	**17.3**

Further it is possible to increase volume and throughput of file storehouse by simple connection to the switchboard of additional servers with disk files and small reconfiguration of the system.

3. The Selection of Architecture Features to a Supercomputer Project

What characteristics may be selected for the new supercomputer project. Proceeding the cluster tasks from mentioned early specific properties, it is possible to formulate the commons requirements to the node of cluster for the effective decision of parallel tasks:

- productivity of node linearly depends on power of processor, and productivity of processor from frequency descriptions of the used bus of main memory and amount of main memory accessible in a node (to some reasonable limit);
- interprocessor data exchange always faster than a interconnect exchange, i.e. preferably to use multiprocessor nodes (with 2–4 processors) and multicore processors;
- productivity of node depends on as used interconnect, two features are here important is latency, i.e. delay arising up at the transmission of minimum package between nodes, and maximal carrying capacity;
- productivity of node depends on intensity of operations of input-output with the devices of storehouse.

Pipeline and systems calls. As a rule, parallel tasks executed at computing node are not used with cyclic algorithms, therefore classic architecture with a short pipeline, used in the processors AMD, much preferably architectures P4 processors INTEL. Every reference to the data of neighbouring process is accompanied by a few transitions in the kernel mode of processor. Price of this transition on the processors of AMD 120-240 times, on the processors of architecture P4 1100-1300 times. However with appearance the recently represented architecture of Intel Conroe and actually by returning of Intel to architecture of P-III and short pipeline, in the second half of 2006 year and first half of 2007 year, i.e. down to the appearance of AMD K8L architecture, placing of forces will be completely other.

HyperThreading. Due to idle time of one of pipelines in incorrectly predicted transition or simply impossibility of parallel execution of instruction on architecture P4 there is possibility of the use of standing resources as a virtual processor (HyperThreading), but in parallel tasks it results only in falling of productivity. The reason is simple – the data exchange between nodes aligns productivity of all processes on speed the slowest and, as on a virtual processor is no more than 40% real processor, general productivity falls in 2–3 times, i.e. this possibility for clusters is practically unavailing.

64 bits versus 32 bits. For today all modern processors either support 64–bits expansions (AMD64, EM64T) or are pure 64-bits processors. Unfortunately, now a prize from the use of word length in 64 bits the programs requiring for calculations with such arithmetic collect in 64 bits receive only, and that not always, the other only lose. The reasons for this are few (due to size of data and address megascopic twice):

- It is required to increase twice processor cache, otherwise there is falling of productivity at the frequent «washing» of a cache.
- At that width of bus of memory plenty of addresses is required twice to main memory, that gives falling to productivity.
- It is required increases twice the size of node main memory.

Power consumption of the processor. The selected power of processor can non-obvious influence on general productivity of all system – at the overheat of one of processors to it the automatic lowering of frequency will be applied, that will result in the common falling of productivity of all system on the whole.

Main memory. Clusters have 1-2 GByte on every processor core of node:

- More than 2 Gbyte on a processor core expediently either at the use of clean 64-bit architecture or after clarification of specificity of the basic applied tasks of cluster, otherwise memory will be essential to idle, however there are situations, when than the more maximal capacity of main memory of node is anymore, so much the better for a task (main memory modelling of SMP-architecture);
- Frequency on which main memory works should be maximal of all supported the chosen processor architecture;
- The used chipset should be able to support the necessary amount of memory.

Interconnect. As the cost of interconnect lies in a very wide range from the zero of to a few thousand dollars on a site, the choice of interconnect is determined by the basic purpose of the cluster system:

Latency of interconnect is one of major indexes, influencing on the real productivity of the cluster system, this time expended by the operating system and device on the transmission of single package to other cluster node. Because a data exchange between nodes takes place by such transmissions, the latency can be described as the time lost by a process. For tasks with a large data exchange between nodes the large latency can give the catastrophic falling of productivity, at the same time for tasks with a small data exchange between nodes the small latency will give nothing in the plan of winning of productivity, but will result in the enormous increase of budget of project.

The throughput of interconnect practically does not tell on general productivity of the system. There are some minimum scopes, but throughput of any device used today as interconnect, is higher than these scopes, i.e. as yet we did not meet a task to the necessity of which in the throughput of interconnect there would be compared with the necessities of ping-pong benchmark.

Conclusion

Presently productivity of existent cluster systems SCIT suffices only for the simultaneous calculation of a few tasks, therefore for satisfaction of present queries from the side of Institutes of NAN of Ukraine, that the decision of large tasks left off to be a bottleneck, it is necessary to heave up productivity of supercomputer systems on an order (to a few teraflops).

Evaluation the characteristics of the cluster system which would on nearest a few years to answer the vital queries of the mentioned directions in physics, biology, technique and etc, are the following:

1. Quantity of dual-core processors with frequency at (2,2 – 2,8) GHz – 300.
2. Main memory (total) — 1,0 TByte.
3. Bulk storage on local HDD — 2.5 TByte.
4. Global storehouse system — 10 TByte.
5. Highest possible productivity — (4,5 –5.5) Tflops.

Bibliography

1. www.lustre.org/
2. http://www.netlib.org/atlas/
3. http://www.netlib.org/blacs/
4. http://www.netlib.org/scalapack/
5. http://www.intel.com/cd/software/products/asmo-na/eng/perflib/mkl/index.htm
6. http://www.gromacs.org/
7. http://www.wien2k.at/
8. www.msg.ameslab.gov/GAMESS/GAMESS.html
9. http://www.scali.com/
10. http://www.open-mpi.org/
11. www.t-platforms.ru/english/about/dnd.html
12. www.intel.com/design/servers/ipmi/spec.htm
13. www.redhat.com/software/rha/gfs/
14. oss.oracle.com/projects/ocfs2/

Authors' Information

Ahdrey L. Golovinskiy *– Institute of Cybernetics NAS Ukraine; Prospekt Academika Glushkova,40, Kiev, 03680 MCP, Ukraine; e-mail: tikus@ukr.net*

Sergey G. Ryabchun *– nstitute of Cybernetics NAS Ukraine; Prospekt Academika Glushkova,40, Kiev, 03680 MCP, Ukraine; e-mail: sr@emt.com.ua*

Anatoliy A. Yakuba *– Institute of Cybernetics NAS Ukraine; Prospekt Academika Glushkova,40, Kiev, 03680 MCP, Ukraine; e-mail: ayacuba@gmail.com*

ТЕХНОЛОГИЯ ПРОГРАММИРОВАНИЯ КЛАСТЕРНОГО КОМПЬЮТЕРА С ПОМОЩЬЮ УДАЛЕННОГО ТЕРМИНАЛА С ОПЕРАЦИОННОЙ СИСТЕМОЙ WINDOWS

Дмитрий Черемисинов, Людмила Черемисинова

Резюме*. Рассматривается проблема подготовки программы к выполнению на мультипроцессорной системе кластерного типа. При разработке программ для кластерного компьютера применяется технология, основанная на использовании удаленного терминала. Рассматривается ситуация, когда таким удаленным терминалом является компьютер с операционной системой Windows. Предлагается набор инструментальных средств, позволяющий выполнять задачи редактирования текста, компиляции программы и запуска программы на кластере. Достоинством предлагаемого способа подготовки программы к выполнению является возможность максимального использования опыта работы программиста в Windows.*

Ключевые слова*: параллельные вычисления, кластерный компьютер, технология программирования.*

Введение

Под кластером понимают совокупность процессоров, объединенных компьютерной сетью и предназначенных для решения одной задачи, как правило, большой вычислительной сложности. На абсолютном большинстве кластеров в качестве операционной системы используется Linux.

Кластер – довольно сложный объект и для работы на нем нужно иметь знания по сетям, устройству компьютеров, операционным системам, специальному программному обеспечению. При разработке программы для кластера хотелось бы изучать как можно меньше технических деталей, связанных с управлением операционной системой при редактировании программы, компиляции, запуске на выполнение, и сосредоточиться на разработке параллельного алгоритма. Этого можно достичь, если использовать компьютер с Windows как удаленный терминал кластерного компьютера. Однако минимальные сведения об устройстве операционных систем на основе UNIX все же необходимы. Цель этой статьи состоит в том, чтобы описать самые основные действия, которые приходится выполнять в ходе процесса разработки программы, и особенности процесса подготовки программы с точки зрения программиста, работающего в среде ОС Windows.

Доступ к кластерному компьютеру

Чтобы иметь возможность работать с кластером, нужно быть зарегистрированным пользователем головной машины кластера. Чтобы это сделать самостоятельно, нужно иметь доступ к идентификационной информации суперпользователя кластера. Обычно регистрацию нового пользователя выполняет системный администратор кластера. В результате регистрации пользователю выделяется место для размещения своих данных на головной машине кластера и сообщаются логин (алфавитно-цифровая строка символов, используемая для идентификации пользователя в операционной системе) и пароль для доступа.

Компьютер с ОС Windows должен быть подключен к той же сети, что и головная машина кластера. Для интерактивной работы с кластером и обмена файлами требуется использовать протокол с шифрованием информации – SSH (Secure SHell). Если использовать протокол telnet, в котором пароли и команды передаются в открытом виде, пароль пользователя может быть перехвачен при передаче по сети и использован для несанкционированного доступа (посторонних) к кластеру. Для подключения пользователей Windows потребуется терминальная программа доступа к кластеру, использующая протокол SSH. Ее необходимо установить и настроить. Для этого удобно использовать свободно доступную программу PuTTY, которую можно найти в Интернет по следующему адресу: *ftp://linux4u.jinr.ru/pub/win9x/crypt/putty-0.53b/x86/putty.exe*

Программа PuTTY представляет собой сервис терминала, т. е. дает возможность вводить информацию через клавиатуру и выводить ее на экран дисплея. Если тексты разрабатываемой программы уже находятся на кластере, PuTTY достаточно для дальнейшей работы. Если же они находятся на вашем компьютере, что более удобно, то необходима программа, обеспечивающая транспортировку данных с вашего компьютера на кластер и наоборот. В принципе для этого предназначена имеющаяся в Windows программа *ftp*. Однако использовать для транспортировки данных незащищенный протокол FTP, в котором данные передаются в открытом виде, нельзя по соображениям безопасности.

Для организации транспортировки данных с использованием защищенного канала передачи удобно использовать программу SecureFX. SecureFX – это клиентская программа протокола SHH для зашифрованной передачи файлов с широкими возможностями настройки конфигурации и протоколов передачи. SecureFX поддерживает режим докачки и восстановления связи в случае ее обрыва. Эта программа распространяется за плату. Программа SecureFX – это клиент сервиса транспорта данных, который может передавать данные как посредством традиционного протокола FTP, так и через шифрующий протокол SSH2.

Использование Midnight Commander

В стандартном состоянии для выполнения сеанса отладки программы для кластера тексты этой программы находятся в одном из каталогов головной машины кластера. Процедура подключения к кластеру после запуска программы PuTTY выполнена, и на ее экране, на клиентской машине с Windows, имеется приглашение для набора командной строки для командного интерпретатора головной машины кластера (рис. 1).

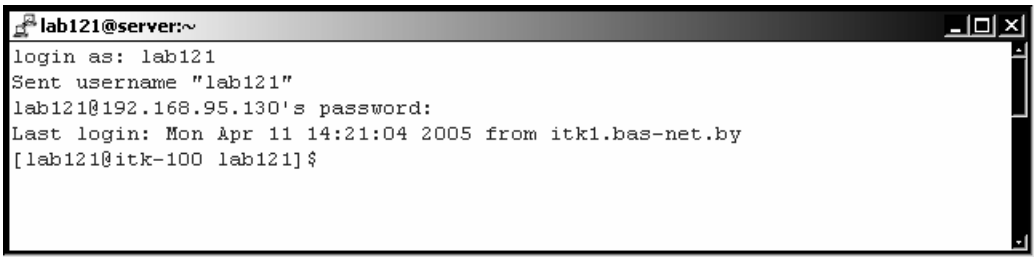


Рис.1. Образец содержимого экрана программы PuTTY после завершения процедуры подключения

О языке командной строки можно прочитать в любом руководстве по Linux. Обычно руководство по системам типа UNIX представляет собой толстую книгу, в которой описание команд командной строки и их параметров составляет основную часть. Для пользователя Windows удобнее для управления системой через консоль использовать программу mc (midnight commander). В Linux последних версий она входит в комплект поставки. Для ее запуска в приглашении командной строки набираем *mc*. После этого экран PuTTY принимает вид экрана известной оболочки «Нортон командер» (рис. 2). Большинство кнопок управления работают также, как в Нортоне [1] для DOS. Для тех, кто использует Нортон (Windows commander) или *FAR*, отличия в использовании *mc* связаны только с учетом особенностей файловой системы Linux. Теперь задачи поиска и просмотра файлов на кластере можно выполнять тем же способом, что и в ОС Windows.

В тех случаях, когда требуется немного подправить текст отлаживаемой программы, можно использовать встроенный редактор программы *mc* (функциональная клавиша F4) и редактировать текст прямо на кластере. В случае больших изменений удобнее редактировать тексты на своей машине, используя привычный редактор. Основные неудобства при редактировании на кластере связаны с невозможностью выполнения коррекций с помощью привычных операций copy-paste, трудностью переключения языка (русский/латинский) и т. п.

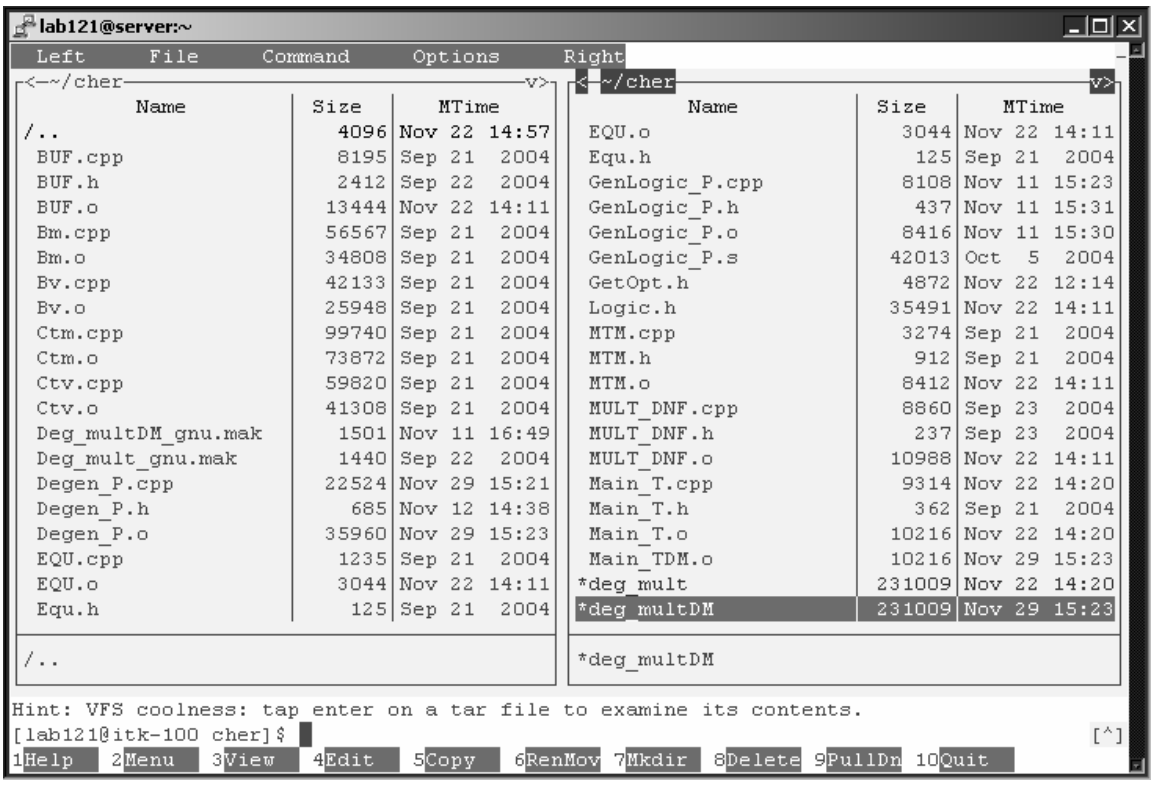


Рис. 2. Образец содержимого экрана программы PuTTY после запуска *mc*

Запуск программы на выполнение выполняется теми же способами, что и в «Нортоне». Нужно учитывать, что исполняемые файлы программ в двоичном виде в Linux не имеют расширения. Признаком того, что файл имеет права на выполнение, является звездочка перед именем файла (рис. 2). Для установки прав доступа используется команда *chmod*, которая выбирается через меню «команды» (функциональная клавиша F9 – как в Нортоне – для доступа к главному меню).

Некоторую проблему представляет собой аварийное прекращение работы запущенной программы по инициативе пользователя. В большинстве случаев прекращение работы вызывается сочетанием клавиш Ctrl-C. Перед этим нужно перейти в режим экрана командного интерпретатора, т. е. закрыть панели программы *mc*. Закрытие консоли на удаленной машине – выход из программы PuTTY – вообще говоря, программу не останавливает.

Режимы запуска программ

Для запуска программы можно просто в поле командной строки программы mc набрать ее имя, нужные параметры и нажать ввод. Пока так запущенная программа работает, пользователь ничего делать не может, должен ждать ее завершения. Кроме обычного режима запуска существует еще режим фонового запуска, который позволяет освободить консоль для выполнения других работ. Что запустить программу в фоновом режиме, надо после имени файла поставить символ «&». Посмотреть список фоновых программ, которые уже запущены, можно командой *jobs*. В полученном списке все команды будут пронумерованы, и если какую-то из них нужно перевести в обычный режим выполнения, то нужно выполнить команду, например, “fg 1” (сокращение от foreground). Для того чтобы перевести в фон запущенную программу можно использовать команду *bg* (сокращение от background). Но если программа уже работает, то команду *bg* ввести не получится. Но выход есть, сначала надо будет нажать сочетание клавиш Ctrl-Z (после чего программа приостановится), а потом уже воспользоваться командами *jobs* и *bg*. Режим запуска программ через поле командной строки *mc* зависит от настройки *mc* (обычно в фоновом режиме).

Подготовка к компиляции программы

В операционной системе Linux имеются среды для разработки программ на C++, которые предоставляют программисту удобства, аналогичные возможностям среды MSVC в Windows, однако работа в этих системах значительно отличается от работы в среде MSVC. Один из крупных недостатков использования для доступа к кластеру компьютера с Windows состоит в том, что в этом случае приходиться ограничиваться теми ресурсами Linux, которые доступны через консоль. Поэтому через удаленный терминал программы PuTTY в принципе нельзя использовать программы Linux с графическим интерфейсом. С другой стороны, использование среды для разработки программ из Linux означает девальвацию навыков, приобретенных в MSVC.

Возможна стратегия разработки на основе концепции переносимого кода, которая обеспечивает независимость разрабатываемой программы от среды разработки. Довольно удобен вариант этой стратегии, позволяющий в какой-то мере обеспечить применение знаний среды MSVC, когда последовательный прототип параллельной программы разрабатывается в среде MSVC с учетом требования переносимости кода. Переносимость кода означает использование архитектуры консольной программы в MSVC. Этот код преобразуется в параллельную программу, которая предназначена для выполнения на кластере. Для этого тексты программы должны быть откомпилированы на Linux.

На головной машине кластера всегда имеется компилятор C++, который управляется средствами командной строки. Без такого компилятора не может существовать ни одна система UNIX. Дело в том, что между разными машинами, даже с одним и тем же типом ОС UNIX, не существует совместимости программ на уровне двоичных кодов. Для машин с Linux существует совместимость программ на уровне двоичных кодов при одинаковой версии ядра операционной системы. Таким образом, установка любой программы (за исключением программ, которые выполняются интерпретаторами) в UNIX обычно требует компиляции ее исходного кода.

Задание на компиляцию программы с использованием компилятора C++ Linux представляет собой управляющий файл для программы *make*. Таким образом, для компиляции программы кроме ее текстов на C++ нужно иметь файл задания для *make*. Среда MSVC позволяет экспортировать файл проекта в

формате *make*. Однако файл, экспортированный из MSVC, мало пригоден для управления компиляцией на Linux, так как требует такой ручной переделки, что проще его написать заново с нуля.

Можно компилировать без задания для программы *make*, вызывая компилятор C++ непосредственно из командной строки, но в этом случае текст программы должен выглядеть как один файл, а все остальные части текста должны подключаться явно с помощью директив *#include*. Это очень неудобно по многим причинам. Одна из существенных причин использования формата *make* состоит в том, он предоставляет возможность ускорить компиляцию программы, состоящей из нескольких файлов. Программа *make* позволяет использовать объектные файлы и выполняет перекомпиляцию только тех файлов с исходным текстом, которые были изменены. Кроме того, при использовании *make* шаги сборки программы можно не знать.

Формат файла для *make* довольно сложен, и построение задания требует значительной работы: нужно помнить, как одни файлы зависят от других, и указать все файлы, которые потребуются при компиляции вашей программы; поэтому часто приходится указывать в управляющем файле для *make* довольно много данных. Если при компиляции требуется использовать несколько различных инструментов (например, редактор связей кроме компилятора), то нужно предусмотреть управление каждым из них. Существуют программы, позволяющие сгенерировать управляющий файл автоматически. С этой целью удобно использовать программу *genmake*, которую можно найти в Интернет по следующему адресу: *http://www.robertnz.net/genmake.htm*

Проект, для которого генерируется управляющий файл для *make*, должен быть подготовлен. Для этого в каждый *h*-файл проекта нужно включить специальные директивы в формате комментария. В этих директивах указываются имена *cpp*-файлов, в которых содержаться определения объектов, декларированных в соответствующем *h*-файле. Для программы *genmake* указывается название *cpp*-файла, содержащего функцию *main* программы. Программа *genmake* просматривает этот файл для нахождения директив *#include* для включения *h*-файлов. Найденные *h*-файлы просматриваются с целью поиска директив *genmake*, в которых указаны cpp-файлы, содержащие определения объектов, декларированных в соответствующем *h*-файле. Таким образом, разыскиваются все требуемые файлы и формируются зависимости между ними. К сожалению, сгенерированный *genmake* файл требует небольшой ручной доработки.

Проблема русского языка

Принципиальное отличие ситуации с русским языком в Linux от других языков связано с тем, что на сегодняшний день существует несколько кодировок для русского языка – Windows 1251, KOI-8r, ISO8859-5 и др. По умолчанию кодовой страницей русского языка для Linux является KOI-8r, а в Windows кодовая страница русского языка – это Windows 1251. Различие в стандартах кодировки ведет к тому, что тексты программ, подготовленные для компиляции в Windows, будут неправильно выглядеть при просмотре на Linux. Но проблема не только в просмотре текста программы, через консоль очень трудно выполнить переключение раскладки клавиатуры на русский язык при редактировании. Это главная причина, по которой редактирование текстов программ удобнее выполнять на удаленном компьютере пользователя. С точки зрения удобства редактирования текста программы удобнее всего в качестве редактора использовать среду MSVC. Ее редактор особенно незаменим, когда необходимо анализировать структуру вложенности скобок. С другой стороны, для правильной работы с русским языком редактор должен обеспечивать возможность преобразования кодировки Windows 1251 в KOI-8r и наоборот.

С точки зрения возможностей просмотра и редактирования текстов программ довольно удобен текстовый редактор Aditor, который можно найти в Интернет по адресу http://nrd.pnpi.spb.ru/UseSoft/tools/Aditor/aditor.htm. Основное его достоинство – полная поддержка если не всех, то почти всех кодировок русского алфавита (KOI, Win, DOS, ISO, Mac). “Полная” поддержка означает то, что можно просматривать и редактировать файлы в любой из этих кодировок, при открытии файла его кодировка определяется автоматически и почти всегда безошибочно (а при ошибке можно и поправить), можно переводить файлы из одной кодировки в другую по желанию пользователя, автоматически перекодируется Clipboard и т. д. Кроме того, в этом редакторе выделяются разными цветами синтаксические конструкции C++.

Технология работы

Основой организации работы с программой для кластера является каталог на компьютере удаленного терминала с текстами программы. Эти тексты отличаются от текстов исходного прототипа для Windows не только параллельностью, но также и тем, что они перекодированы в KOI-8r (с помощью редактора Aditor) и в *h*-файлы включены директивы *genmake*. Первым этапом разработки программы для кластерного компьютера является создание и заполнение такого каталога. Затем с помощью программы *genmake* нужно построить *make*-файл и исправить его вручную. Далее этот каталог с данными дублируется в каталог на головной машине кластера. Для этого данные из каталога на машине удаленного терминала транспортируются в каталог на головной машине кластере (с помощью программы SecureFX). На этом подготовительная работа для компиляции и запуска программы заканчивается.

Далее каталог на кластере служит зеркалом каталога на компьютере разработчика. Файлы, с которыми идет работа, удобно держать открытыми в программе Aditor. Для компиляции и выполнения программы нужно запустить программу PuTTY и в ней программу Midnight Commander. Если при компиляции возникли ошибки, номера строк программы и диагностику ошибок можно посмотреть под панелями программы *mc*. Используя открытые в редакторе Aditor файлы и эти номера, можно найти соответствующие строки кода программы и устранить ошибки. После редактирования кода нужно сохранить исправления (не закрывая редактор Aditor) и с помощью программы SecureFX транспортировать исправленные файлы в каталог на кластере. После обновления каталога на кластере (содержимое каталога на кластере должно быть идентично содержимому каталога на компьютере пользователя) нужно повторить компиляцию.

Если компиляция прошла успешно, нужно проследить, чтобы двоичный файл программы появился на панели *mc* и был отмечен звездочкой. Теперь можно запускать этот файл на выполнение. Результаты работы можно посмотреть под панелями *mc*.

Заключение

В работе содержатся рекомендации по созданию комплекса программных средств, обеспечивающих выполнение типовых операций по подготовке к выполнению и запуску программ для кластера, головная машина которого представляет собой UNIX сервер, с помощью удаленного компьютера с Windows. Предлагаемая конфигурация программных средств не является единственной из возможных. Однако использование предлагаемой технологии позволяет программисту, имеющему навыки работы с инструментами для Windows, использовать эти знания при разработке программ для кластера. Это, конечно, дается ценой отказа от использования некоторых средств, предоставляемых рабочей системой на основе Linux. Основной недостаток предлагаемого подхода состоит в невозможности использования инструментов разработки программ Linux с графическим интерфейсом. Рекомендации для пользователей кластера, работающих на рабочих компьютерах с Linux, приведены в работе [2].

Практическое применение предлагаемой конфигурации для разработки нескольких параллельных программ [3, 4], последовательные прототипы которых были построены в среде MSVC, позволило добиться того, что перенос последовательного алгоритма в Linux не требовал умения работать в Linux и основные трудозатраты состояли в разработке и отладке параллельного алгоритма, а не в овладении комплексом инструментальных средств. Более подробно описываемая технология разработки программ для кластерных компьютеров описана в статье [5].

Библиография

1. Фигурнов В.Э. “IBM PC для пользователя. От начинающего до опытного”, М.: ИНФРА-М, 640 с., 2002.
2. Галактионов В.В., Голоскокова Т.М., Громова Н.И. и др. “Руководство для пользователей LINUX кластера ЛИТ ОИЯИ”, Дубна, 2004.
3. Торопов Н.Р. “Параллельная проверка ДНФ на тавтологию”, Информатика, № 2 (6), с. 35–42, 2005.
4. Торопов Н.Р. “Параллельные логико-комбинаторные вычисления в среде MPI”, Информатика, № 3 (7), с. 82–90, 2005.
5. Черемисинов Д.И. “Работа с кластерным компьютером из Windows”, Информатика, № 3 (7), с. 91–99, 2005.

Информация об авторах

Дмитрий Иванович Черемисинов *– Объединенный институт проблем информатики Национальной академии наук Беларуси, ул. Сурганова, 6, Минск, 220012, Беларусь, e-mail: cher@newman.bas-net.by*

Людмила Дмитриевна Черемисинова *– Объединенный институт проблем информатики Национальной академии наук Беларуси, ул. Сурганова, 6, Минск, 220012, Беларусь, e-mail: cld@newman.bas-net.by*

ИНСТРУМЕНТАЛЬНЫЕ СРЕДСТВА УДАЛЁННОГО ПАРАЛЛЕЛЬНОГО МОДЕЛИРОВАНИЯ

Александр Миков, Елена Замятина, Антон Фирсов

Реферат*: В докладе представлено описание распределённой системы имитации Triad.Net и инструментальных средств, позволяющих географически удалённым пользователям через Интернет осуществлять совместную работу над имитационными моделями и наблюдать за ходом имитационного эксперимента.*

Ключевые слова*: Распределённое/параллельное имитационное моделирование, удалённый доступ, портал.*

ACM Classification Keywords*: I.6 Simulation And Modeling: I.6.7 Simulation Support Systems – Environments; I.6.8 Types of Simulation – Distributed*

Введение

Имитационное моделирование и в настоящее время остаётся общепризнанным средством для исследований в разных областях знаний. Поскольку сложность решаемых с помощью имитационного моделирования задач растёт, возрастает и необходимость в создании распределенных и параллельных систем имитации, которые используют вычислительные ресурсы нескольких процессоров или нескольких компьютеров [1,2,3].

По замечанию известных исследователей в области распределённого имитационного моделирования R.Fujimoto и K.Perumalla [4], существуют три важных фактора, которые способствуют продвижению параллельных информационных технологий в области имитационного моделирования. К ним можно отнести:

- наличие графического интерфейса;
- наличия удалённых средств доступа к параллельным вычислительным ресурсам;
- наличие программных средств, которые дают возможность пользователям, находящимся на удалённом расстоянии друг от друга через Internet вести совместную работу с одной и той же имитационной моделью.

В докладе представлены языковые и программные средства имитационной системы Triad.Net, позволяющие исследователям удаленно и совместно с другими исследователями через Internet взаимодействовать с имитационной моделью, а также, и наблюдать за поведением модели во время имитационного эксперимента.

Описание имитационной модели в Triad.Net

Система автоматизированного проектирования Triad была разработана на кафедре математического обеспечения вычислительных систем Пермского государственного университета [5,6,7] и предназначалась для проектирования встроенных вычислительных систем.

Описание имитационной модели в Triad состоит из трех слоёв: слоя структур (STR), слоя рутин (ROUT) и слоя сообщений (MES). Таким образом, модель в системе Triad можно определить как M={STR,ROUT,MES}.

Слой структур представляет собой совокупность объектов, взаимодействующих друг с другом посредством посылки сообщений. Каждый объект имеет полюса (входные P_{in} и выходные P_{out}), которые служат соответственно для приёма и передачи сообщений. Слой структур можно представить графом. В качестве вершин графа следует рассматривать отдельные объекты. Дуги графа определяют связи между объектами.

Объекты действуют по определённому алгоритму поведения, который описывают с помощью рутины (rout). Рутина представляет собой последовательность событий e_i, планирующих друг друга ($e_i \in E$, $i=1 \div n$, E-множество событий, множество событий рутины является частично упорядоченным в модельном времени). Выполнение события сопровождается изменением состояния q_k объекта. Состояние объекта определяется значениями переменных var_j рутины ($var_j \in Var$, $j=1 \div m$, Var – множество переменных рутины). Таким образом, система имитации является событийно-ориентированной. Рутина так же, как и объект, имеет входные (Pr_{in} и выходные Pr_{out}) полюса. Входные полюса служат соответственно для приёма сообщений, выходные полюса – для их передачи. В множестве событий рутины выделено входное событие e_{in}. Все входные полюса рутины обрабатываются входным событием. Обработка выходных полюсов осуществляется остальными событиями рутины. Для передачи сообщения служит специальный оператор out (out <сообщение> through <имя полюса>). Совокупность рутин определяет слой рутин ROUT. Слой сообщений (MES) предназначен для описания сообщений сложной структуры.

Система Triad реализована таким образом, что пользователь может описать только один слой. Так, если возникает необходимость в исследовании структурных особенностей модели, то можно описать в модели только слой структур.

Система имитации Triad.Net является развитием системы Triad. Она представляют собой параллельную (распределённую) версию. В системе имитации Triad.Net реализованы распределённые алгоритмы синхронизации функционирующих во времени объектов (консервативный и оптимистический, математическая модель алгоритмов описана, в частности, в [13]). Кроме того, для системы Triad.Net характерно следующее [8]:

1. Входной язык описания моделей содержит переменные типа «модель». Над моделями определены операции. Операции определены как для моделей в целом, так и для каждого слоя. Например, в слое структур возможно выполнить добавление и удаление вершины, добавить или удалить ребро (дугу), полюсы, найти объединение или пересечение графов. Кроме того, вершине из слоя структур можно приписать (по определённым правилам) ту или иную рутину из слоя рутин, тем самым, изменив алгоритм её поведения, и т.д. Таким образом, имитационная модель может быть описана языковыми средствами, а может быть и построена в результате исполнения некоторого алгоритма преобразования модели. При выполнении операций над моделью перетрансляции модели не требуется.
2. Модель является иерархической, т.е. каждая вершина в слое структур может быть расшифрована подструктурой.
3. Подсистема анализа модели должна обеспечить получение информации по заранее сформулированному запросу, а не ограничивать пользователя строго регламентированным набором собираемых данных. Такой подход к сбору информации позволяет избежать избыточности собранной информации или того, что она окажется недостаточной. Во время моделирования пользователь имеет возможность изменить набор собираемых данных, при этом модель остается неизменной. Подсистема анализа должна располагать интеллектуальными

инструментальными средствами, которые выполняют анализ результатов имитационного эксперимента и вырабатывают рекомендации по проведению следующих экспериментов.

4. Алгоритм имитации должен быть эффективным и масштабируемым, т.е. объекты должны быть таким образом распределены по компьютерам, чтобы снизить потери, связанные с передачей сообщений по каналам связи от компьютера(процессора) к компьютеру(процессору) и учитывать загрузку компьютеров(процессоров).

5. Система должна быть объектно-ориентированной (должно поддерживаться наследование, переиспользование кода).

Графический редактор модели формирует xml-документ, включающий описание объектов, рутин, структуры сложных сообщений и информационных процедур. Информационные процедуры предназначены для сбора статистики и речь о них пойдёт далее. Кроме того, в xml-документе указывают физическое расположение объектов по компьютерам(процессорам) [17].

Использование языка xml придаёт системе имитации дополнительную гибкость (интероперабельность, переиспользование кода) [8]. Следует отметить, что в настоящее время большое количество систем имитации использует язык xml [10,11,12 и т.д.].

Triad.Net можно отнести к *«монолитными»* системам моделирования [1], т.к. система имеет собственную среду для выполнения имитационного эксперимента, а также инструментальные средства для анализа и представления результатов моделирования, а с другой стороны система является *компонентно-ориентированной*, т.к. каждый объект представлен отдельным компонентом.

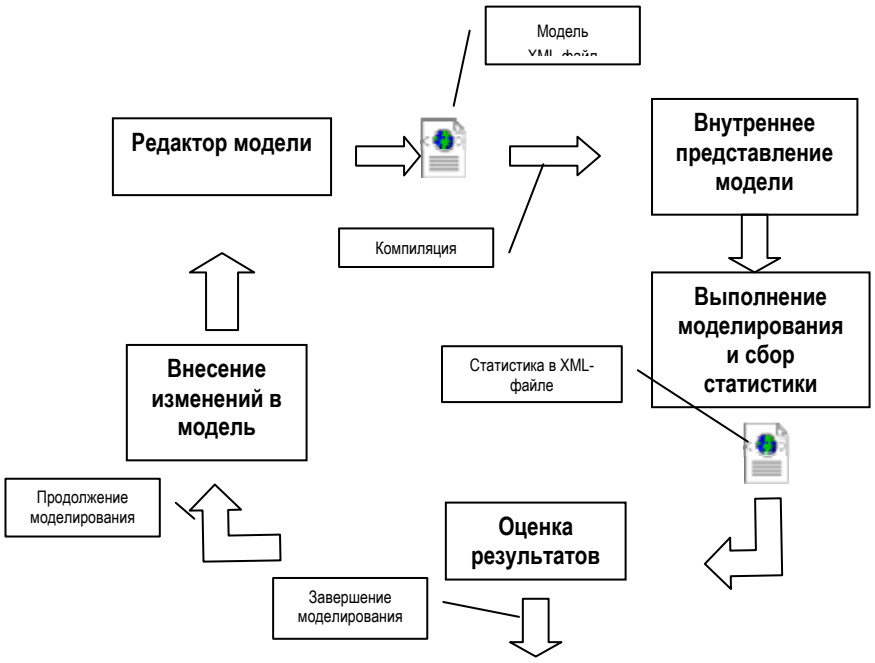
Информационные процедуры

Алгоритмом имитации назовём *объекты*, функционирующие по определённым сценариям, и синхронизирующий их *алгоритм*.

Для сбора, обработки и анализа имитационных моделей в системе Triad.Net существуют специальные объекты – информационные процедуры и условия моделирования. Информационные процедуры и условия моделирования реализуют алгоритм исследования.

Информационные процедуры ведут наблюдение только за теми элементами модели (событиями, переменными, входными и выходными полюсами), которые указаны пользователем. Если в какой-нибудь момент времени имитационного эксперимента пользователь решит, что следует установить наблюдение за другими элементами или выполнять иную обработку собираемой информации, он может сделать соответствующие указания, подключив к модели другой набор информационных процедур. Информационные процедуры являются единственным средством системы для одновременного доступа к элементам модели, принадлежащим разным объектам. Именно с помощью информационных процедур пользователь может осуществить взаимодействие (в том числе и удалённое) с объектами модели во время имитации.

Условия моделирования анализируют результат работы информационных процедур и определяют, выполнены ли условия завершения моделирования. Система имитации Triad.Net располагает языковыми средствами для описания алгоритмов работы информационных процедур. В описание информационных процедур входят: описание начальной части (выполняется до начала имитационного эксперимента), описание заключительной части (эта часть выполняется по окончании эксперимента), описания тела информационных процедур, настроечных параметров и параметров интерфейса. При изменении значения переменной, за которой ведётся наблюдение, при выполнении события, указанного пользователем, или после прихода (передачи) сообщения на входной полюс происходит подключение информационной процедуры (тела) к конкретному элементу модели (используются параметры интерфейса) и данные обрабатываются по заданному в информационной процедуре алгоритму. Воспользовавшись средствами графического редактора, пользователь получает описание модели в документе xml-формата. Результаты моделирования, полученные по окончании имитационного прогона, также представлены xml-документом (рис.1).



Процесс моделирования в Triad.NET

Таким образом, результаты моделирования могут быть обработаны как *встроенными* средствами Triad.Net, так и *стандартными* средствами обработки xml-документов.

В имитационной системе Triad.Net информационные процедуры и условия моделирования реализуют алгоритм исследования. Алгоритм исследования отделён от алгоритма имитации и удовлетворяет требованиям, указанным выше (требование 3). Алгоритм исследования и алгоритм имитации также могут выполняться параллельно (на разных компьютерах или процессорах).

Визуализация

В обоих случаях, как при последовательном, так и параллельном моделировании, визуализация является незаменимым средством для пользователя. В случае параллельного моделирования дополнительной мотивацией для разработки программных средств, визуализирующих процесс имитации, является отладка имитационной модели [4]. С помощью графической визуализации решение этой задачи, а также задачи, связанной с повышением эффективности алгоритма имитации (требование 4), становится менее затруднённым.

Большинство последовательных коммерческих систем имитации оснащено дружественным графическим интерфейсом. Этого нельзя сказать о параллельных системах имитации. Тем не менее, работы в этой области ведутся [4,13,14]. Графический интерфейс, реализующий удалённый доступ через Internet, необходим для исследователей, ведущих совместную работу и находящихся на удалённом расстоянии друг от друга.

Интересной разработкой такого рода является проект Jane [4]. Проект Jane имеет клиент-серверную архитектуру и является графическим интерфейсом, реализующим взаимодействие пользователя с моделью и удалённый доступ через Internet. Сервер написан на C, а клиент - на Java (архитектура системы позволяет поддерживать клиентов на Visual Basic). При разработке проекта авторы выдвинули требование относительной независимости графического интерфейса и параллельного/распределённого алгоритма имитации. По этой причине, программное обеспечение проекта Jane является «нейтральным», не зависящим от системы имитации и имитационной модели (используется как для визуализации TeD, так и RTI [1]).

В Triad.Net для визуализации хода имитационного эксперимента и визуализации результатов моделирования используют информационные процедуры и условия моделирования. Информационные

процедуры реализуют удалённый доступ к распределённой имитационной модели и решают проблемы взаимодействия удалённых пользователей с моделью и их совместной работы над проектом.

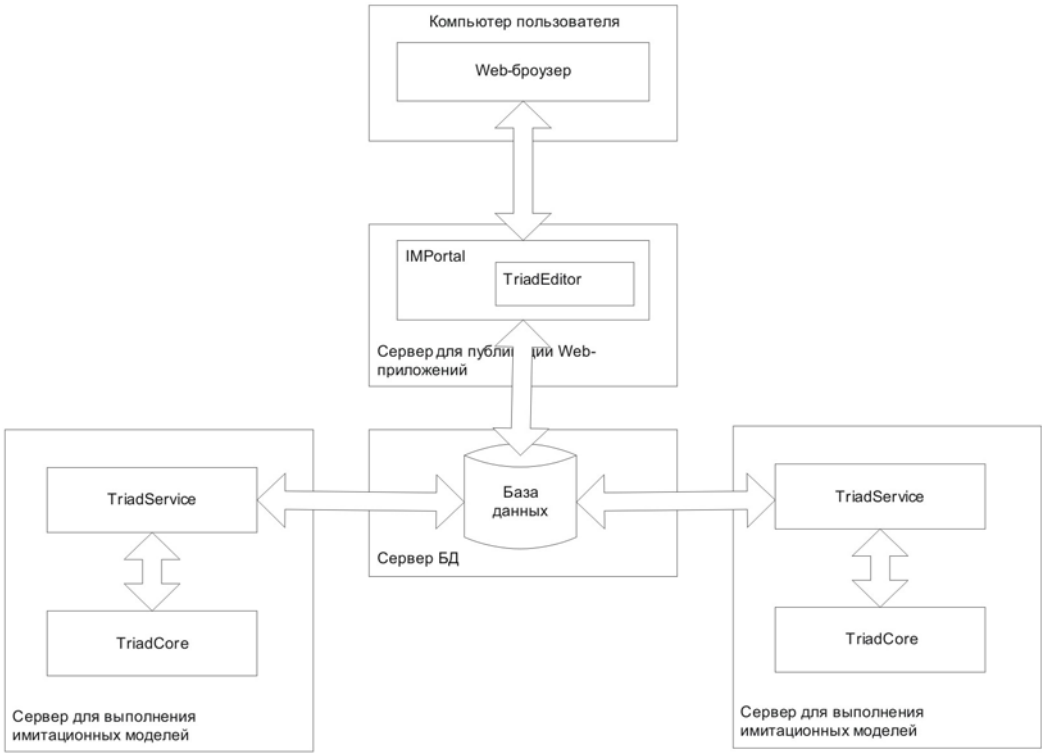
Вопросы реализации Triad.Net

Система имитации Triad.Net реализована с использованием технологии .Net. Для описания объектов модели и информационных процедур используют C#. Использование технологии .Net, а также описание модели в виде xml-документов и хранение результатов моделирования в xml-формате дают возможность реализовать гибкую компонентно-ориентированную систему, а также достичь определённого уровня переиспользования кода [15,16]. Решения, которые были приняты в системы Triad.Net, позволяют реализовать графический интерфейс, используя при этом информационные процедуры, визуализировать и анимировать результаты моделирования (в настоящее время ведутся работы по применению программного обеспечения на языке VRML для визуализации хода эксперимента, используя накопленные во время эксперимента данные), осуществить удалённый доступ к модели [9,17].

Для реализации удалённого доступа в Triad.Net были разработаны соответствующие инструментальные средства, в результате система Triad.Net включает следующие модули:

1. IMPortal – Интернет-портал, основанный на метаданных, который может использоваться отдельно от всего остального проекта как самостоятельное приложение. Изменяя метаданные, можно настроить структуру данных и наполнить портал содержимым из любой другой области, в том числе и не связанной с имитационным моделированием. В этом смысле IMPortal является *каркасом* Интернет-портала, а каким содержимым он будет наполнен, зависит от администратора портала.
2. TriadCore – представляет собой библиотеку типов, содержащую описание базовых структур, используемых в имитационной модели, таких как:
   - BaseObjectClass – класс, описывающий структуру базового класса объектов, – от него наследуются все создаваемые в системе классы объектов.
   - BaseObject – класс, описывающий структуру базового объекта, - от него наследуются все создаваемые в систему объекты.
   - BaseSpy – класс, описывающий структуру базовой информационной процедуры – от него наследуются все создаваемые в системе информационные процедуры.
   - ModelRunner – класс, который выполняет построенную модель.
   - Channel, Port, Message, Event и др.: функциональное назначение которых понятно из названия (канал, порт, сообщение, событие).

   Модуль TriadCore может использоваться независимо ото всей остальной системы, как показано в тестовом приложении TriadCoreTester.
3. TriadEditor – модуль, предоставляющий графический пользовательский интерфейс для редактирования моделей. TriadEditor является пользовательским элементом управления Windows и, следовательно, может встраиваться в другие приложения Windows. Модель, являющуюся результатом действий пользователя, можно получить в виде xml-документа, обратившись к свойству этого компонента. TriadEditor используется в модуле IMPortal для предоставления пользователям системы возможности совместной удаленной работы над моделями.
4. TriadClient – приложение Windows, которое можно применять при однопользовательской работе с моделью или когда нет доступа к Интернет. Это приложение использует модули TriadEditor для редактирования и TriadCore для выполнения моделей.
5. TriadService –сервис Windows, используемый для выполнения моделей.



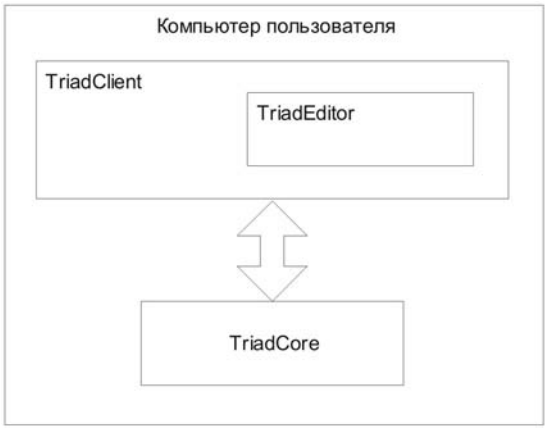
Структура системы Triad.NET для удаленной работы

Как видно из структуры Triad.NET (рис.2. и рис.3.), пользователь может работать с системой как удаленно, так и локально.

Отдельные компоненты модели могут быть выполнены на различных серверах. Количество серверов для выполнения имитационных моделей не ограничено и зависит от загрузки системы.

В случае, когда у пользователя нет доступа к Интернет, он может использовать версию системы для локальной работы. Она состоит из одного приложения Windows, которое использует компонент TriadEditor и библиотеку классов TriadCore. Результаты работы он может сохранять в файлах с расширением *.mod, обмениваться ими с коллегами или добавлять на сервер.

Для создания, редактирования и запуска имитационных моделей используется компонент TriadEditor. Этот компонент может встраиваться как в Web-страницу, так и в обычное приложение Windows. TriadEditor использует функциональность Microsoft Framework 2.0 и требует его наличия на машине клиента.



Структура системы Triad.NET для локальной работы

Заключение

Итак, в докладе приводится краткое описание распределённой системы имитации Triad.Net и подсистемы, реализующей удалённый доступ пользователей к имитационной модели. Взаимодействие модели с пользователем выполняется с помощью специальных программных средств Triad.Net – информационных процедур. Информационные процедуры подключены к интересующим пользователя элементам модели и используются для получения информации о ходе моделирования. Подсистема удалённого доступа включает портал, с помощью программных средств которого пользователь не только получает рецензируемую информацию по интересующей его тематике, имеет возможность обсуждать проблемы с коллегами, но и участвует в совместных проектах по имитационному моделированию. Таким образом, Triad.Net является удобным и полезным инструментальным средством для исследователей.

Библиографический список

1. Ferenci S, Perumalla K., and Fujimoto R. An Approach to Federating Parallel Simulators //Workshop on Parallel and Distributed Simulation, May 2000 (http://www.cc.gatech.edu/ computing/ pads/papers.html)
2. pcl.cs.ucla.edu/projects/parsec
3. www.isi.edu/nsnam/ns/
4. Perumalla K., and Fujimoto R. Interactive Parallel Simulations with the *JANE* Framework//Workshop on Parallel and Distributed Simulation(http://www.cc.gatech.edu/ computing/ pads/papers.html)
5. Mikov A.I. Formal Method for Design of Dynamic Objects and Its Implementation in CAD Systems // Gero J.S. and F.Sudweeks F.(eds), Advances in Formal Design Methods for CAD, Preprints of the IFIP WG 5.2 Workshop on Formal Design Methods for Computer-Aided Design, Mexico, Mexico, 1995. pp.105-127.
6. Mikov A.I. Simulation and Design of Hardware and Software with Triad// Proc.2nd Intl.Conf. on Electronic Hardware Description Languages, Las Vegas, USA, 1995. pp. 15-20
7. Миков А.И. Определение характеристик ожидания в однолинейной системе по моментам исходных распределений// Изв. АН СССР. Техническая кибернетика, №3, 1980
8. Миков А.И., Замятина Е.Б., Фатыхов А.Х. Система оперирования распределенными имитационными моделями сетей коммуникации. //В кн. «Материалы Всероссийской научно-технической конференции «Методы и средства обработки информации МСО-2003».1-3 октября 2003 г., стр. 437-443
9. Миков А.И., Замятина Е.Б. О разработке исследовательского портала «Имитационное моделирование»//В кн. «Материалы Всероссийской научно-технической конференции «ИММОД-2003».23-24 октября 2003, стр.89-91
10. Syrjakow M., Syrjakow E., Szczerbicka H. Towards a Component-Oriented Design of Modeling and Simulation Tools// Proceedings of the International Conference on AI, Simulation and Planning in High Autonomy Systems (AIS 2002), Lisbon, Portugal, April 7-10, 2002
11. Thomas Wiedemann. Next Generation Simulation Environments Founded on Open Source Software And XML-based Standard Interfaces //Proceedings of the 2002 Winter Simulation Conference (E. Yücesan, C.-H. Chen, J. L. Snowdon, and J. M. Charnes, eds.), pp.623-628.
12. Thorsten S. Daum, Robert G. Sargent, A Web -Ready HiMASS: Facilitating Collaborative, Reusable, And Distributed Modeling And Execution Of Simulation Models With XML//Proceedings of the 2002 Winter Simulation Conference E. Yücesan, C.-H. Chen, J. L. Snowdon, and J. M. Charnes, eds.,pp. 634-640
13. Вознесенская Т.В. Математическая модель алгоритмов синхронизации времени для распределённого имитационного моделирования.//В кн. Программные системы и инструменты. Тематический сборник факультета ВМиК МГУ им. Ломоносова №1., стр.56-66
14. Bagrodia, R., The Maisie Visual Programming Environment, Department of Computer Science, University of California, Los Angeles, http://may.cs.ucla.edu/projects/mvpe/.
15. Миков А.И., Замятина Е.Б. Система имитационного моделирования с XML интерфейсом.// Сб. трудов международной школы-семинара «Современные проблемы механики и прикладной математики», Ч1, т1, Воронеж, 2003, с.244-247
16. Миков А.И., Замятина Е.Б. Система имитации с удалённым доступом.//В кн. «Материалы третьей междисциплинарной конференции с международным участием (НБИТТ-21), 21-23 июня 2004, Петрозаводск, стр.73

17. Миков А.И., Замятина Е.Б., Осмехин К.А. Метод динамической балансировки процессов имитационного моделирования.//В кн. «Материалы Всероссийской научно-технической конференции «Методы и средства обработки информации МСО-2003».1-3 октября 2005, стр. 472-477
18. Замятина Е.Б., Муллаханов Р.Х., Фирсов А.Н. On Approach of Researching Portal Development //В кн. «Информационные технологии и телекоммуникации в образовании и науке», 15-22 May, 2005, Turkey, pp.243-246

Информация об авторах

Александр Миков *– директор АНО «Институт компьютинга», Букирева,15, Пермь, Россия, e-mail: Alexander_Mikov@mail.ru*

Елена Замятина *– Пермский государственный университет, Букирева,15, Пермь, Россия, e-mail: e_zamyatina@mail.ru*

Антон Фирсов *– Пермский государственный университет, Букирева,15, Пермь, Россия, e-mail: a_firsov@mail.ru*

ON RELATIONSHIP BETWEEN A SEARCH ALGORITHM AND A CLASS OF FUNCTIONS ON DISCRETE SPACE[7]

Victor Nedel'ko, Svetlana Nedel'ko

Abstract*: The task of revealing the relationship between a search algorithm and a class of functions those it solves is considered. Particularly, there was found a class of functions solvable by some adaptive search algorithm for a discrete space of low cardinality. To find an optimal algorithm exhaustive search was used. Algorithm quality criterion based on equivalence classes was also introduced.*

Keywords*: search algorithm, adaptive search, optimization, global extreme, no free launch theorem.*

Introduction

According to the no free launch theorem [Wolpert, 1996] a cardinality of the set of discrete functions solvable by a search algorithm is the same for all the algorithms. A function here may be thought as a permutation of integer numbers from 1 to n, and a function is called solvable by an algorithm if the algorithm will find global extreme for given number of steps. This theorem means that having no information about class of functions one can't prove one algorithm versus other ones.

At the same time the using some comprehensive search algorithm, for example genetic or adaptive one [Lbov, 1965; Lbov, 1999], practically seems to be more reasonable than the random search without any learning. But a rational argumentation of such preference is possible only if functions solvable by chosen comprehensive algorithm occur in real world tasks more often than others. This work presents the results of a numerical experiment that consists in explicit finding the classes of functions solvable by some algorithms including an adaptive search algorithm.

One can introduce on classes of functions a preferability measure that is projected on algorithms. As such a measure a functions class variety may be used. The simplest variety measure is a number of different extremum positions over all functions in the class. The informativity criterion [Lbov, 2001; Nedel'ko, 2004] also may be used as a variety measure.

On algorithms it's also possible to introduce some complexity measure, for example, as a number of different points arrangements those may by generated by the algorithm over the all functions. For deterministic algorithms this measure corresponds to intuitive measure of "nontriviality".

[7] The work is supported by RFBR, grant 04-01-00858-a

The numerical experiment performed reveals a correlation between an algorithm complexity and a functions class variety.

Since a metric is given on definitional domain of functions, the set of all functions may be split onto equivalence subclasses. If one defines that an algorithm solves a given function only if it solves also all the functions of the correspondent equivalence subclass, the number of solvable functions for different algorithms cease to be equal.

Thereby it is possible to introduce some criteria those allow to say that one algorithm is better than another one.

Formal Statement of the Problem

The task of global extremum search will be considered in the following statement.

Let unknown numerical function f be defined on a discrete set $X = \left\{ x_j \,\middle|\, j = \overline{1,n} \right\}$ and $j \neq k \Rightarrow f(x_j) \neq f(x_k)$. Let's arrange all the values of the function and denote a rank of value $f(x_j)$ by f_j. Then a function f may be thought as a permutation f_j of integer numbers from 1 to n.

A search of extremum (assume that we need a minimum) consists in successive selection of m points from X, i. e. in selection of some subset of indices $J_m = \left\{ j_i \in I_n \,\middle|\, i = \overline{1,m} \right\}$, where I_n is a set of integer numbers from 1 to n. When the next point $x_{j_i} \in X$ is chosen not the true value $f(x_j)$, but only $r_i \in I_i$ – a rank of this value among the all previously taken values got to be known.

So a selection of a next point in X may be represented as a function of all the ranks obtained by previous steps, i. e. $j_i = Q_i(r_1, ..., r_{i-1})$. A set of the all such functions for m steps comprises a search algorithm $Q = \left\{ Q_i \,\middle|\, i = \overline{1,m} \right\}$. Note that for each f an algorithm selects a certain set J_m.

A search algorithm may be represented by a decision tree, where nodes are assigned by j_i – indices of selected points and branches correspond to possible values of ranks r_i. So the root is assigned by the index of a point selected first. Since for the first point there are no points to compare the only one branch issues from the root. The second value may be greater or smaller than the first one therefore there are two branches issuing from the second layer node. Similarly, i branches issue from each node of i-th layer.

As a search quality criterion we shall use $K(Q, f) = \min\limits_{i=\overline{1,m}} f_{j_i}$ – the lowest rank of function values among the points selected by the algorithm.

We shall say that a function f is solvable by an algorithm Q if $K(Q, f) = 1$, i. e. the minimum was found.

The goal of this paper is to reveal a relationship between an algorithm and a class of functions solvable by it.

Research Method

The basic research method in this work is a direct running over all the variants. It poses strong dimensionality restrictions.

The number of all functions is $n!$, so an exhaustive search over all the functions is possible until $n \approx 14$.

The number of nodes in the decision tree for a search algorithm is $\sum\limits_{i=0}^{m} i!$. Since a tree need to be stored parameter m should not be greater than 11.

Under these restrictions it is possible to find the class of functions solvable by any given algorithm using exhaustive search over all the functions.

We are interesting also in finding an algorithm being optimal for given class of functions. To solve this task one need to store all the combinations of selected j_i and received r_i. The number of such variants is $\sum\limits_{i=0}^{m} \frac{n!\, i!}{(n-i-1)!}$.

Therefore the ultimate dimensionality is given by $n \approx 12$, $m \approx 5$.

Permitted for an exhaustive search task complexity is very far from complexities of real practical problems. But it allows modeling quite nontrivial function classes and search algorithms, including an adaptive search algorithm.

No Free Launch Theorem

According to NFL theorem an algorithm quality averaged over the all functions is the same for all the algorithms of search. For the considered statement of a search problem it means that the number of functions solved by an algorithm is the same for all the algorithms. This number is $\frac{m}{n}n!$.

This fact means that to prove an advantage of some algorithm one need to have information about functions to be analyzed. Another way is to introduce some characteristics those provide a ranking of classes of functions.

The Equivalence of Functions

Until now we didn't take into account any properties of space X, although such properties are very important for an optimization algorithm.

One of the most important properties is a presence of the metric.

Assume in X the metric $\delta(x_{j_1}, x_{j_2}) = \Delta(j_1, j_2) = \min(|j_1 - j_2|, n - |j_1 - j_2|)$ to be given. Note that the points x_1 and x_n are adjacent. This metric may be illustrated via placing the points on a circle.

The metric introduced allows to define equivalence relationship on functions. Two functions will be equivalent if they may be transformed one to another via some preserving a distance mapping the space X onto itself. It's easy to see that functions $f'(x)$ and $f''(x)$ are equivalent if and only if the correspondent permutations f'_j and f''_j may be transformed one to another via a round permutation and a mirror reflection if needed.

We shall say that an algorithm solves a function if all the functions of correspondent equivalence class are solvable by this algorithm.

Note that while the number of functions solvable by any algorithm is the same, the number of solvable equivalence classes may be different, so it may be used as a measure of effectiveness.

Adaptive Search Algorithm

Placing five points in a space of cardinality 10 one can only approximately model an adaptive search, but the basic ideas of this algorithm may be represented. The main idea of adaptive search is an inclination of the next selected point both to less investigated areas of X and to already known points with smallest $f(x)$.

But this idea may be realized by various ways, so to avoid ambiguity let's use a class of functions. It's clear that functions solved by adaptive search algorithm should satisfy the condition of that the function values in close points are usually close. Let's take a class F_κ of functions with restricted variation speed, i. e. $f \in F_\kappa \Leftrightarrow |f_{j_1} - f_{j_2}| \le \kappa \Delta(j_1, j_2)$. Let $\kappa = 2$ that is the lowest possible value. Note that if $f \in F_\kappa$ then the all equivalent functions also belong to F_κ.

The content of F_2 by m = 4, n = 10 is shown in table 1 where only one function from each equivalence class is presented.

Under m = 4, n = 10 the class F_2 is solved by the algorithm that represented on figure 1 via a decision tree. Note that this algorithm completely reflects the ideas of adaptive planning. Indeed, the first three points are arranged equidistantly: on the first step the algorithm selects the point number 1, on the second step — the point number 4, on the third step — the point number 7 if $f_4 < f_1$ and the point number 8 else. On the fourth step the algorithm selects a point near to previous points where smaller function values encountered, for example if $f_7 < f_4 < f_1$ then the algorithm selects the point number 6, if $f_4 < f_7 < f_1$ — the point number 5 etc.

Besides introduced in tab. 1 functions the algorithm solves 955 equivalence classes more. Some elements of these classes are in table 2.

The results obtained show that an adaptive algorithm can solve some equivalence classes including functions with rather quick changing. But in all solved classes the functions have minimum and maximum on the far distance.

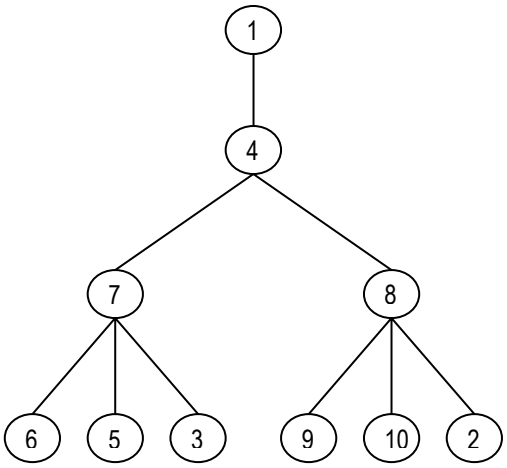


Fig. 1. The decision tree for the algorithm being optimized on the class F_2 by m = 4, n = 10.

Tab. 1. The class F_2

7	5	3	1	0	2	4	6	8	9
7	5	4	1	0	2	3	6	8	9
7	6	3	1	0	2	4	5	8	9
7	6	4	1	0	2	3	5	8	9
8	5	3	1	0	2	4	6	7	9
8	5	4	1	0	2	3	6	7	9
8	6	3	1	0	2	4	5	7	9
8	6	4	1	0	2	3	5	7	9

Tab. 2. Some extra classes of functions solved by the algorithm being optimized on the class F_2

4	5	7	2	0	6	3	1	9	8
5	1	7	3	0	6	4	2	8	9
4	6	7	2	0	5	3	1	9	8
5	6	7	2	0	4	3	1	9	8
6	2	7	1	0	5	3	4	8	9
9	2	5	3	0	4	7	1	6	8
8	1	6	3	0	5	4	2	7	9

Conclusion

The research performed demonstrates that the method of a direct running over all the variants is appropriate for revealing some properties of search algorithms and a relationship between an algorithm and a class of functions solvable by it. The method allows explicit finding an algorithm being optimal for given class of functions. Note that the analytic finding an optimal algorithm is very difficult because an algorithm selecting on a regular step a point with maximal probability of extremum is not generally an optimal algorithm, so one need while planning a regular point to take account of the following steps.

Using this method we have found out the adaptive search algorithm to be optimal for the class of functions with the lowest changing speed.

The using a number of metrical equivalence classes solved by an algorithm as an effectiveness measure was also suggested.

Bibliography

[Wolpert, 1996] D. H. Wolpert, W. G. Macready. No Free Lunch Theorems for Search. Santa Fe Institute repot, SFI-TR-95-02-010, 1996.

[Lbov, 1965] G. S. Lbov. The choice of effective system of interdependent features. // Computational systems, N 19, Novosibirsk, IM SB RAS, 1965, pp. 21–34 (in Russian).

[Lbov, 1999] G. S. Lbov, N. G. Startseva. Logical decision functions and issues of statistical robustness. Novosibirsk, IM SB RAS, 1999, 211 p. (in Russian).

[Lbov, 2001] G.S. Lbov, V.M. Nedel'ko. A Maximum informativity criterion for the forecasting Several variables of different types. // Computer data analysis and modeling. Robustness and computer intensive methods. Minsk, 2001, vol 2, 43–48.

[Nedel'ko, 2004] S. V. Nedel'ko. A transitional matrix informativity criterion and forecasting heterogeneous time series. // Artificial Intelligence, №2, Ukraine, 2004, pp.145–149. (in Russian).

Authors' Information

Victor Mikhailovich Nedel'ko – *Institute of Mathematics SB RAS, Laboratory of Data Analysis, 660090, pr. Koptyuga, 4, Novosibirsk, Russia, e-mail: nedelko@math.nsc.ru*

Svetlana Valeryevna Nedel'ko – *Institute of Mathematics SB RAS, Laboratory of Data Analysis, 630090, pr. Koptyuga, 4, Novosibirsk, Russia, e-mail: nedelko@math.nsc.ru*

A DOMINANT SCHEDULE FOR THE UNCERTAIN TWO-MACHINE SHOP-SCHEDULING PROBLEM

Natalja Leshchenko, Yuri Sotskov

Abstract: *Non-preemptive shop-scheduling problem with random but bounded processing times is studied. In an uncertain version of a shop-scheduling problem there may not exist a unique schedule that remains optimal for all possible realizations of the job processing times t_{jm}. We find necessary and sufficient conditions when there exists a dominant permutation that is an optimal Johnson's permutation for all possible realizations of the processing times t_{jm}. Computational studies show the percentage of the problems solvable under these conditions for the cases of randomly generated instances with n ≤ 100 jobs.*

Keywords: *Scheduling, makespan, uncertainty.*

ACM Classification Keywords: *F.2.2 Non-numerical algorithms and problems: sequencing and scheduling*

Introduction

In scheduling theory, many systems under consideration assume complete information about the scheduling problems to be solved and a static environment within the schedule to be executed. However, the real world is not static: machines break down, activities take longer to execute than expected, and jobs may be added or canceled. There exist a lot of approaches concerning management of uncertainty in scheduling (see surveys [Aytug et al., 2005; Davenport, Beck, 2000; Gupta, Stafford, 2006]). The stochastic method [Elmaghraby, Thoney, 2000; Pinedo, 1995] for dealing with uncertainty is useful when the schedule has enough prior information to characterize the probability distributions of the random processing times and there are a large number of realizations of the similar processes. In the particular case of the stochastic scheduling problem, random processing times may be controllable, and the objective is to choose the optimal processing times (which are under the control of the decision-maker) and the optimal schedule with the chosen processing times. For such a problem, the objective function depends on both the job processing times and the job completion times (see [Jansen, Mastrolilli, Solis-Oba, 2005]). The current trends in the field of scheduling under the fuzziness notion

have been presented in [Slowinski, Hapke, 1999]. In the field of operations research for the problems under uncertainty auxiliary criteria are often used. The most popular auxiliary criteria are criteria by Wald, Hurwicz and Savage (see [Shafransky, 2005] for survey).

In spite of several developments, flow-shop scheduling problem remains largely unsolved (see [Gupta, Stafford, 2006]). In most of these developments, Johnson's rule and analysis methods play a significant role. In this paper we consider a shop-scheduling problem with interval processing times. A scheduling problem with interval processing times is rather general, since most events that are uncertain before scheduling may be considered as factors that vary the job processing times. The processing time may depend on the distance between machines, the type of transport used, traffic conditions, intervals of availability of machines, possible machine breakdowns, emergence of new unexpected jobs with high priority, early or late arrival of raw materials. In survey [Gupta, Stafford, 2006], there were presented 21 restrictions involved in the classical flow-shop problem $F \,||\, C_{\max}$ with fixed job processing times, where criterion $C_{\max}$ denotes minimization of schedule length. Nine of these 21restrictions addressed the criterion and type of the processing system. All the remaining restrictions may be overcome by using the suitable intervals for possible variations of the job processing times.

Problem Statement

First, we consider the non-preemptive flow-shop scheduling problem $F2 \,|\, t_{jm}^{L} \le t_{jm} \le t_{jm}^{U} \,|\, C_{\max}$ with random but bounded processing times. Two machines $M = \{M_1, M_2\}$ have to process *n* jobs $J = \{1, 2, ..., n\}$ with the same two-stage (machine) routes. All the *n* jobs are available to be processed from time τ = 0. In contrast to deterministic scheduling problem, it is assumed that processing time t_{jm} of job $j \in J$ by machine $M_m \in M$ is not fixed before scheduling. In a realization of the process, t_{jm} may be equal to any real value between lower bound t_{jm}^{L} and upper bound t_{jm}^{U} being given before scheduling (the probability distribution of random processing time is unknown before scheduling). We address the stochastic flow-shop problem for the case when it is hard to obtain exact probability distributions for random but bounded processing times, and when assuming a specific probability distribution is not realistic before scheduling. (In such a case there may not exist a unique schedule that remains optimal for all possible realizations of the job processing times.) It has been observed that, although the exact probability distribution of the job processing times may be unknown in advance, upper and lower bounds on the job processing times are easy to obtain in many practical cases.

If equality $t_{jm}^{L} = t_{jm}^{U}$ holds for each job $j \in J$ and each machine $M_m \in M$, then problem $F2 \,|\, t_{jm}^{L} \le t_{jm} \le t_{jm}^{U} \,|\, C_{\max}$ turns into a deterministic flow-shop problem (denoted as $F2 \,||\, C_{\max}$) that is polynomially solvable due to [Johnson, 1954]. In contrast to deterministic problem $F2 \,||\, C_{\max}$ we call problem $F2 \,|\, t_{jm}^{L} \le t_{jm} \le t_{jm}^{U} \,|\, C_{\max}$ an uncertain scheduling problem.

In paper [Johnson, 1954], it was proposed the $O(n \log n)$ algorithm for constructing an optimal schedule for the deterministic flow-shop problem $F2 \,||\, C_{\max}$. Permutation $\pi = (i_1, i_2, ..., i_{n_{12}})$ that satisfies conditions

$$\min\{t_{i_l 1}, t_{i_{l+1} 2}\} \le \min\{t_{i_{l+1} 1}, t_{i_l 2}\},\ l = \overline{1, n_{12} - 1},$$

is called a Johnson's permutation. At least one optimal permutation for problem $F2 \,||\, C_{\max}$ is a Johnson's permutation. Note, that for the problem $F2 \,||\, C_{\max}$ optimal schedule may also be defined by permutation that does not satisfy the above Johnson's conditions.

Existence of a Dominant Johnson's Permutation for the Uncertain Flow-Shop Problem

Let *T* denote a set of feasible vectors $t = (t_{11}, t_{12}, ..., t_{n1}, t_{n2})$ of the job processing times:

$$T = \left\{ t \,|\, t_{jm}^{L} \le t_{jm} \le t_{jm}^{U},\ j \in J, m \in M \right\}.$$

The set *S* of all feasible permutations (schedules) has the cardinality $|S| = n!$. Permutation $\pi_i \in S$ *dominates* each other permutation $\pi_k \in S$ with $k \neq i$ if inequality $C_{\max}(\pi_i, t) \leq C_{\max}(\pi_k, t)$ hold for each $\pi_k \in S$, where $C_{\max}(\pi_i, t)$ denotes objective value $C_{\max}$ to the deterministic problem $F2 \| C_{\max}$ with the vector $t \in T$ of the job processing times.

We call permutation $\pi_i \in S$ *a dominant Johnson's permutation to the uncertain problem* $F2 | t_{jm}^L \leq t_{jm} \leq t_{jm}^U | C_{\max}$ if for any feasible vector $t \in T$ of the job processing times permutation π_i is a Johnson's permutation for the deterministic problem $F2 \| C_{\max}$ with this vector $t \in T$ of the job processing times.

We consider the case when inequality $t_{jm}^L < t_{jm}^U$ holds for each job $j \in J$ and each machine $M_m \in M$. For this case in [Leshchenko, Sotskov, 2005] the following theorem has been proven.

Theorem 1 *Let* $t_{jm}^L < t_{jm}^U$, $j \in J$, $M_m \in M$. *Then there exists a dominant Johnson's permutation* $\pi_i \in S$ *to the uncertain problem* $F2 | t_{jm}^L \leq t_{jm} \leq t_{jm}^U | C_{\max}$ *if and only if:*

a) for any pair of jobs j_i *and* j_j *with* $t_{k1}^U \leq t_{k2}^L$, k = i, j (with $t_{k2}^U \leq t_{k1}^L$, k = i, j, respectively) either $t_{i1}^U \leq t_{j1}^L$ *or* $t_{j1}^U \leq t_{i1}^L$ *(either* $t_{i2}^U \leq t_{j2}^L$ *or* $t_{j2}^U \leq t_{i2}^L$*),*

b) there exists at most one job j^* *such that* $t_{j1}^L < t_{j2}^U, t_{j2}^L < t_{j1}^U$, *and the following inequalities hold:* $t_{j^*1}^L \geq \max\{t_{i1}^U \mid t_{i1}^U \leq t_{i2}^L\}$, $t_{j^*2}^L \geq \max\{t_{i2}^U \mid t_{i2}^U \leq t_{i1}^L\}$.

Existence of a Dominant Jackson's Pair of Permutations for the Uncertain Job-Shop Problem

We consider uncertain job-shop problem $J2 | t_{jm}^L \leq t_{jm} \leq t_{jm}^U | C_{\max}$. Let $J_{12} \in J$ be set of jobs with route (M_1, M_2) (first, job $j \in J_{12}$ has to be processed by machine M_1, and then by machine M_2), $J_{21} \in J$ be set of jobs with route (M_2, M_1), and $J_m \in J$ be set of jobs that are processed only by one machine $M_m \in M$.

If equality $t_{jm}^L = t_{jm}^U$ holds for each job $j \in J$ and each machine $m \in M$, then problem $J2 | t_{jm}^L \leq t_{jm} \leq t_{jm}^U | C_{\max}$ turns into a deterministic job-shop problem (denoted as $J2 \| C_{\max}$). In paper [Jackson, 1956], it was proven that the optimal schedule for the problem $J2 \| C_{\max}$ may be defined by the two permutations (π', π''), where π' is a permutation of jobs $J_1 \cup J_{12} \cup J_{21}$ on machine M_1, and π'' is a permutation of jobs $J_2 \cup J_{12} \cup J_{21}$ on machine M_2. One can find an optimal schedule as a pair of permutations (π', π'') in the following form: ($\pi' = (\pi_{12}, \pi_1, \pi_{21})$, $\pi'' = (\pi_{21}, \pi_2, \pi_{12})$), where job *j* belongs to permutation π_l if and only if $j \in J_l$, $l \in \{1,2,12,21\}$ and permutations π_{12} (permutation π_{21}) is the Johnson's permutation of jobs of set J_{12} (of jobs of set J_{21} when machine M_2 is the first machine in the route and machine M_1 is the second machine in the route). Since the order of the jobs in set J_1 and in set J_2 may be arbitrary, we can fix both permutations π_1 and π_2, e.g., we can order jobs in these permutations with respect to their job numbers. We call pair of permutations (π', π'') a Jackson's pair of permutations if it satisfies all the above conditions.

The jobs of set J_{12} (set J_{21}) give uncertain flow-shop problem with machine route (M_1, M_2) (with opposite machine route (M_2, M_1) for jobs of set J_{21}). Therefore, for such a particular case of an uncertain flow-shop problem we can use Theorem 1, and so there exists a Johnson's permutation that is optimal for any vector $t \in T$ of job processing times if and only if the corresponding conditions of Theorem 1 hold.

We call pair of permutations (π', π'') *a dominant Jackson's pair of permutations to the uncertain problem* $J2 \mid t_{jm}^L \le t_{jm} \le t_{jm}^U \mid C_{\max}$ if for any feasible vector $t \in T$ of the job processing times pair of permutations (π', π'') is a Jackson's pair of permutations for the deterministic problem $J2 \mid\mid C_{\max}$ with this vector $t \in T$ of the job processing times.

Again, we consider the case when inequality $t_{jm}^L < t_{jm}^U$ holds for each job $j \in J$ and each machine $M_m \in M$. Theorem 7 in paper [Leshchenko, Sotskov, 2005] gives sufficient condition for existence of a solution to an uncertain job-shop problem. As a result, we can obtain the following condition for existence of a pair of permutations (π', π'') that is a dominant Jackson's pair of permutations for the deterministic problem $J2 \mid\mid C_{\max}$ with any feasible vector $t \in T$ of the job processing times.

Theorem 2 *Let* $t_{jm}^L < t_{jm}^U$, $j \in J$, $M_m \in M$. *Then there exists a dominant Jackson's pair of permutations* (π', π'') *for the problem* $J2 \mid t_{jm}^L \le t_{jm} \le t_{jm}^U \mid C_{\max}$ *if for jobs from set* J_{12} *conditions a) and b) of Theorem 1 hold, and for jobs from set* J_{21} *the corresponding conditions a) and b) of Theorem 1 hold.*

Computational Results

In this section, we consider randomly generated uncertain flow-shop problems $F2 \mid t_{jm}^L \le t_{jm} \le t_{jm}^U \mid C_{\max}$ and answer experimentally the question of how many uncertain instances have a Johnson's permutation that is optimal for all corresponding deterministic problems $F2 \mid\mid C_{\max}$ with any feasible vector $t \in T$ of the job processing times. For each randomly generated instance $F2 \mid t_{jm}^L \le t_{jm} \le t_{jm}^U \mid C_{\max}$ we tested whether condition of Theorem 1 held.

The computational algorithm was coded in MATLAB. For the computational experiments, we used an AMD 3000 MHz processor with 1024 MB main memory. For all instances we made 1000 tests in each series (for each combination of *n* and *L*). Of course, the running time increases with increasing the product *nL*. Fortunately, the running time for our analysis remains rather small (less than 4 seconds) for all series under consideration.

Random instances were generated as follows. First, we tested problems with "short" intervals of the job processing times:

$n \in \{5, 10, 15, \ldots, 100\}$ jobs, with integer processing times uniformly distributed in the range [1,1000], and

$L \in \{1, 2, 3, \ldots, 10\}$.

For each operation, the lower bound was randomly generated and the upper bound was computed as follows: $t_{jm}^U = t_{jm}^L + L$.

We tested also problems with "long" intervals of the job processing times:

$n \in \{5, 10, 15, \ldots, 100\}$ jobs, with integer processing times uniformly distributed in the range [1,1000], and

$L \in \{1, 2, 3, \ldots, 10\}$.

For each operation, the lower bound was randomly generated and the upper bound was computed as follows: $t_{jm}^U = t_{jm}^L (1 + L\% / 100\%)$.

Tables 1 - 4 present the percentage of instances in the series for which conditions of Theorem 1 hold.

Along with the processing times uniformly distributed in the range [1,1000], we tested the cases when the processing times are uniformly distributed in the range [1,10000]. The computational results for the latter problems are given in Tables 2 and 4, which are analogous to Tables 1 and 3.

Theoretically, the case with $t_{jm}^{U} = t_{jm}^{L}(1 + L\%/100\%)$ seems to be harder than that with $t_{jm}^{U} = t_{jm}^{L} + L$ since lengths of the intervals of the job processing times increase when value of lower bound increases. From our experiment, it follows that increasing simultaneously both numbers *n* and *L* decreases the number of solvable instances. Comparing Tables 3 and 4 show that there are no computational differences between cases with "long" intervals and the processing times uniformly distributed in the range [1,1000] and in the range [1,10000]. These problems are the hardest ones for our analysis based on Theorem 1.

The easiest class of problems in our experiments was that with condition $t_{jm}^{U} = t_{jm}^{L} + L$ and processing times uniformly distributed in the range [1,10000]. Obviously, in this case uncertain problem $F2 \mid t_{jm}^{L} \le t_{jm} \le t_{jm}^{U} \mid C_{\max}$ is closed to the deterministic problem $F2 \mid\mid C_{\max}$.

For the job-shop problem $J2 \mid t_{jm}^{L} \le t_{jm} \le t_{jm}^{U} \mid C_{\max}$ the number of instances, solvable due to Theorem 2 may be close to that for the flow-shop problem $F2 \mid t_{jm}^{L} \le t_{jm} \le t_{jm}^{U} \mid C_{\max}$ since there exists a main machine that defines the makespan, for the job-shop problem, and we have to test condition of Theorem 1 only for jobs of set J_{12} (or for jobs of set J_{21}, respectively)

Table 1. Percentage of instances solvable due to Theorem 1 with the processing times uniformly distributed in the range [1,1000], and with $t_{jm}^{U} = t_{jm}^{L} + L$.

L *n*	5	10	15	20	25	30	35	40	45	50	55	60	65	70	80	90	100
1	99.2	95.2	91.2	86.1	79.2	72.8	63.9	58.1	49.2	41.9	34.2	29.5	24	17.5	10.1	6.5	3.5
2	97.2	89.8	77.6	63.6	51	39.6	28.2	16.4	13.3	7.8	3.3	2.9	0.9	0.4	0.1	0	0.1
3	95	80.9	66.4	47.6	32.8	20.6	9.4	6.2	3.2	1.6	0.5	0.1	0	0	0	0	0
4	91.8	78.6	56	39.2	20.3	10.7	4.5	2.1	0.6	0.2	0	0	0	0	0	0	0
5	91	69.4	44.9	28.9	14.6	6	2	0.4	0.2	0	0	0	0	0	0	0	0
6	89.1	65	42.2	22.1	8.2	2.7	0.4	0.4	0.1	0.1	0	0	0	0	0	0	0
7	87.6	61	33.7	13.4	4.6	1.2	0.4	0.1	0	0	0	0	0	0	0	0	0
8	86.8	54.2	27.3	12.7	3.2	0.9	0.2	0	0	0	0	0	0	0	0	0	0
9	86.9	48.3	24.7	7.2	1.9	0.2	0	0	0	0	0	0	0	0	0	0	0
10	84.8	50.3	19	5.5	2	0.5	0.1	0	0	0	0	0	0	0	0	0	0

Table 2. Percentage of instances solvable due to Theorem 1 with the processing times uniformly distributed in the range [1,10000], and with $t_{jm}^{U} = t_{jm}^{L} + L$.

L *n*	5	10	15	20	25	30	35	40	45	50	55	60	65	70	80	90	100
1	100	99.6	99.6	98.6	97.6	96.8	96.2	94	94.3	90.4	88.9	87.6	86.4	84.8	80.2	75.1	71.3
2	99.7	98.9	97.6	95.2	93.6	92.7	86.8	85.3	82.9	76.4	73.2	71.1	63	59.4	50	44.5	34
3	99.9	97.4	95.2	93.7	90.2	86	82.6	73.5	69	64.2	58.5	55.8	46.3	42.5	34.3	25.2	17.5
4	98.9	97.6	93	89.2	83.2	81.5	74.7	67.9	62.7	52.8	49.3	42.6	39.5	31.8	22.3	14.8	9.9
5	99.4	95.9	92.9	87.1	80.1	77.2	68.6	60.8	53.1	45.1	40.8	31.6	27	20.2	14.7	7.8	4.4
6	98.7	96.2	91.9	85.9	79.9	70.8	61.4	54.9	49.1	38.7	32.9	24.9	20.6	15.5	9	4.6	3.1
7	99.2	95.4	89	82.9	72.5	67.7	56.2	49.5	41.2	32.5	25.7	19.7	14.2	12.6	5.6	3.1	0.7
8	98.1	93.6	87.6	79.7	68.2	62.2	53.8	44.3	35.9	24.4	20.8	16	12.1	6.6	3.6	1.2	0.2
9	98	93.5	86.1	76.4	68.4	58.4	48.1	41.8	28.9	23.1	17.4	12	7	5.5	2.5	0.6	0.1
10	98.2	95.2	86.5	76.5	66	52.4	45.7	34.8	27	17.9	12.6	8.7	6.6	3.9	1.9	0.1	0

Table 3. Percentage of instances solvable due to Theorem 1 with the processing times uniformly distributed in the range [1,1000] and with $t_{jm}^{U} = t_{jm}^{L}(1 + L\%/100\%)$

L *n*	5	10	15	20	25	30	35	40	45	50	55	60	65	70	75	80	85	90	95	100
1	94.5	85.1	70	53.5	36.9	24.9	14.7	8.2	4.8	1.9	1.3	0.6	0.3	0	0	0	0	0	0	0
2	91.2	69.3	45.6	24.2	11.8	4.2	1.1	0.6	0.1	0	0	0	0	0	0	0	0	0	0	0
3	87.7	58.4	28.3	10.3	3.8	1.3	0.2	0	0	0	0	0	0	0	0	0	0	0	0	0
4	82.4	47.4	18.8	5.5	0.8	0.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	75.5	37.4	11.3	2.2	0.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	72.5	32.4	7.8	1.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	66.9	25	5.4	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	65.8	19.9	2.1	0.4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	63.4	18	2.4	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	59.2	14.4	2	0.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 4. Percentage of instances solvable due to Theorem 1 with the processing times uniformly distributed in the range [1,10000] and with $t_{jm}^{U} = t_{jm}^{L}(1 + L\%/100\%)$.

L *n*	5	10	15	20	25	30	35	40	45	50	55	60	65	70	75	80	85	90	95	100
1	94.5	81.5	63.4	45.9	28.9	18.9	11.3	5.3	3.2	1	0.4	0.3	0	0	0	0	0	0	0	0
2	89	69.2	39.3	22.5	10.1	4.1	1.2	0.4	0	0	0	0	0	0	0	0	0	0	0	0
3	84.3	55.8	26.9	9.7	2.3	0.4	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	80.2	44.6	17.5	4.4	1.2	0	0.2	0	0	0	0	0	0	0	0	0	0	0	0	0
5	76.1	35.2	10.4	2.2	0.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	70.3	30.4	6.8	0.8	0.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	71	24.6	6.3	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	63.6	20.6	3.4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	61.6	18.5	1.9	0.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	59.9	12.4	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Conclusion and Acknowledgments

It is clear that in spite of uncertainty of the job processing times it is necessary to choose only one schedule for the practical realization of the process. Theorem 1 allows obtain (without fail) a dominant permutation before realization of the process. Indeed, such a dominant schedule (if any) is the best one for any feasible realization of the job processing times.

Clearly, this approach is useful if the level of uncertainty is low enough (and the best results were obtained for the case considered in Table 2). If uncertainty exceeds a certain threshold, then others approaches outperform the approach based on Theorem 1. If there is no possibility to construct a dominant schedule, it may be fruitful to construct more general schedule form on the basis of partial job order. One can also consider the realization stage of a flow-shop scheduling, when a part of the schedule is already realized. It is interesting to know how to use additional information about realized operations in order to obtain better solution than that constructed before scheduling. It is also interesting to find sufficient conditions for choosing the unique permutation that is optimal for any feasible processing times of the remaining operations. In such a case, a realistic solution process can be seen as consisting of *static* and *dynamic* phases. At the *static* phase, a scheduler can construct a family of the dominant permutations. At the *dynamic* phase of the decision-making, a scheduler has to select an appropriate schedule from such a family of the dominant permutations to react in real-time to the actual processing times of the already completed jobs. If a scheduler cannot make right decision at time point $\tau > 0$, he (she) has to use one of the solution policies, which does not guarantee to find an optimal schedule for any realization of the remaining job processing times. The solution policy may be optimistic or pessimistic (see [Aytud *et al.*, 2005]), or a scheduler can minimize objective function in average.

Some results of such investigations are given in paper [Ng *et al.*, 2006], where it is illustrated how to use a family of the dominant permutations during practical realization of the process when side information about processing times of the jobs that already completed becomes available for a decision-maker. Such an approach falls into the category of predictive-reactive scheduling. The static phase may be considered as predictive scheduling and dynamic phase as reactive scheduling (see [Aytud *et al.*, 2005; Gupta, Stafford, 2006]).

For an uncertain problem it may be necessary to look for an optimal scheduling policy that stochastically minimizes the makespan [Ku, Niu, 1986; Pinedo, 1995]. To this end, it is necessary to obtain the reliable probability distributions for the random processing times. In general case, the choice of the job may also be based on minimization of possible loss of the objective function value.

Acknowledgements

The research was partially supported by INTAS (project 03-51-5501) and ISTC (project B-986). The authors would like to thank Ms. Natalia Egorova for help in computational experiments.

Bibliography

[Aytug *et al*., 2005] H. Aytug, M.A. Lawley, K. McKay, S. Mohan, and R. Uzsoy. Executing production schedules in the face of uncertainties: A review and some future directions. European Journal of Operational Research. 161, 86-110, 2005.

[Davenport, Beck, 2000] A.J. Davenport, and J.C. Beck A survey for techniques for scheduling with uncertainty. Unpublished, available from http://www.eil.toronto.ca/profiles/chris/chris.papers.html, 2000.

[Elmaghraby, Thoney, 2000] S. Elmaghraby, and K.A. Thoney. Two-machine flowshop problem with arbitrary processing time distributions. IIE Transactions, 31, 467-477, 2000.

[Gupta, Stafford, 2006] J.N.D. Gupta, and E.F. Stafford Jr. Flowshop scheduling research after five decades. European Journal of Operational Research. 169, 699-711, 2006.

[Jackson, 1956] J.R. Jackson. An extension of Johnson's results on job lot scheduling. Naval Res. Logist. Quart., 3, 201-203, 1956.

[Jansen, Mastrolilli, Solis-Oba, 2005] K. Jansen, M. Mastrolilli, R. Solis-Oba. Approximation schemes for job shop scheduling problems with controllable processing times. European Journal of Operational Research. 167, 297-319, 2005.

[Johnson, 1954] S.M. Johnson. Optimal two- and three-stage production schedules with setup times included. Naval Research Logistics Quarterly, 1, 61—68, 1954.

[Ku, Niu, 1986] P.S. Ku, and S.C. Niu. On Johnson's two-machine flow-shop with random processing times. Operations Research. 34, 130-136, 1986.

[Leshchenko, Sotskov, 2005] N.M. Leshchenko, and Yu.N. Sotskov. Two-machine minimum-length shop-scheduling problems with uncertain processing times. In: Proceedings of XI International Conference “Knowledge-Dialogue-Solution”, June 20-24, Varna, Bulgaria, 375-381, 2005.

[Ng *et al.*, 2006] C.T. Ng, N.M. Leshchenko, Yu.N. Sotskov and T.C.E. Cheng. Two-machine flow-shop minimum-length scheduling problem with interval processing times. Computers & Operations Research (submitted), 2006.

[Pinedo, 1995] M. Pinedo. Scheduling: Theory, Algorithms, and Systems. Prentice-Hall, Englewood Cliffs. 1995.

[Slowinski, Hapke, 1999] R. Slowinski, and M. Hapke. Scheduling Under Fuzziness. Physica-Verlag, Heidelberg, New York. 1999.

[Shafransky, 2005] Ya.M. Shafransky. Scheduling problems with uncertain parameters: research directions and some results. Informatika. 3, 5-15, 2005 (in Russian).

Authors' Information

Natalja M. Leshchenko – *United Institute of Informatics Problems of National Academy of Sciences of Belarus, Surganova str., 6, 220012, Minsk, Belarus; e-mail: leshchenko@newman.bas-net.by*

Yuri N. Sotskov – *Prof., DSc. United Institute of Informatics Problems of National Academy of Sciences of Belarus, Surganova str., 6, 220012, Minsk, Belarus; e-mail: sotskov@newman.bas-net.by*

STABILITY ANALYSIS OF AN OPTIMAL ASSEMBLY LINE BALANCE WITH RESPECT TO UNCERTAIN OPERATION TIMES

Yuri N. Sotskov

Abstract: *Two simple assembly line balancing problems are addressed. The first problem is to minimize number of linearly ordered stations for processing n partially ordered operations V = {1, 2, ..., n} within the fixed cycle time c. This problem is denoted as SALBP-1. The second problem is to minimize cycle time for processing partially ordered operations V on the fixed set of m linearly ordered stations. The latter problem is denoted as SALBP-2. The processing time t_i of operation $i \in V$ are given before solving SALBP-1 and SALBP-2. However, during the life cycle of the assembly line the values t_i are definitely fixed only for the subset of automated operations $V \setminus \widetilde{V}$. Another subset $\widetilde{V} \subseteq V$ includes manual operations, for which it is impossible to fix exact processing times during the whole life cycle of the assembly line. If $j \in \widetilde{V}$, then operation times t_j can differ for different cycles of production process. For the optimal line balance* ***b*** *of such an assembly line with vector t = (t_1, t_2, …, t_n) of the operation times, we investigate stability of its optimality with respect to possible variations of the processing times t_j of the manual operations $j \in \widetilde{V}$. In particular, we present necessary and sufficient conditions when optimality of line balance* ***b*** *is stable for problem SALBP-1 (SALBP-2) with respect to variations of operation times t_j, $j \in \widetilde{V}$. It is shown how to calculate the maximal value of independent variations of the processing times of all the manual operations, which definitely keep the feasibility and optimality of the line balance* ***b***.

Keywords: *Scheduling, robustness and sensitivity analysis, assembly line.*

ACM Classification Keywords: *F.2.2 Nonnumerical algorithms and problems: sequencing and scheduling.*

Introduction

A single-model paced assembly line, which continuously manufactures homogeneous product in large quantities, is addressed. Terminology and main notations are given in survey [Baybars, 1986] and monograph [Scholl, 1999]. The assembly line is a sequence of *m* linearly ordered stations, which are linked by a conveyor belt or other material handling equipment. Each station of the assembly line has to perform the same set of operations repeatedly during the life cycle of the assembly line. Set of operations *V,* which have to be processed on the assembly line within one cycle time *c,* is fixed. *Simple Assembly Line Balancing Problem* (SALBP) is to find an optimal balance of the assembly line for cycle time *c*, i.e., to find a *feasible assignment* of all operations *V* into a minimal possible number *m* of stations. In [Scholl, 1999], abbreviation SALBP-1 is used for such a problem provided that cycle time *c* is fixed. Each operation $i \in V$ is considered indivisible: an operation has to be completely processed on one station within one cycle time. All the *m* stations start simultaneously the sequences of their operations and buffers between stations are absent.

In this paper, it is assumed that set *V* includes operations of two main types. On the one hand, subset $\widetilde{V} \subseteq V$ includes all the operations, for which it is impossible to fix exact processing times for the whole life cycle of the assembly line (*manual operations*). On the other hand, operation $i \in V \setminus \widetilde{V}$ is one with operation time t_i being fixed during the life cycle of the assembly line (*automated operations*). The known technological factors define a partial order on the set of operations *V*. Digraph *G = (V, A)* with vertices *V* and arcs *A* defines partially ordered set of operations *V* = {1, 2, ..., *n*}, which have to be processed on the assembly line within cycle time *c*. Without loss of generality, it is assumed that $\widetilde{V} = \{1, 2, ..., \widetilde{n}\}$ and $V \setminus \widetilde{V} = \{\widetilde{n} + 1, \widetilde{n} + 2, ..., n\}$, where $0 \le \widetilde{n} \le n$. The vectors of the operation times are denoted as $\widetilde{t} = (t_1, t_2, ..., t_{\widetilde{n}})$, $\bar{t} = (t_{\widetilde{n}+1}, t_{\widetilde{n}+2}, ..., t_n)$, $t = (\widetilde{t}, \bar{t})$ = (t_1, t_2, ..., t_n). Thus, the set of *n* operations may be presented as follows: $V = \{1, 2, ..., \widetilde{n}, \widetilde{n}+1, ..., n\}$. If $\widetilde{n} = 0$, then $\widetilde{V} = \varnothing$. If $\widetilde{n} = n$, then $\widetilde{V}$ = *V*. The assignment $V = V_1 \cup V_2 \cup ... \cup V_m$ of set of *n* operations *V*

to *m* linearly ordered stations $S = (S_1, S_2, \ldots, S_m)$ (i.e., partition of set *V* into *m* mutually disjoint non-empty subsets V_k, $k \in \{1, 2, \ldots, m\}$) is *feasible* operation assignment (called *line balance*) if the following two conditions hold.

Condition 1: *Feasible operation assignment does not violate the precedence constraints given by digraph G = (V, A), i.e., inclusion (i, j) $\in$ A implies that operation i is assigned to station S_k: $i \in V_k$, and operation j is assigned to station S_l: $j \in V_l$, such that $1 \le k \le l \le m$.*

Condition 2: Cycle time c is not violated for each station S_k, $k \in \{1, 2, \ldots, m\}$, i.e., sum of the processing times of all the operations assigned to station S_k (called station time), has to be not greater than cycle time c:

$$\sum_{i \in V_k} t_i \le c . \qquad (1)$$

For SALBP-1, line balance ***b*** is *optimal* when it uses the minimal number of *m* stations and when Condition 1 and Condition 2 are satisfied for line balance ***b***.

Constructing an optimal line balance for SALBP-1 is binary NP-hard problem even for the case of two stations used in the optimal line balance (i.e., if $S = (S_1, S_2)$), empty precedence constraints (i.e., if $A = \varnothing$), and fixed processing times of all the operations V processed on the assembly line (i.e., if $\widetilde{V} = \varnothing$).

The latter claim may be easily proven by polynomial reduction of NP-complete *partition problem* [Garey, Johnson, 1979] to SALBP-1 with two stations and with $A = \varnothing$ (see [Scholl, 1999]). For the sake of simplicity, notation

$$t(V_k) = \sum_{i \in V_k} t_i \qquad (2)$$

is used for the original vector $t = (t_1, t_2, \ldots, t_n)$ of the operation times. Set *V* includes operations of two main types: subset $\widetilde{V}$ of operations with variable processing time (*manual operations)* and subset $V \setminus \widetilde{V}$ of operations with fixed processing time (*automated operations*). We assume that, if $j \in \widetilde{V}$, then operation time t_j is a given non-negative real number: $t_j \ge 0$. However, the value of this operation time can vary during life cycle of the assembly line and can even be equal to zero. Zero operation time t'_j will mean that operation $j \in V_k \bigcap \widetilde{V}$ will be processed (by an *additional worker*) in such a way that processing operation *j* will do not increase station time for S_k for the new vector $t = (\widetilde{t}', \bar{t}) = (t'_1, t'_2, \ldots, t'_{\widetilde{n}}, t_{\widetilde{n}+1}, \ldots, t_n)$ of the operation times: $\sum_{i \in V_k} t_i' = \sum_{i \in V_k \setminus \{j\}} t_i'$. The latter equality is possible if $t'_j = 0$. If $i \in V \setminus \widetilde{V}$, then operation time t_i is given real number fixed during the life cycle of the assembly line. We assume that $t_i > 0$ for each operation $i \in V \setminus \widetilde{V}$. As far as the processing time of the automated operation is fixed, one can consider only such automated operations, which have strictly positive processing times. Indeed, an operation with *fixed zero* processing time has no influence on the solution of SALBP-1. In contrast to usual *stochastic* problems (see surveys [Erel, Sarin, 1998; Sarin, Erel, Dar-El, 1999]), we do not assume any probability distribution known in advance for the random processing times of the manual operations. Moreover, this paper does not deal with concrete algorithms for constructing an optimal line balance in a stochastic environment. It is assumed that the optimal line balance ***b*** is already constructed for the given vector $t = (t_1, t_2, \ldots, t_n)$ of the operation times. Our main aim is to investigate the stability of the optimality of a line balance ***b*** with respect to independent variations of the processing times of all the manual operations $\widetilde{V} = \{1, 2, \ldots, \widetilde{n}\}$ or a portion of the manual operations. More precisely, we investigate the *stability radius* of an optimal line balance ***b***, which may be interpreted as the maximum of simultaneous independent variations of the manual operation times with definitely keeping optimality of line balance *b*.

Motivation, Notations, and Definition

An assembly line balancing problem with fixed cycle time (denoted as SALBP-1) arises when a new assembly line must be installed, and the internal demands and properties of the assembly line have to be estimated. Cycle time *c* is defined on the basis of customer demands in the finished products. The value of *c* may be calculated as the ratio of available operating time of the assembly line and production volume for the same calendar interval. One of the common mathematical problems at the stage of assembly line design is SALBP-1.

This problem may also arise when cycle time *c* of acting assembly line has to be changed because of changing customer demands in the finished product. In the real-world assemble lines, processing times of some operations may be known exactly and fixed for a long time (e.g., if operation has to be done by *fully-automated* machine or by *semi-automated* machine). Modern machines and robots are able to work permanently at a constant speed for a long time. However, in many cases it is not realistic to assume constant operation times for some operations, e.g., if an operation has to be done by a human operator with rather simple tools. In the case of a human work, operation time is subject to physical, psychological, and other factors. Due to the learning of operators, the operation times during the first days (weeks, months) of a life cycle of the assembly line may differ considerably from the processing times of the same operations during the later days (weeks, months). Moreover, some workers can leave the plant, and new workers with lower (or higher) skills have to replace them.

In the case of changeable operation times, it is important to know the credibility of the optimal line balance at hand with respect to possible independent variations of all or a portion of the operation times. Line balance ***b***, which is optimal for the original vector $t = (\tilde{t}, \bar{t})$ of the operation times, may lose its optimality (and even feasibility) for some new vector $t' = (\tilde{t}', \bar{t})$ of the operation times. For example, due to increasing of some operation times, line balance ***b*** may become infeasible for cycle time *c* since inequality (1) may be violated. In such a case, it is necessary to look for another line balance and to use it, if possible, for a suitable modification of production process on the assembly line. Also, line balance ***b*** may lose its optimality with saving its feasibility. It may occur if another operation assignment, say, $\boldsymbol{b_s}$ become feasible for the modified vector $t' = (\tilde{t}', \bar{t})$ of the operation times, and $\boldsymbol{b_s}$ uses less stations than line balance ***b*** uses. Of course, each re-engineering and modification of the assembly line being in process takes an additional time and other expenditure. So, assembly line modification has to be started, if it is really necessary: when the income from the re-engineering and modification will be larger than the total expenditure caused by the re-engineering. Thus, an evaluation of expenditures and benefits should be conducted before deciding whether re-engineering of assembly line is necessary. However, these expenditures and benefits are difficult to evaluate before the end of the re-engineering process. In this paper, we present some sufficient conditions for keeping the optimality of the line balance being in process. For example, re-engineering is not necessary if the currently applied line balance remains optimal in spite of the changes of the operation times $\tilde{t}$.

To test whether line balance ***b*** remains *feasible* for the new vector $t' = (\tilde{t}', \bar{t})$ of the operation times takes $O(\tilde{n})$ time (if station times are included in the input data) or $O(n)$ time (otherwise). Indeed, for the new operation times we have to verify inequality (1) for each subset $V_k, k = 1,2,...,m,$ that includes at least one manual operation with changed processing time in the new vector t'. On the other hand, in the case of feasibility of the line balance ***b*** for the new vector t', in order to test its optimality for t' we have to solve the NP-hard problem SALBP-1. Intuitively, it is clear that sufficiently small changes of the operation times $t_1, t_2, ..., t_{\tilde{n}}$ may keep line balance ***b*** feasible and optimal for the new vector $t' = (\tilde{t}', \bar{t})$ of the operation times. The aim of this paper is to estimate or (what is better) to calculate the largest independent variations of the operation times $t_i, i \in \tilde{V},$ that do not violate the feasibility and optimality of the line balance ***b*** at hand.

Also, note that at the stage of the design of the assembly line, there may exist a lot of optimal line balances. Using stability analysis, one can select such an optimal line balance, which feasibility and optimality are more stable with respect to possible variations of the operation times $t_i, i \in \tilde{V}$.

Let *B* denote the set of all assignments of operations *V* to stations S_1, S_2, …, S_m (for all possible numbers *m* of the stations: $1 \le m \le n$), which satisfy Condition 1. Subset of set *B* of all operation assignments (line balances) which also satisfy Condition 2 for the given vector $t = (t_1, t_2, ..., t_n)$ of the operation times is denoted by $B(t) = \{\boldsymbol{b}_0, \boldsymbol{b}_1, ..., \boldsymbol{b}_h\}$ where $\boldsymbol{b}_k$, $k \in \{0, 1, ..., h\}$, means the following line balance: $V = V_1^{\boldsymbol{b_k}} \cup V_2^{\boldsymbol{b_k}} \cup ... \cup V_{m_{\boldsymbol{b_k}}}^{\boldsymbol{b_k}}$. Let subset of set $B(t)$ of all the optimal line balances be denoted by $B_{opt}(t)$. Inclusion $\boldsymbol{b} \in B_{opt}(t)$ implies that line balance ***b***: $V = V_1^{\boldsymbol{b}} \cup V_2^{\boldsymbol{b}} \cup ... \cup V_{m_{\boldsymbol{b}}}^{\boldsymbol{b}}$, satisfies Condition 1, Condition 2, and the following optimality condition for the vector $t = (t_1, t_2, ..., t_n)$ of the operation times.

Condition 3: $m_{\boldsymbol{b}} = \min\{ m_{\boldsymbol{b}_k} : \boldsymbol{b}_k \in B(t)\}$.

Since line balance $\boldsymbol{b}$ is contained in the set *B(t)*, we obtain $\boldsymbol{b} = \boldsymbol{b}_r \in B(t)$ for some index $r \in \{0, 1, \ldots, h\}$. However, as a matter of convenience, index *r* will be omitted for the optimal line balance $\boldsymbol{b}$, which stability will be investigated. Note that in both definitions of set *B* and set *B(t),* number *m* of stations is not fixed. Namely, for each line balance $\boldsymbol{b}_k$ from the set *B(t)*, inequalities $m_{\boldsymbol{b}} \le m_{\boldsymbol{b}_k} \le n$ must hold, and number of stations in an operation assignment from set *B* has to belong to set {1, 2, …, *n*}. The main questions under consideration may be formulated as follows. How much can be modified the components of the vector $\widetilde{t}$ simultaneously and independently from each other that the given line balance $\boldsymbol{b}$ remains feasible and optimal? Let $\boldsymbol{R}^n$ (and $\boldsymbol{R}^{\boldsymbol{n}}_+$) denote the space of *n*-dimensional real vectors $t = (t_1, t_2, \ldots, t_n)$ (and the space of *n*-dimensional non-negative real vectors, respectively) with the maximum metric, i.e., distance $d(t,t^*)$ between vector $t \in \boldsymbol{R}^n$ and vector $t^* = (t_1^*, t_2^*, \ldots, t_n^*) \in \boldsymbol{R}^n$ is calculated as follows: $d(t,t^*) = max\{ \left| t_i - t_i^* \right| : i \in V\}$. Here $\left| t_i - t_i^* \right|$ denotes the absolute value of the difference $t_i - t_i^*$. Let line balance $\boldsymbol{b}$ be optimal for the given non-negative real vector $t = (\widetilde{t}, \bar{t}) = (t_1, t_2, \ldots, t_n) \in \boldsymbol{R}^{\boldsymbol{n}}_+$ of the operation times, i.e., $\boldsymbol{b} \in B_{opt}(t)$. For SALBP-1, the formal definition of stability radius of an optimal line balance may be introduced as follows.

Definition 1: *The open ball $O_\rho(\widetilde{t})$ with radius $\rho \in \boldsymbol{R}^{\boldsymbol{1}}_+$ and center $\widetilde{t} \in \boldsymbol{R}_+^{\widetilde{n}}$ in the space $\boldsymbol{R}^{\widetilde{n}}$ is called a stability ball of the line balance $\boldsymbol{b} \in B_{opt}(t)$, if for each vector $t^* = (\widetilde{t}^*, \bar{t})$ of the operation times with $\widetilde{t}^* \in O_\rho(\widetilde{t}) \bigcap \boldsymbol{R}^{\widetilde{n}}_+$ line balance $\boldsymbol{b}$ remains feasible and optimal. The maximal value of the radius ρ of stability ball $O_\rho(\widetilde{t})$ of the line balance $\boldsymbol{b}$ is called stability radius denoted by $\rho_{\boldsymbol{b}}(t)$.*

It should be noted that in Definition 1 vector $\bar{t} = (t_{\widetilde{n}+1}, t_{\widetilde{n}+2}, \ldots, t_n)$ of the processing times of the automated operations and the complete vector $t = (\widetilde{t}, \bar{t}) = (t_1, t_2, \ldots, t_n)$ of the operation times are fixed, while vector $\widetilde{t}^* = (t_1^*, t_2^*, \ldots, t_{\widetilde{n}}^*)$ may vary within the intersection of the open ball $O_\rho(\widetilde{t}) \subset \boldsymbol{R}^n$ with the space $\boldsymbol{R}_+^{\widetilde{n}}$ of the non-negative real vectors. The stability radius $\rho_{\boldsymbol{b}}(t)$ is equal to the minimal upper bound of independent variations ε_i of the processing times t_i of all the manual operations $i \in \widetilde{V}$, which definitely keep the optimality of the line balance $\boldsymbol{b}$, i.e., inclusion $\boldsymbol{b} \in B_{opt}(t^*)$ holds with $t_i^* = \max\{0, t_i - \varepsilon_i\}$ or $t_i^* = t_i + \varepsilon_i$. In the rest of this paper, we survey some recent results proven in [Sotskov, Dolgui, 2001; Sotskov, Dolgui, Portmann, 2006; Sotskov *et al*, 2005] on stability analysis of an optimal line balance for SALBP-1 and for SALBP-2.

Stability Radius of an Optimal Line Balance for SALBP-1

Let $\widetilde{V}_k^{\boldsymbol{b}}$ denote the subset of manual operations of set $V_k^{\boldsymbol{b}}$, and let $\overline{V}_k^{\boldsymbol{b}}$ denote the subset of automated operations of set $V_k^{\boldsymbol{b}}$. For each index $k \in \{1, 2, \ldots, m_{\boldsymbol{b}}\}$, equality $V_k^{\boldsymbol{b}} = \widetilde{V}_k^{\boldsymbol{b}} \cup \overline{V}_k^{\boldsymbol{b}}$ holds. The following remark is useful in stability analysis of an optimal line balance for SALBP-1.

Remark 1: *Let us consider the line balance $\boldsymbol{b} \in B_{opt}(t)$ being in process and the modified vector $t' = (\widetilde{t}', \bar{t})$ of the given operation times. If there exists subset $V_k^{\boldsymbol{b}}$, $k \in \{1, 2, \ldots, m_{\boldsymbol{b}}\}$, in the line balance $\boldsymbol{b}$ such that*

$$\sum_{i \in V_k^{\boldsymbol{b}}} t_i' = 0, \tag{3}$$

we continue to affirm that the line balance $\boldsymbol{b}$ uses $m_{\boldsymbol{b}}$ stations for the modified vector $t' = (\widetilde{t}', \bar{t})$.

We can argue Remark-1 as follows. In spite of the equality (3) valid for the vector $t' = (\tilde{t}', \bar{t})$, station S_k is still exists in the assembly line with line balance $\boldsymbol{b}$. At very least to delete station S_k causes additional cost and additional time for re-engineering the assembly line. Moreover, after deleting station S_k we obtain another line balance, say, $\boldsymbol{b}^* \in B$:

$$V = \bigcup_{i \in \{1,2,\ldots,m_b\}, i \neq k} V_i^{\boldsymbol{b}} = V_1^{\boldsymbol{b}^*} \cup V_2^{\boldsymbol{b}^*} \cup \ldots \cup V_{m_{\boldsymbol{b}^*}}^{\boldsymbol{b}^*}$$

where $m_{\boldsymbol{b}^*} = m_{\boldsymbol{b}} - 1$. Note that, due to the validity of inequality $t_i > 0$ for each automated operation $i \in V \setminus \tilde{V}$, equality (3) is only possible if $V_k^{\boldsymbol{b}} = \tilde{V}_k^{\boldsymbol{b}}$. In [Sotskov, Dolgui, Portmann, 2006], it was proven the following necessary and sufficient conditions for the case when optimality of the line balance $\boldsymbol{b} \in B_{opt}(t)$ is unstable.

Theorem 1: *For the line balance $\boldsymbol{b} \in B_{opt}(t)$ equality $\rho_{\boldsymbol{b}}(t)=0$ holds if and only if there exists a subset $V_k^{\boldsymbol{b}}$, $k \in \{1, 2, \ldots, m_{\boldsymbol{b}}\}$, such that $\tilde{V}_k^{\boldsymbol{b}} \neq \varnothing$ and $t(V_k^{\boldsymbol{b}}) = c$.*

To present a formula for exact value of stability radius $\rho_{\boldsymbol{b}}(t)$ of optimal line balance $\boldsymbol{b} \in B_{opt}(t)$ and lower bound for $\rho_{\boldsymbol{b}}(t)$ we need the following notations:

$$\delta^{\boldsymbol{b}} = min\{\delta_k^{\boldsymbol{b}} : \tilde{V}_k^{\boldsymbol{b}} \neq \varnothing,\ k \in \{1, 2, \ldots, m_{\boldsymbol{b}}\}\}, \tag{4}$$

where

$$\delta_k^{\boldsymbol{b}} = \frac{c - t(V_k^{\boldsymbol{b}})}{\left| \tilde{V}_k^{\boldsymbol{b}} \right|}$$

and value $t(V_k)$ is defined in (2). It is easy to see that testing criterion given in Theorem 1 takes $O(n)$ time. The asymptotic bound $O(n)$ is defined due to calculating station times $t(V_p^{\boldsymbol{b}}), p = 1, 2, \ldots, m_{\boldsymbol{b}}$. Therefore, if station times are included in the input data of the algorithm, then algorithm takes $O(m_{\boldsymbol{b}})$ time.
The following lower bound of stability radius has been obtained within the proof of Theorem 1.

Corollary 1*: If optimality of line balance $\boldsymbol{b} \in B_{opt}(t)$ is stable, then $\rho_{\boldsymbol{b}}(t) \geq min\{\delta^{\boldsymbol{b}}, \Delta^{\boldsymbol{b}}\}$* where

$$\Delta^{\boldsymbol{b}} = \min\{\Delta(V_p^{\boldsymbol{b}^*}) : \boldsymbol{b}^* \in B\} \text{ and } \Delta(V_p^{\boldsymbol{b}^*}) = \frac{t(V_p^{\boldsymbol{b}^*}) - c}{\left| \tilde{V}_p^{\boldsymbol{b}^*} \right|}.$$

Let $\tilde{V}_p^{\boldsymbol{b}(d)} = \{i_1, i_2, \ldots, i_u\}$, where $u = \left| \tilde{V}_p^{\boldsymbol{b}(d)} \right|$, and indices v of operations i_v are assigned in such a way that the following inequalities hold: $t_{i_1} \leq t_{i_2} \leq \ldots \leq t_{i_u}$. We set $t_{i_0} = 0$. Vector $t'' = (\tilde{t}'', \bar{t}) \in \boldsymbol{R}_+^n$ closest to t, for which subset $V_p^{\boldsymbol{b}(d)}$ is feasible (i.e., inequality (1) holds for subset $V_p^{\boldsymbol{b}(d)}$ with vector $t'' = (\tilde{t}'', \bar{t})$ of the operation times), can be obtained if for each operation $i_q \in \tilde{V}_p^{\boldsymbol{b}(d)}$ we set $t''_{i_q} = max\{0, t_j - \hat{\Delta}(V_p^{\boldsymbol{b}(d)})\}$ where j and i_q denotes the same manual operation ($j = i_q$), and value $\hat{\Delta}(V_p^{\boldsymbol{b}(d)})$ is calculated as follows:

$$\hat{\Delta}(V_p^{\boldsymbol{b}(d)}) = max\left\{ \frac{\sum_{i \in V_p^{\boldsymbol{b}(d)}} t_i - c - \sum_{\alpha=0}^{\beta} t_{i_\alpha}}{\left| \tilde{V}_p^{\boldsymbol{b}(d)} \right| - \beta} : \beta = 0, 1, \ldots, \left| \tilde{V}_p^{\boldsymbol{b}(d)} \right| - 1 \right\}$$

where maximum is taken among right-hand fractions calculated for all $\beta = 0, 1, \ldots, \left| \tilde{V}_p^{\boldsymbol{b}(d)} \right| - 1$.

We can define

$$\Delta(\boldsymbol{b}^{(d)}) = max\ \{\ \hat{\Delta}(V_p^{\boldsymbol{b}^{(d)}}) : t(V_p^{\boldsymbol{b}^{(d)}}) > c\ \}$$
$$\hat{\Delta}^{\boldsymbol{b}} = min\{\Delta(\boldsymbol{b}^{(d)}) : \boldsymbol{b}^{(d)} \in B^{(m_{\boldsymbol{b}}-1)}\} \quad (5)$$

In [Sotskov, Dolgui, Portmann, 2006], the following formula for calculating the exact value of stability radius $\rho_{\boldsymbol{b}}(t)$ has been derived.

Theorem 2: *If optimality of line balance $\boldsymbol{b} \in B_{opt}(t)$ is stable, then $\rho_{\boldsymbol{b}}(t) = min\{\delta^{\boldsymbol{b}}, \hat{\Delta}^{\boldsymbol{b}}\}$ with $\delta^{\boldsymbol{b}}$ being defined in (4) and $\hat{\Delta}^{\boldsymbol{b}}$ in (5).*

Let $\lceil a \rceil$ denote the smallest integer greater than or equal to a. Theorem 2 implies the following corollaries.

Corollary 2: *If* $m_b = \left\lceil \frac{t(V)}{c} \right\rceil$, *then* $\rho_{\boldsymbol{b}}(t) \geq min\ \{\ \delta^{\boldsymbol{b}}; \frac{t(V) - c(m_{\boldsymbol{b}} - 1)}{\tilde{n}}\ \}$.

Corollary 3: *If* $m_{\boldsymbol{b}} = \left\lceil \frac{t(V)}{c} \right\rceil$ *and* $\delta^{\boldsymbol{b}} \leq \frac{t(V) - c(m_{\boldsymbol{b}} - 1)}{\tilde{n}}$, *then* $\rho_{\boldsymbol{b}}(t) = \delta^{\boldsymbol{b}}$.

Corollary 4: *If* $\boldsymbol{b} \in B_{opt}(t)$, *then* $\rho_{\boldsymbol{b}}(t) \leq min\ \{\delta^{\boldsymbol{b}}, max_{i \in \tilde{V}}\ t_i\}$.

Corollary 5: *If* $\boldsymbol{b} \in B_{opt}(t)$, *then* $\rho_{\boldsymbol{b}}(t) \leq min\ \{c - max_{i \in \tilde{V}}\ t_j, max_{i \in \tilde{V}}\ t_j\}$.

Corollary 6: *If* $\boldsymbol{b} \in B_{opt}(t)$ and $\boldsymbol{b}^{(d)} \in B^{(m_{\boldsymbol{b}}-1)}$, *then* $\rho_{\boldsymbol{b}}(t) \leq min\{\delta^{\boldsymbol{b}},\ \Delta(\boldsymbol{b}^{(d)})\}$.

Stability of an Optimal Line Balance for SALBP-2

In this section, we consider the Simple Assembly Line Balancing Problem when number *m* of stations is fixed while the cycle time has to be minimized. In other words, we consider SALBP-2: to find an optimal balance of the assembly line for a given number *m* of stations, i.e., to find a feasible assignment of all operations *V* to exactly *m* stations in such a way that the cycle-time *c* is minimal. For SALBP-2, line balance $\boldsymbol{b}_r$ is *optimal* if along with conditions 1 and 2, it has the minimal cycle time. We denote the cycle time for line balance $\boldsymbol{b}_r$ with the vector *t* of operation times as $c(\boldsymbol{b}_r, t)$: $c(\boldsymbol{b}_r, t) = max_{k=1}^{m} \sum_{i \in V_k^{b_r}} t_i$. For SALBP-2, optimality of line balance $\boldsymbol{b}=\boldsymbol{b}_s$ with vector *t* of the operation times may be defined as the following condition.

Condition 4: *$c(\boldsymbol{b}_s, t) = min\{c(b_r, t) : b_r \in B(t)\}$, where $B(t) = \{\boldsymbol{b}_0, \boldsymbol{b}_1, \ldots, \boldsymbol{b}_h\}$ is the set of all line balances.*

For SALBP-2, the formal definition of the stability radius of an optimal line balance may be introduced as follows.

Definition 2: *The closed ball $\underline{O}_\rho(\tilde{t}\,)$ in the space $R^{\tilde{n}}$ with the radius $\rho \in R_+^1$ and the center $\tilde{t} \in R_+^{\tilde{n}}$ is called a stability ball of the line balance $\boldsymbol{b} \in B(t)$, if for each vector $t^* = (\tilde{t}^*, \bar{t}\,)$ of the operation times with $\tilde{t}^* \in \underline{O}_\rho(\tilde{t}) \cup R_+^{\tilde{n}}$ line balance $\boldsymbol{b}$ remains optimal. The maximal value of the radius ρ of a stability ball $\underline{O}_\rho(\tilde{t}\,)$ of the line balance $\boldsymbol{b}$ is called the stability radius denoted by $\underline{\rho}_b(t)$.*

In Definition 2, vector $\bar{t} = (t_{\tilde{n}+1}, t_{\tilde{n}+2}, \ldots, t_n)$ of the automated operation times and vector $t = (\tilde{t}, \bar{t}\,) = (t_1, t_2, \ldots, t_n)$ of all the operation times are fixed, while vector $\tilde{t}^* = (t_1^*, t_2^*, \ldots, t_{\tilde{n}}^*)$ of the manual operation times may vary within the intersection of the closed ball $\underline{O}_\rho(\tilde{t}\,)$ with the space $R_+^{\tilde{n}}$. For each optimal line balance $\boldsymbol{b}_r \in B_{opt}(t)$, we can define a set $W(\boldsymbol{b}_r)$ of all subsets $\tilde{V}_k^{b_r}$, $k \in \{1, 2, \ldots, m\}$, such that $\sum_{i \in V_k^{b_r}} t_i = c(\boldsymbol{b}, t)$. It should be noted that set $W(\boldsymbol{b}_r)$ may include the empty set as its element.

In [Sotskov *et al*, 2005], the following claims have been proven.

Theorem 3: *Let inequality $t_i > 0$ hold for each manual operation $i \in \widetilde{V}$. Then for line balance $\mathbf{b}_s \in B(t)$, equality $\underline{\rho}_{b_s}(t) = 0$ holds if and only if there exists a line balance $\mathbf{b}_r \in B_{opt}(t)$ such that condition $W(\mathbf{b}_s) \subseteq W(\mathbf{b}_r)$ does not hold.*

Corollary 7: *If $B_{opt}(t) = \{\mathbf{b}_s\}$, then $\underline{\rho}_{b_s}(t) > 0$.*

If there exists an index $k \in \{1, 2, \ldots, m\}$ such that

$$\sum_{i \in V_k^{b_0}} t_i < c(\mathbf{b}_0, t), \tag{6}$$

then we set $\lambda(b_0) = \{c(b_0,t) - \max\{\sum_{i \in V_k^{b_0}} t_i : \widetilde{V}_k^{b_0} \notin W(b_0)\}\} \Big/ \widetilde{n}$. Due to (6), the strict inequality $\lambda(b_0) > 0$ must hold. If $\sum_{i \in V_k^{b_0}} t_i = c(\mathbf{b}_0, t)$ for each index $k \in \{1, 2, \ldots, m\}$, then we set $\lambda(b_0) = \min\{t_i : i \in \widetilde{V}\}$. We denote

$\Delta = \min\{\Delta(\mathbf{b}_s): \mathbf{b}_s \in B \setminus B(t)\}$, where $\Delta(\mathbf{b}_s) = \dfrac{c(b_s,t) - c(b_0,t)}{\widetilde{n}}$. Theorem 3 implies the following claim.

Corollary 8: *If $\underline{\rho}_{b_s}(t) > 0$, then $\underline{\rho}_{b_s}(t) \geq \min\{\Delta, \lambda(b_0)\}$.*

The problem of calculating exact value of stability radius $\underline{\rho}_b(t)$ is close to calculating stability radius of the optimal schedule for the makespan criterion (see [Sotskov, 1991; Sotskov, Tanaev, Werner, 1998]).

Conclusion and Recommendations

In this paper, the known results on stability analysis of an optimal line balance are presented. To this end, we used the notion of stability radius, which is similar to the stability radius of an optimal schedule introduced in [Sotskov, 1991] for scheduling problems. (A survey of known results on stability analysis in machine scheduling is given in [Sotskov, Tanaev, Werner, 1998].) If stability radius of line balance $\boldsymbol{b}$ is strictly positive, then any independent changes of the operation times t_j, $j \in \widetilde{V}$, within the ball with this radius, definitely keep the optimality of line balance $\boldsymbol{b}$. On the other hand, if stability radius of $\boldsymbol{b}$ is equal to zero (i.e., if the optimality of line balance $\boldsymbol{b}$ is *unstable*, see Theorems 1 and 3), then some even small changes of the processing times of all or a portion of the manual operations may deprive the optimality of line balance $\boldsymbol{b}$. It is worth noting that all conditions presented in this paper (except Theorems 2 and 3 and Corollaries 1 and 8) may be tested in polynomial time, which is important for real-world assembly lines with large numbers of operations and stations. Moreover, for exact value of stability radius, feasibility of the line balance $\boldsymbol{b}$, which is defined by the value $\delta^{\boldsymbol{b}}$, may be tested in polynomial time even in Theorem 2.

In practice, the tendency at the design stage must be to find optimal line balance for which stability is as much as possible. Of course, the common objective is to assign to each station a set of operations with roughly the same total operation time (see [Bukchin, Tzur, 2000; Erel, Sarin, 1998; Lee, Johnson, 1991; Sarin, Erel, Dar-El, 1999]). However, due to the above results, we have to defer stations with manual operations and stations without manual operations. Theorem 1 shows that for the station with manual operations it is desirable to have some slack between cycle time and station time. The larger this slack is, the large stability radius of the line balance may be. On the other hand, for the stations with only automated operations, such a slack may be as small as possible, which gives the possibility to increase slacks for stations loaded by manual operations. Since the stability radius $\rho_{\mathbf{b}}(t)$ cannot be larger than $c/2$, one has to pay special attention to the manual operations with possible variations of processing times more than $c/2$ (such operations may cause instability of optimality of the line balance at hand). If it is possible at the design stage, such an operation has to be divided into shorter manual operations.

If line balance will be used for a long time for assembling the same finished product, it is desirable at the design stage, to construct several optimal line balances, and select among them the one with the best stability characteristics. So, it is useful to develop algorithms, which construct a set of optimal line balances (instead of only one optimal line balance), in order to carry out a stability analysis for them. Or, better yet, it is useful to

include in the branch-and-bound or other algorithms used for SALBP-1 and SALBP-2 specific rules in order to construct optimal line balance with larger stability radius. In a concrete study, the set $\widetilde{V}$ of manual operations can be reduced (e.g., only critical manual operations may be considered) or, on the contrary, set $\widetilde{V}$ may be completed by some unstable automated operations. By changing the set of operations with variable times, the designer of the assembly line can study the influence of different operations on stability of optimality and feasibility of line balances.

In [Sotskov, Dolgui, 2001], slightly different definitions of stability radius and stability ball of an optimal line balance have been used for SALBP-1. Namely, it was assumed that $O_{\rho_{\boldsymbol{b}}(t)} \subset \boldsymbol{R}_{+}^{\widetilde{n}}$. Therefore in that paper, generally smaller stability radii were obtained (in particular, it cannot be greater than $min\{ \mathrm{t_i} : i \in \widetilde{V}\}$). The above Definition 1 seems to be more appropriate for practical assembly lines.

Acknowledgements

This research was supported by **ISTC (Project B-986)**. The author would like to thank **Prof. Alexandre Dolgui**, **Prof. Marie-Claude Portmann**, and **Prof. Frank Werner** for joint research in stability analysis.

Bibliography

[Baybars, 1986] I. Baybars, A survey of exact algorithms for the simple assembly line balancing problem, *Management Science* 32 (8) (1986) 909-932.

[Bukchin, Tzur, 2000] J. Bukchin, M. Tzur, Design of flexible assembly line to minimize equipment cost, *IIE Transactions* 32 (2000) 585-598.

[Erel, Sarin, 1998] E. Erel, S.C. Sarin, A survey of the assembly line balancing procedures, *Production Planning & Control* 9 (5) (1998) 414-434.

[Garey, Johnson, 1979] M.R. Garey, D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness.* Freeman, San Francisco, USA, 1979.

[Lee, Johnson, 1991] H.F. Lee, R.V. Johnson, A line-balancing strategy for designing flexible assembly systems, *The International Journal of Flexible Manufacturing Systems* 3 (1991) 91-120.

[Sarin, Erel, Dar-El, 1999] S.C. Sarin, E. Erel, E.M. Dar-El, A methodology for solving single-model, stochastic assembly line balancing problem, *OMEGA – International Journal of Management Science* 27 (1999) 525-535.

[Scholl, 1999] A. Scholl, *Balancing and Sequencing of Assembly Lines*, Heidelberg: Physica-Verlag, Germany,1999.

[Sotskov, 1991] Yu.N. Sotskov, Stability of an optimal schedule, *European Journal of Operational Research* 55 (1991) 91-102.

[Sotskov, Dolgui, 2001] Yu.N. Sotskov, A. Dolgui, Stability radius of the optimal assembly line balance with fixed cycle time. In: *Proceedings of the IEEE Conference ETFA'2001* (2002) 623-628.

[Sotskov, Dolgui, Portmann, 2006] Yu.N. Sotskov, A. Dolgui, M-C. Portmann, Stability of an optimal balance for an assembly line with fixed cycle time, *European Journal of Operational Research* 168 (2006) 783-797.

[Sotskov, *et al.* 2005] Yu.N. Sotskov, A. Dolgui, N. Sotskova, F. Werner, Stability of optimal line balance with given station set. In: *Supply Chain Optimisation* (2005) 135-149.

[Sotskov, Tanaev, Werner, 1998] Yu.N. Sotskov, V.S. Tanaev, F. Werner, Stability radius of an optimal schedule: A survey and recent developments, Chapter in: G. Yu (Ed.) *"Industrial Applications of Combinatorial Optimization"* 16, Kluwer Academic Publishers, Boston, MA, USA, 1998, 72-108.

Author's Information

Yuri N. Sotskov – *Professor, DSc, United Institute of Informatics Problems of National Academy of Sciences of Belarus, Surganova str., 6, Minsk, Belarus; e-mail: sotskov@newman.bas-net.by*

ОЦЕНКА РЕАЛИЗУЕМОСТИ АЛГОРИТМОВ ДЕЯТЕЛЬНОСТИ ЭКИПАЖА АНТРОПОЦЕНТРИЧЕСКОГО ОБЪЕКТА ПРИ РАЗРАБОТКЕ СПЕЦИФИКАЦИЙ ЕГО БОРТОВЫХ АЛГОРИТМОВ

Борис Е. Федунов

Аннотация. *При проектировании спецификаций бортовых алгоритмов системообразующего ядра антропоцентрического объекта (алгоритмов, реализуемые на бортовых цифровых вычислительных машинах, алгоритмы деятельности экипажа (АДЭ)) возникает задача оценки возможности выполнения экипажем определившихся АДЭ. Для решения этой задачи вся деятельность экипажа структурирована по видам его работ (принятие решений, участие в системах слежения, реализация решений) и проведена количественная оценка временных затрат экипажа на их выполнение. Предложен критерий оценки реализуемости АДЭ экипажем.*

Ключевые слова*: алгоритмы деятельности экипажа (АДЭ): решения, операции слежения, диспетчеризация; оценки времени выполнения АДЭ, критерии реализуемости*

Введение

При проектировании спецификаций бортовых алгоритмов системообразующего ядра антропоцентрического объекта (Антр/объекта) инженерами – разработчиками определяются состав и облик алгоритмов, реализуемых на бортовых цифровых вычислительных машинах (БЦВМ-алгоритмы), и алгоритмов реализуемых оператором, членом экипажа Антр/объекта (алгоритмы деятельности экипажа (АДЭ)). При этом возникают задачи:

- разработать АДЭ с глубиной, позволяющей оценить их реализуемость (выполнимость) оператором;
- разработать критерии оценки реализуемости АДЭ;
- создать компьютерную систему, позволяющую инженеру-практику проводить оценку реализуемости конкретного состава АДЭ.

Проектирование спецификаций бортовых алгоритмов в настоящее время опирается на использование математической модели (ММ) Антр/объекта [1], представляющей его внутренний семантический облик через три глобальных уровня управления (ГлУУ), а весь объем работы Антр/объекта - через сеансы функционирования. На Антр/объекте различают уровень оперативного целеполагания (первый глобальный уровень управления I ГлУУ), уровень определения рационального способа достижения оперативно поставленной цели (второй глобальный уровень управления II ГлУУ), уровень реализации принятого способа достижения оперативно поставленной цели (третий глобальный уровень управления III ГлУУ).

I ГлУУ и II ГлУУ составляют семантическую суть системообразующего ядра Антр/объекта.

Каждый сеанс функционирования в этой ММ представляется семантической сетью типовых ситуаций (ТС), выбираемых из единого множества ТС. В свою очередь ТС представима через семантическую сеть проблемных субситуаций (ПрС/С).

Структура деятельности оператора в техническом антропоцентрическом объекте

При системном проектировании спецификаций алгоритмов бортового интеллекта деятельность оператора Антр/объекта представляется через следующие составляющие. Оператор на борту Антр/объекта в рамках активизированной концептуальной модели поведения выполняет следующие типы работ:

а) принимает решения по оперативно возникающей проблеме,

б) реализует эти решения,

в) участвует в различных операциях слежения как элемент следящей системы [3 - 7]. В процессе своей работы он может неоднократно менять свою концептуальную модель поведения.

Вся необходимая для деятельности оператора информация представляется ему на индикаторах информационно управляющего поля (ИУП) кабины экипажа и/или сообщается ему через кабинные речевые информаторы. Реализация решений и участие в операциях слежения осуществляется экипажем через органы управления ИУП.

Исходной информацией для проектирования АДЭ служат:

- естественно языковые документы «Логика работы системы «экипаж-бортовая аппаратура»», структурированные по членам экипажа (операторам) Антр/объекта, по ГлУУ, ТС и ПрС/С в них,
- описания информационно управляющего поля (ИУП) кабины каждого члена экипажа.

Заметим, что в процессе реального проектирования БЦВМ-алгоритмы и АДЭ разрабатываются по каждой ПрС/С каждой ТС одновременно и согласуются по обмену информацией между ними.

При проектировании каждое решение оператора относится к одному из следующих типов: π-решения (перцептивно-опознавательные), ρ-решения (речемыслительные) и π-ρ-решения (эвристические) [2-7].

Каждое *π-решений* характеризуется мгновенной реакцией оператора на определенный стимул-сигнал. Временные затраты оператора на принятие такого решения состоят только из временных затрат на обнаружение и опознание соответствующего стимул-сигнала. Такие решения представляюся в спецификациях бортовых алгоритмов:

- составом информации или речевым сообщением (стимул-сигналом), которые необходимы оператору для принятия решения;
- выходной информацией: составом и последовательностью ручных операций, необходимых для реализации оператором принятого решения.

При оценке времени на восприятие информация оператором и на ее осмысливание она представляется через набор оперативных единиц восприятия (ОЕВ), через состав и продолжительность речевого сообщения, которое передается оператору кабинным речевым информатором и которое используется при принятии этого решения. Необходимые для этого оценки времени восприятия каждой ОЕВ приведены в [3 – 5, 7], продолжительность речевого сообщения оценивается экспериментально.

Каждое ρ *-решение* может быть представлено через последовательность элементарных актов выработки решения, опирающихся на соответствующий состав информационных символов на индикаторах ИУП. Такие решения характеризуются в спецификациях бортовых алгоритмов:

- входной информацией, включающей в себя состав информации на ИУП кабины (описывается составом ОЕВ), по которой оператор должен принимать это решение; составом и продолжительностью речевых сообщений, которые передается оператору кабинным речевым информатором и которые используется при принятии этого решения.
- структурой решения, составом и последовательностью элементарных актов выработки решения (ЭАВР), описываемых через индикационную символику информационных кадров кабинных индикаторов. Необходимые для этого оценки времени восприятия каждой ОЕВ и реализации каждого ЭАВР приведены в [3 – 5, 7], продолжительность каждого речевого сообщения оценивается экспериментально.
- выходной информацией, представляемой составом и последовательностью ручных операций, необходимых для реализации принятого решения.

Каждое π-*ρ-решение* является эвристическим. Такие решения характеризуются в спецификациях бортовых алгоритмов:

- входной информацией, представляемой составом информации на ИУП кабины, по которой оператор должен принимать это решение; составом и продолжительностью речевых сообщений, которые передаются оператору кабинным речевым информатором и которые используется при принятии этого решения;
- временем на принятие этого решения, получаемом для каждого π-ρ-решения при имитационном моделировании с профессиональными операторами проектируемого Антр/объекта. В таких экспериментах оценивается на только некоторое усредненное по операторам время на принятия

такого решения, но и полнота предъявляемой для этого оператору информации и, что не менее важно, эффективность принятого решения;

- выходной информацией, характеризуемой составом и последовательностью ручных операций, необходимых для реализации принятого решения.

Алгоритмы деятельности оператора по принятию решения обозначаются в спецификациях аббревиатурой АДЭ с указанием типа принимаемого решения (π-решение, ρ-решение или π-ρ-решение).

Алгоритмы деятельности оператора по реализации принятого решения обозначаются в спецификациях аббревиатурой АДЭ-Р. При наличии проработанной топологии ИУП (есть чертежи пространственного размещение органов управления и указаны их конструктивное оформление) каждая ручная операция по использованию соответствующего органа управления характеризуется своим временем, полученным в лабораторном эксперименте или рассчитанном по [3 – 5,7]. При отсутствии такой информации об органах управления ИУП каждая ручная операция характеризуются единым для всех «универсальным» временем выполнения.

Алгоритмы деятельности оператора по участию его в *процессах слежения* на этапе разработки спецификаций бортовых алгоритмов описываются в достаточно общем виде. Для оценки времени, которое тратит оператор на процесс слежения, сделаны следующие допущения. При выполнении операций слежения предполагается, что оператор работает в дискретно – непрерывном режиме, отвлекаясь от операции слежения на время принятия и реализацию решения (решений). После отвлечения оператор опять возвращается к процессу слежения и устраняет ошибку слежения, накопившуюся за время его отвлечения. Моменты отвлечения оператора на операции слежения не могут разрывать процесс принятия решения и процесс его реализации. Процесс слежения так же не должен вклиниваться между принятым решением и процессом его реализации.

При этом на рассматриваемом этапе проектирования алгоритмов бортового интеллекта состав и описание динамических звеньев следящей системы, звеном которой становится оператор, как правило, отсутствуют. Имеется только представление о зависимости $\tau_{слеж} = f(\tau_{отв})$ времени отработки оператором ($\tau_{слеж}$) ошибки слежения, накопившейся за время его отвлечения от процесса слежения, от времени этого отвлечения ($\tau_{отв}$). Среди АДЭ могут быть несколько типов слежения, каждый из которых характеризуется своей зависимостью. Процессы слежения могут быть вложенными друг в друга.

И, наконец, все названные элементы деятельности оператора объединены *концептуальной моделью поведения оператора*, оперативная смена которой оператором в процессе его деятельности требует определенной затраты времени. Это время характеризуется единой величиной для всех концептуальных моделей.

Граф решений оператора

Все типы работ оператора, упорядоченные по причинно следственному отношению, представляются графом решений оператора (ГРО). Каждая вершина ГРО принадлежать к одному из установленных типов и имеет на графе свое оформление (см. табл.1).

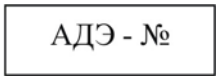
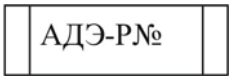
Вершины графа «*АДЭ-тип решения*» характеризуются временем, затрачиваемым оператором на восприятие с ИУП необходимой информации, ее осмысливание, выработку решений. Вершины *АДЭ-Р (реализация решения*) характеризуются временем, затрачиваемым оператором на выполнение ручных операций. Вершина, соответствующая *смене концептуальной модели* поведения оператора, характеризуется временем смены этой модели. Между вершинами существуют причинно – следственные связи, на графе изображаемые дугами. Деятельность оператора начинаться с вершины «Начало слежения», расположенной в корне графа, и разворачивается от корневой вершины к конечной вершине «Конец слежения». Могут быть вложенные операции слежения, размещаемые между вершинами «Начало слежения» и «Конец слежения» другого слежения. При этом получается граф с древовидной структурой (одно начало, множество концов).

На ГРО меткой «+»отмечаются места возможного (по семантике) отвлечения оператора от процесса слежения.

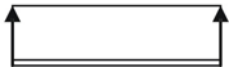
Каждая вершина типа *АДЭ-тип решения* характеризуется составом символов на индикаторах ИУП и составом речевых сообщений, по которым оператор принимает решение, соответствующее этой вершине.

Вершина, обозначающая начало слежения, должна сопровождаться таблицей или графиком $\tau_{слеж} = f(\tau_{отв})$ зависимости времени отработки оператором накопившейся ошибки слежения $\tau_{слеж}$ от интервала времени, в течение которого оператор отвлекался от процесса слежения $\tau_{отв}$.

Табл.1 Типы вершин ГРО.

	Прямоугольниками обозначаются алгоритмы принятия решений с указанием типа решения(π-решение, ρ-решение, π-ρ – решение). Каждое АДЭ - решение обязательно сопровождается фрагментом кадра информации на индикаторах ИУП (возможна речевая информация, по которой оно принимается).
	Вершина, обозначающая алгоритмы реализации оператором принятого решения с соответствующим номером. Каждое АДЭ – реализация обязательно сопровождается указанием на используемые элементы управления и их размещения (если уже есть топология ИУП) или указанием на обобщенное их описание.
	Место смены концептуальной модели поведения оператора

Ниже приведены вершины, не вносящие временных затрат, в общую временную оценку.

	Вершина, обозначающая ветвление в ГРО. Содержит логическое условие для действий оператора. Имеет два выхода: истина и ложь.
	Вершины, обозначающие алгоритмы участия оператора в работе следящей системы, соответственно начало и конец слежения.
	
(+)	Метка ставится на дугах графа в тех местах, где, по мнению разработчика графа, оператор может отвлечься на реализацию операции слежения.

Каждый элемент деятельности оператора оценивается временными затратами оператора на его выполнение [3-7]. Поэтому ГРО для каждой ПрС/С можно оценить по временным затратам оператора на его выполнение (потребное время на реализацию АДЭ, представленных в ГРО).

Обычно для каждой ПрС/С (а иногда даже и фрагмента ПрС/С) существуют естественные ограничения ее продолжительности (располагаемое время). Соотнесение располагаемого и потребного времен позволяет судить о возможности реализации оператором ГРО в заданной обстановке.

Для практической работы инженеров прикладников создана компьютерная система «ГРО-оценка» [8]. В базу данных этой системы заложены количественные оценки времен, потребного оператору на выполнение соответствующего фрагмента.

Методика оценки реализуемости ГРО

Подлежащей оценке ГРО должен содержать следующую информацию:

а) по каждому решению указан его тип (π-решение, ρ-решение):

- для π-решения: символы на индикации, используемые для принятия π-решения, и входные речевые сообщения с указанием продолжительности каждого сообщения (входная информация), число ОЕВ, число ручных операций для реализации решения;

– для ρ-решения: символы на индикации, используемые для принятия ρ-решения, и входные речевые сообщения с указанием продолжительности каждого сообщения (входная информация), число ОЕВ, число и вид ЭАВР, число ручных операций для реализации решения;

б) метки по смене концептуальной модели поведения оператора;

в) последовательности решений, которые должны выполняться “одновременно” с операцией слежения с указанием вида зависимости $\tau_{слеж} = f(\tau_{отв})$;

г) предельно допустимое время выполнения всего графа ($T_{граф}$) и отдельных его фрагментов ($T_{фраг}$).

Оценку выполнимости ГРО следует проводить в следующей последовательности.

Первый шаг. Используя результаты работ [3 - 5, 7] рассчитать время выполнения и реализации τ_i каждого решения ГРО. По ветвям ГРО составить последовательности $\{\tau_i\}$.

Второй шаг. По зависимостям $\tau_{слеж} = f(\tau_{отв})$ рассчитать время $\Sigma(\tau_{отр})$, затраченное оператором на процесс отслеживания ошибок слежения при оптимальном распределении интервалов слежения.

Третий шаг. Для фрагментов ГРО, для которых указаны максимально допустимые времена выполнения фрагмента $T_{фраг}$, сравнить с временем реализации фрагмента $T_{реал\text{-}фраг}$:

$$T_{реал-фраг} = \sum \tau_i + T_{отр\Sigma} + 3\sqrt{\sum \sigma_i^2 + (0.3\tau_{отр\Sigma})^2} >< T_{фраг},$$

где

τ_i – времена выполнения решений, входящих в исследуемый фрагмент,

$\tau_{отр\Sigma} = \Sigma(\tau_{отр.})$ – время, затраченное оператором на процессы слежения, которые оператор выполнял в этом фрагменте (при оптимальном расположении интервалов слежения);

$\sum \sigma_i^2$ - сумма квадратов среднеквадратических отклонений АДЭ, входящих в рассматриваемый фрагмент ГРО;

$0.3\tau_{отр\Sigma}$- среднеквадратическое отклонение суммарного времени слежения.

Оценку выполнимости оператором работ по всему ГРО производим путем сравнения максимального времени реализации фрагмента по всем фрагментам ГРО (max $T_{реал\text{-}фраг}$) с $T_{графа.}$

При выполнении всех этих неравенств ГРО считается реализуем. В противном случае требуется либо перепроектировать спецификации бортовых алгоритмов этой ПрС/С либо увеличение $T_{фраг}$, $T_{графа}$ (ограничение сложности внешней обстановки, в которой будет работать проектируемый Антр/объект).

Пример оценки реализуемости АДЭ

Пусть дан фрагмент ГРО (рис.1) с традиционным представлением на нем информации, необходимой для оценки его выполнимости оператором.

Пусть этот фрагмент ГРО будет выполняться в некоторой, уже активизированной, концептуальной модели деятельности, оператором, который будет работать в номинальной рабочей среде [3].

На фрагменте ГРО указываются (рис.1):

а) алгоритмы принятия решения (АДЭ-1, АДЭ-2) с указанием типа решения (ρ-решения). Алгоритмы обозначены простыми прямоугольниками. Возле каждого алгоритма указан фрагмент кадра индикации, содержащий символы, по которым оператор должен принимать решение;

б) место смены концептуальной модели поведения оператора, обозначенное прямоугольником с двойным окаймлением;

в) алгоритмы реализации оператором принятого решения (АДЭ-Р1), обозначенные прямоугольником с двойными вертикальными сторонами;

г) алгоритмы участия оператора в работе следящей системы (АДЭ-С1), повторенные дважды: в начале ветки ГРО, где оператор должен включиться в работу следящей системы (прямоугольник с двойными горизонтальными сторонами и с вертикальными сторонами в виде стрелок, направленных вниз), и в конце ветки ГРО, где оператор прекращает работу в этой следящей системе (такой же прямоугольник, но вертикальные стрелки направлены вверх);

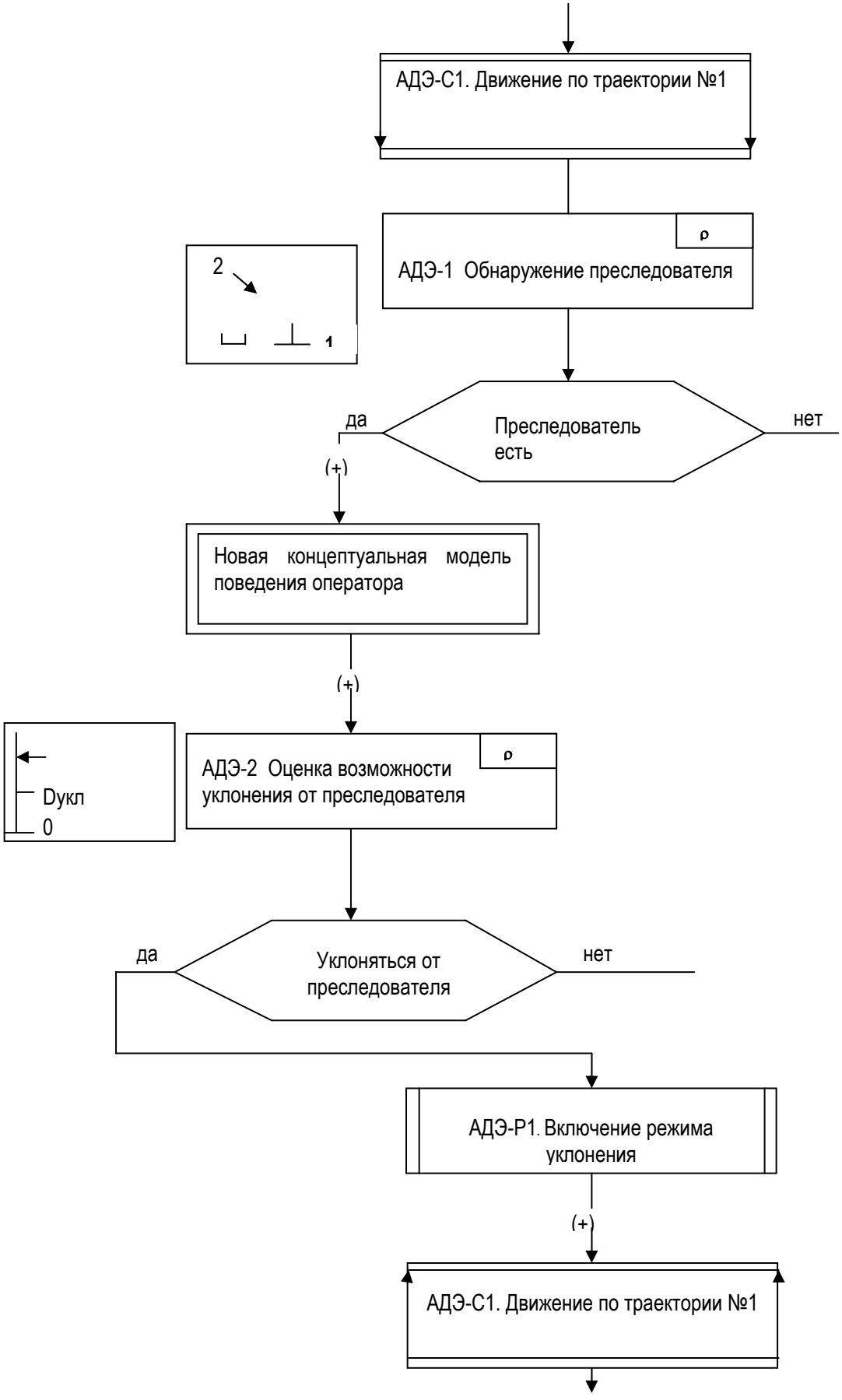

АДЭ-С1. Движение по траектории №1
ρ
АДЭ-1 Обнаружение преследователя
2
1
да
Преследователь есть
нет
(+)
Новая концептуальная модель поведения оператора
(+)
Dукл
0
ρ
АДЭ-2 Оценка возможности уклонения от преследователя
да
Уклоняться от преследователя
нет
АДЭ-Р1. Включение режима уклонения
(+)
АДЭ-С1. Движение по траектории №1

Рис. 1

д) метками (+) указаны места, где оператор может участвовать в работе следящей системы (участвовать в слежении) при реализации дискретно-непрерывного слежения;

е) по каждому алгоритму деятельности в ГРО представляется формуляр, в котором дается информация, необходимая для оценки его реализуемости оператором (см. ниже по каждому алгоритму деятельности).

Проведем оценки временных затрат оператора по отдельным составляющим его деятельности, используя количественные оценки их элементов (ЭАВР, ОЕВ) из [3-7].

АДЭ-1. Обнаружение преследователя (см. рис.1).

Речемыслительный алгоритм (ρ -решение). Вся необходимая информация представлена на фрагменте индикационного кадра в левой верхней части рис.1.

Этап обнаружения и восприятия необходимых сигналов. ОЕВ: наличие на индикаторе ИУП символа «2» со стрелкой (верхняя левая часть рисунка).

Рефлекторные раздражения: $t_р = 0.18с; \quad \sigma_р = 0.2$

Восприятие информации: $t_{лп} = 0.1с; \quad \sigma_{лп} = 0.03$ (латентный период),

$t_ф = 0.3с; \quad \sigma_ф = 0.09$ (ОЕВ – условный знак в сложной информационной модели).

Общее время ($\tau_{обн\text{-}вос}$) на весь процесс «обнаружение восприятие»

$$\tau_{обн-восп} = 0.18 + 0.1 + 0.3 = 0.58c\,;\ \sigma_{обн-восп} = \sqrt{0.05^2 + 0.03^2 + 0.09^2} = 0.11$$

Этап выработки решения.

ЭАВР1: сравнение длины отрезка, связывающего символ «1» и «2» (на индикаторе не показывается) с масштабным отрезком, представленным в левом нижнем углу индикатора)

$$\tau = 2.5с; \qquad \sigma = 0.2.$$

ЭАВР2. Сравнение длины отрезка, полученной в ЭАВР1, с некоторым числом из активизированной концептуальной модели поведения оператора (табл 2, п. 4 “или”)

$$\tau = 0.29с; \qquad \sigma = 0.11$$

ЭАВР3. Упорядочение символов “2” и “1” по углу разворота символа “2” по отношению к символу “1” со сравнением угла разворота с углом из концептуальной модели (упорядочение по одному признаку”)

$$\tau = 0.7с; \qquad \sigma = 0.1$$

ЭАВР4. Решение комбинаторной логической задачи по результатам ЭАВР2 и ЭАВР3. (число условий 2)

$$\tau = 4с; \quad \sigma = 1.2$$

Общее время ($\tau_{\text{выр/р}}$) на выработку ρ-решения:

$$(\tau_{\text{выр/р}}) = 2.5 + 0.29 + 0.7 + 4 = 7.49с$$

$$\sigma_{выр\,/\,р} = \sqrt{0.2^2 + 0.11^2 + 0.1^2 + 1.2^2} = \sqrt{0.04 + 0.01 + 0.01 + 1.44} = \sqrt{1.5} = 1.22$$

Этап реализации принятого решения: ручных операций на реализации решения в АДЭ-1 нет.

Общее время на принятие решения $\tau_{\text{АДЭ-1}}$ в АДЭ-1:

$$\tau_{\text{АДЭ-1}} = \tau_{\text{обн-восп}} + \tau_{\text{выр/р}} = 0.29 + 7.49 = 7.78с.$$

Среднеквадратическое отклонение $\tau_{\text{АДЭ-1}}$

$$\sigma_{АДЭ-1} = \sqrt{\sigma^2_{обн-восп} + \sigma^2_{выр\,/\,р}} = \sqrt{0.11^2 + 1.22^2} = \sqrt{1.51} = 1.22c$$

АДЭ-2. Оценка возможности уклонения от преследователя.

Речемыслительный алгоритм (ρ-решение). Вся необходимая информация представлена на фрагменте индикационного кадра в левой центральной части рис. 1.

Этап обнаружения и восприятия необходимых сигналов.

ОЕВ: наличие на индикаторе стрелки и отметки с формуляром $D_{укл}$.

По структуре и каналу восприятия (зрительная) этот этап полностью идентичный такому же этапу в АДЭ-1.

Воспользовавшись проведенными там расчетами, получим следующие временные затраты этапа:

$$\tau_{\text{обн-восп}} = 0.58\text{c}; \quad \sigma_{\text{обн-восп}} = 0.11\text{c}.$$

Этап выработки решения.

ЭАВР1. Сравнение длин двух отрезков: отрезка от стрелки до начала шкалы (точка "0") с отрезком от отметки $D_{укл}$ до начала шкалы (табл.2, п.2. «Упорядочение геометрических фигур по одному признаку»)

$$\tau = 0.7\text{c}; \quad \sigma = 0.1\text{c}$$

Этап реализации принятого решения (АДЭ-Р1): одна ручная операция

$$\tau_{\text{руч}} = 0.5\text{c}; \quad \sigma_{\text{руч}} = 0.15\text{c}.$$

Общее время на принятие и реализацию решения $\tau_{\text{АДЭ-2}}$ в АДЭ-2:

$$\tau_{\text{АДЭ-2}} = \tau_{\text{обн-восп}} + \tau_{\text{выр/р}} + \tau_{\text{руч}} = 0.58 + 0.7 + 0.5 = 1.78$$

$$\sigma_{АДЭ-2} = \sqrt{0.11^2 + 0.1^2 + 0.15^2} = \sqrt{0.012 + 0.010 + 0.023} = \sqrt{0.045} = 0.21$$

На ГРО символом (+) указаны точки возможного размещения операции слежения. По результатам расчетов времен принятия решения мы имеем следующую последовательность отрезков времени, между которыми можно размещать участки слежения.

$$\tau_{\text{АДЭ-1}} = 7.78; \quad \tau_{\text{конц.мод.}} = 1.2; \quad \tau_{\text{АДЭ-2}} = 1.78$$

Пусть экспериментальная зависимость $\tau_{слеж} = f(\tau_{отв})$ и оптимальное размещение участков слежения таковы, что время, потраченное оператором на процесс слежения составляет $\tau_{\text{ОТР}\Sigma}$ = 13.65 с

Среднеквадратическое отклонение этого времени рассчитываем с использованием рекомендуемой зависимости $\sigma_{\text{ОТР}\Sigma} = 0.3\,\tau_{\text{ОТР}\Sigma}$.

Тогда общее время, затраченное оператором на выполнение рассматриваемого фрагмента ГРО, составит

$$\tau_{\text{фраг}} = \tau_{\text{АДЭ-1}} + \tau_{\text{конц.мод.}} + \tau_{\text{АДЭ-2}} + \tau_{\text{ОТР}\Sigma} = 7.78 + 1.2 + 1.78 + 13.65 = 24.41 \text{ с}.$$

Среднеквадратическое отклонение общего времени:

$$\sigma_{фраг} = \sqrt{1.22^2 + 0.21^2 + 0.2^2 + (13.65 * 0.3)^2} = 4.3c$$

При наличии ограничения $T_{фраг}$ времени выполнения этого фрагмента ГРО необходимо проверить выполнение неравенства

$$T_{\text{реал-фраг}} \geq \tau_{\text{фраг}} + 3\sigma_{\text{фраг}}$$

Выводы

Разработанная математическая модель работы оператора антропоцентрического объекта и методика оценки ГРО ориентирована на использование ее на начальных этапах системного проектирования бортового алгоритмического и индикационного обеспечения (АиИО) функционирования антропоцентрического объекта. На этой стадии разработки бортовых алгоритмов еще нет полной и достоверной информации о компоновке кабины экипажа и размещении информации на кабинных индикаторах; о дифференциальных уравнениях, описывающих звенья следящих систем, в контуры которых включен оператор; о временных затратах эвристических решений оператора.

Методика позволяет в «реальном» времени (по меркам этапа системного проектирования алгоритмов бортового интеллекта) по доступной на этом этапе информации о проектируемом объекте провести предварительную оценку реализации спроектированного ГРО и внести необходимые корректировки в проектируемое АиИО, снижая тем самым технический риск его реализации на последующих этапах проектирования антропоцентрического объекта.

На ее основе такого подхода создана компьютерная система «ГРО-оценка» [8], включающая в себя составленную по [3-7] базу данных по оценке времен выполнения отдельных элементов составляющих АДЭ. Система «ГРО-оценка» позволяет инженерам практикам проводить оценки реализуемости ГРО при реальном проектировании бортового алгоритмического и индикационного обеспечения, не обращаясь к соответствующей литературе. Она позволяет упорядочить и автоматизировать процесс оценки реализуемости. Она обладает удобным интерфейсом для пользователя и достаточным (для решения задач такого уровня) быстродействием.

Система «ГРО-оценка» ориентирована как на ручной ввод данных, так и на ввод данных из компьютерной системы «Борт» [9], поддерживающую процесс проектирования спецификаций бортового АиИО.

Литература

1. Федунов Б.Е. Конструктивная семантика антропоцентрических систем для разработки и анализа спецификаций алгоритмов бортового интеллекта.// Изв. РАН. ТиСУ. 1998. №5
2. Федунов Б.Е. Методика оценки реализуемости графа решений оператора антропоцентрического объекта при разработке алгоритмов бортового интеллекта. 2002г., Москва, ГосНИИАС.
3. Основы инженерной психологии / Под ред. Б.Ф. Ломова. М.: Высш. школа, 1977г.
4. Введение в эргономику / Под ред. В.П. Зинченко. М.: Сов. радио, 1974г.
5. Зараковский Г.М. Психологический анализ трудовой деятельности. М.: Наука, 1967г.
6. Цибулевский И.Е. Человек как звено следящей системы. М.: Наука, 1981г.
7. Справочник по инженерной психологии / Под ред. Б.Ф. Ломова. М.: Машиностроение, 1982г.
8. Абрамов А.П., Выдрук Д.Г., Федунов Б.Е. Компьютерная система оценки реализуемости алгоритмов деятельности экипажа. // Изв.РАН. ТиСУ. 2006. №.3
9. Кондрикова Т.А. Федунов Б.Е. «Борт» - компьютерная система поддержки процесса проектирования спецификаций бортового интеллекта // Изв. РАН. ТиСУ. 1999. №3.

Информация об авторе

Boris Evgenjevich Fedunov *– State Scientific Research Institute for Aviation System (GosNIIAS), ul. Viktorenko 7, Moscow, 125319 Russia; e-mail boris_fed@gosniias.ru*

INSTANTANEOUS DATABASE ACCESS

Guy Francis, Mark Lishman, Vladimir Lovitskii, Michael Thrasher, David Traynor

Abstract: *The biggest threat to any business is a lack of timely and accurate information. Without all the facts, businesses are pressured to make critical decisions and assess risks and opportunities based largely on guesswork, sometimes resulting in financial losses and missed opportunities. The meteoric rise of Databases (DB) appears to confirm the adage that "information is power", but the stark reality is that information is useless if one has no way to find what one needs to know. It is more accurate perhaps to state that, "the ability to find information is power". In this paper we show how **I**nstantaneous **D**atabase **A**ccess **S**ystem (**IDAS**) can make a crucial difference by pulling data together and allowing users to summarise information quickly from all areas of a business organisation.*

Keywords: *data mining, natural language, parsing, SQL-query and production rules*

Introduction

1. The rapid advance of computer technology, particularly, the explosive growth of databases (DB), has resulted in the availability of ever increasing amounts of information. Both the number of DB and their contents are growing fast. The total amount of information in the world is estimated to be doubling every 20 months, and much of this is being stored in computer DB. Within businesses what tends to happen is for management to not have access to this information in any user-friendly summary format. This means businesses have built up a reservoir of information but have restricted means for tapping that information for decision making purposes.
2. Data lies at the heart of every modern enterprise. The way it is used, how data is managed, and its quality and accuracy all impact on the success or failure of organisations in every industrial sector. Organisations need information quickly and accurately. But to access and verify records held within a vast DB used to be time consuming, complicated and expensive.

3. More and more people are required to make critical decisions because of increased competitive pressures but they need help to find the relevant data that could guide them in the decision-making process. This situation is termed the *"fact gap"* when many user/managers make decisions in a virtual vacuum using outdated information, borrowed perspectives or pure guesswork.
4. Data rich organisations which have large or varied data sources, often face problems of inconsistency and inaccuracy because they have historically suffered from an unmanageable array of data sources, collected by different people at different times from a variety of channels on a daily basis. Large, disparate DB mean that organisations frequently suffer from a poor standard of data quality and accuracy. Hence, individual records containing, for example, potentially valuable customer information are not harnessed for their true potential and organisations then miss crucial details through lack of knowledge.
5. SQL is the standard query language for accessing data held within a relational DB. With its powerful syntax, SQL represents a leap forward in DB access for all levels of management and computing professionals.
6. Many businesses, from small companies to major multi-nationals, have staff that would benefit from simple access to the organisation's data but are denied it by the complexities of query syntax, such as SQL, and the data structures involved. Training staff can be prohibitively expensive and conventional systems demand higher degrees of computer literacy than may be available. To solve this problem several intelligent tools have been created [1,2].

The central question to be addressed by this paper is how to improve access to DB for users. Such users may not understand DB, may not know exactly what is in the DB and how data is stored there. Crucially, it follows that they do not possess the means for extracting data. Hereafter, let us use the general term "**A**pplication **D**omain" (**AD**) to refer to the joined tables of DB and corresponding knowledge about DB contents and metadata (where metadata is a DB value's field' description and tables connectivity). Such knowledge will be stored in a **K**nowledge **B**ase (**KB**). Within this, there are (at least) the following two important issues to discuss: (1) natural User-AD interface, and (2) natural user's enquiry to SQL-query conversion. We feel that many of the concepts we have developed over the years revolve around the problem of representing complex database schemas using simple natural language terminology. This can be of great benefit to any type of data access tool, or to any data access situation.

Natural User-AD Interface

The main requirement for IDAS is - to handle non-standard or poorly formed (but, nevertheless, meaningful) user's enquiries. Let us distinguish four different types of **N**atural **U**sers' **E**nquiries (**NUE**): (1) **N**atural **L**anguage **E**nquiry (**NLE**); (2) NLE **T**emplate (**NLET**); (3) **E**nquiry **D**escriptors (**ED**), and (4) **I**mmediate **E**nquiry (**IE**). Such enquiries permit users to communicate with a DB in a natural way rather than through the medium of formal query languages. Obviously issues in these four NUE are related, and the knowledge needed to deal with them may be distributed throughout a NL **I**nterface (**NLI**) system. We want to underline here that the selection of NUE type is not just a user decision because some DB may not be appropriate targets for NLI. It is important to have a clear understanding of these problems so that the NLI can mediate between the user view, as represented by the NLE, and the underlying database structure. Let us consider these four types of NLE.

Natural Language Enquiry provides end users with the ability to retrieve data from a DB by asking questions using plain English. But there are several problems of using NLE:

- The end users are generally unable to describe completely and unambiguously what it is they are looking for at the start of a search. They need to refine their enquiry by giving feedback on the results of initial search e.g. *"I'm looking for a **nice** city in France for holiday"* (where *Nice* is a city in France but also an adjective in English). Parsing of such simple NLE is quite complicated and requires powerful KB from IDAS [4]. Except lingual ambiguity a lot of problem cause ***"DB field's values"*** ambiguity. For example, in NLE *"I'm looking for the address of an insurance company in Bolton"* the word *Bolton* is a value of the field *City*, part of the value in the field *Company* (e.g. *"Bolton Insurance Company"*), as well as being part of an address (e.g. *"Bolton Road"*);
- Very often a user's NLE cannot be interpreted because the concepts involved are outside of the AD. Therefore IDAS should have an ability to decide whether the NLE is meaningful or not. In the result of analysis of no meaningful NLE, IDAS should describe to the user what is wrong with the NLE and how the enquiry

might be rephrased to get the desired information. Such an approach, however, requires a very complicated KB in order to establish a meaningful communication with the user during the dialogue. Moreover, clarifying dialogue for the user creates a bad impression about IDAS because the user wants to be understood by IDAS immediately, without any additional effort on their part.

- It is simply impossible to require the users to know the exact values in DB (e.g. name of constituency in ***Election*** AD) in order to ask correctly what is a very simple question: "*Who won the election in Suffolk Central & Ipswich North in 2001?"*. For example, if the user instead of using the symbol ***'&'*** instead types in ***"and"*** IDAS will not find the constituency in DB. But IDAS is an intelligent system and in the result of NLE analysis IDAS understand that user possibly mentioned two different constituencies *Suffolk Central* and *Ipswich North* but both of them incorrectly because there also exists *Suffolk Coastal*, *Suffolk South*, *Suffolk West* and *Ipswich* constituencies. Clarification dialog generated by IDAS irritates user:

  IDAS: *Do you mean Suffolk Coastal, Suffolk South, or Suffolk West constituency?*

  User: *No, I mean Suffolk Central.*

  IDAS: *Suffolk Central constituency does not exist but there is Suffolk Central & Ipswich North constituency.*

  User: *It's exactly what I meant.*

  IDAS: *Thank you.*

- Very often NLE is ungrammatical.
- Direct observation of user NLE shows that all users are lazy i.e. they want to achieve the desired result by using minimum effort. They do not want to type in the long NLE such as "Identify the parts supplied by each vendor and the cost and sales value of all these items at present on order". *This is natural behaviour of human being in accordance with* the ***principle of simplicity***, or ***Occam's razor principle*** *(Occam's (or Ockham's) razor is a principle attributed to the 14th century logician and Franciscan friar; William of Occam. Ockham was the village in the English county of Surrey where he was born)*. The principle states that "**Everything should be made as simple as possible, but not simpler"** (*The final word is of unknown origin, although it's often attributed to Einstein, himself a master of the quotable one liner*). *Finding a balance between simplicity and sophistication at the input side has been discussed in [5].*

  Thus, firstly, NLE does not necessarily mean the enquiry is in plain English, secondly, IDAS should provide different levels of simplicity for NLE. The first step in this direction is NLET.

Natural Language Enquiry Template combines a list of values to be selected when required and generalization of users' NLEs. Examples of some **F**requently **A**sked **Q**uestions (**FAQ**) in AD ***Election*** are shown below:

- *What was the result in <constituency>?*
- *How many votes did <party> win in <constituency>?*
- *Which party won the election in <constituency>?*
- *Who won an election in <constituency>?*

Initial set of FAQ has been created by export in AD ***Election*** but in the result of activities new NLE have been collected by IDAS, analysed, generalized and then added to FAQ.

When the user selects an appropriate NLET with some descriptor in angular brackets IDAS immediately displays the list of corresponding values. As soon as the user finds the demand value by simply starting to type it and press button <Enter> result will be displayed (see Figure 1).

At first glance, the NLET is an ideal way to communicate with AD but in reality there are some problems, which need to be solved to provide lightness of communication. To highlight such problems is enough to consider quite a simple NLET: "*Who won an election in <constituency>?".* Without knowing *"who is who"* and meaning of *"won election"* IDAS cannot answer this question. To explain it to IDAS the **P**roduction **R**ules (**PR**) need to be involved. Many researchers are investigating how to reduce the difficulty of moving a NLI from one AD to another. The problems in doing this include what information is needed and how the information needs to be represented. From our point of view, **P**reconditioned PR (PPR) is a quite powerful approach to solve this problem. The subset of PPR in format: **<Precondition>** $\mapsto$ **<Antecedent>** $\Rightarrow$ **<Consequent>** is shown below.

1. AD:Election ↦ who ⇒ candidate;
2. AD:Election ↦ [candidate]:<win⊕won> ⇒ [SQL]:<MAX(votes)>;
3. AD:Athletics ↦ [runner]:<win⊕won> ⇒ [SQL]:<MIN(time);
4. AD:Athletics ↦ [shooter]:<win⊕won> ⇒ [SQL]:<MAX(distance);
5. AD:Election & DB:MS Access ↦ votes ⇒ [Field]:<CANDIDATE.VOTE>;
6. AD:Election & DB:MS Access ↦ candidate ⇒ [Field]:<CANDIDATE.[CANDIDATE NAME]>;
7. AD:Election & DB:Oracle ↦ [party]:<win⊕won> ⇒ [SQL]:<MAX(SUM(votes))>;
8. AD:Election & DB:MS Access ↦ [party]:<win⊕won> ⇒ [SQL]:<TOP1, SUM(votes), SUM(votes) DESC>,

where ⊕ **-** denotes "*exclusive OR*"**. Precondition** consist of **class$_1$:value$_1$ {& class$_i$:value$_i$}**. **Antecedent** might be represented by: (i) **single word** (e.g. *who, won, August*, *seven*, *etc.*), (ii) **sequence of words** (e.g. *as soon as*, *create KB*, *How are you doing*, etc.), or (iii) **pair - [context]:<value>**. Context allows one to avoid word ambiguity and thereby distinguish difference between "*Candidate won an election"* and "*Party won an election"*. Presentation of **Consequent** is similar to Antecedent structure except (iii). For Consequent pair represents **[descriptor]:<value>**.

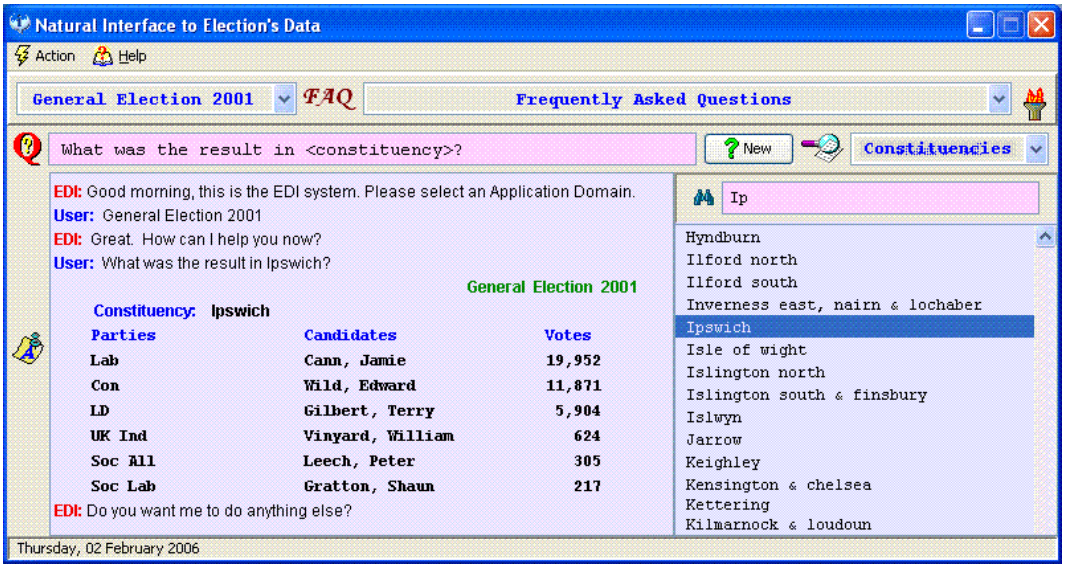


Figure 1. Natural Language Enquiry Template

For AD *Election* subset (1, 2, 5..8) of PPR is used. PPR 3 and 4 in fact show another meaning of the same word *won* but for a different AD. The last two PPR show the simplest way to cover the difference in SQL for different DB. Result of parsing considered NLET using selected PPR is shown on Figure 2.

Thus, NLET allows the user to "be lazy" but requires great effort to create the proper set of PPR as part of KB to describe better the more meaningful words. But using NLE and NLET we cannot say that all meaningful words have been described even for quite restricted AD. As a result some users will be disappointed by the IDAS reply. ED is a step in the direction towards simplifying KB and increasing the reliability of IDAS.

Before moving to ED it would be sensible once more to address some NLE and NLET problems. The cognitive process of understanding is itself not understood. First, we must ask: "*What it means to understand a* NLE?". The usual answer to that question is to model its meaning. But this answer just generates another question: "*What does meaning means?".* The meaning of a NLE depends not only on the things it describes, explicitly and implicitly, but also on both aspects of its causality: "*What caused it to be said*" and "*What result is intended by saying it*". In other words, the meaning of a NLE depends not only on the sentence itself, but also on **Who** is asking the question and **How** the question is phrased.

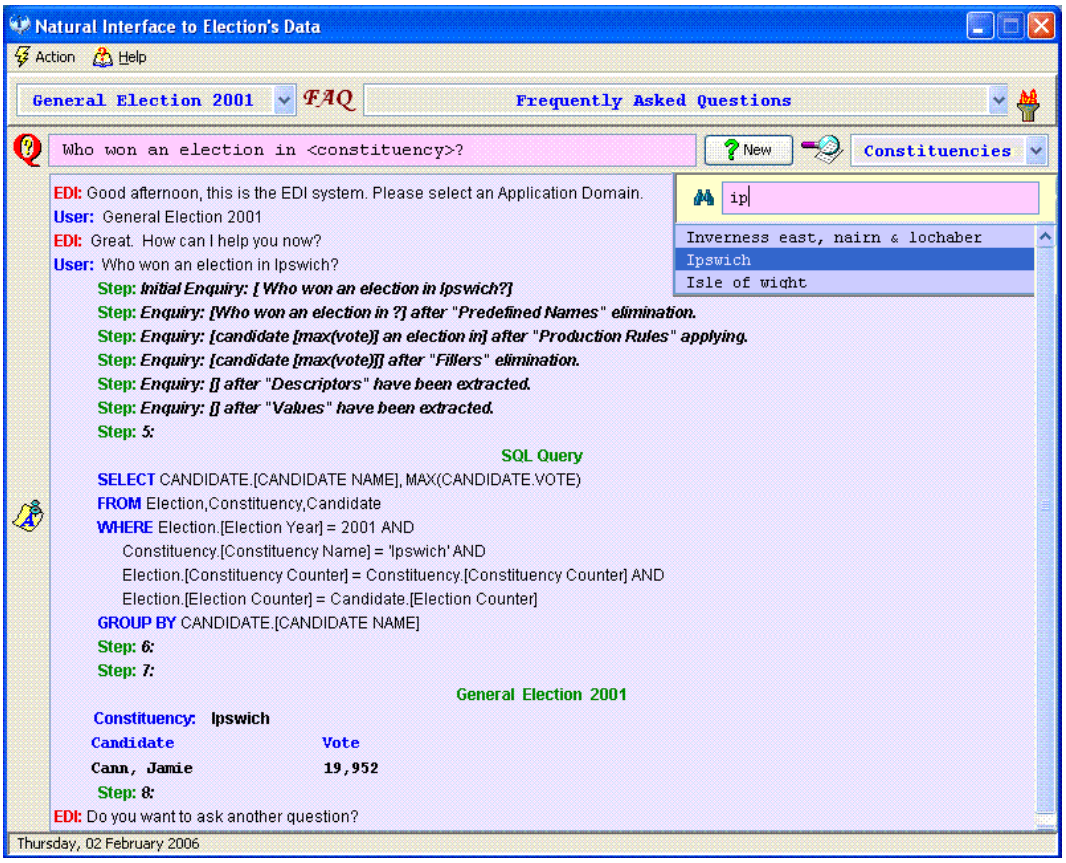


Figure 2. Natural Language Enquiry Parsing

From the linguistics point of view the process of understanding is possible under the following, as a minimum, three conditions [6]:

- IDAS must comprehend and understand separate words but lexical ambiguity sometimes makes such understanding difficult. A classic example of lexical ambiguity is the sentence: *"Time flies like an arrow"*. Each of the first three words could be the main verb of the sentence, and *"time"* could be a noun or an adjective, *"flies"* could be a noun, and *"like"* could be a preposition. Thus, the sentence could have various interpretations other than the accepted proverbial one. It could, for example, be interpreted as a command to an experimenter to perform temporal measurements on flies in the same way they are done on arrows. Or it could be a declaration that a certain species of fly has affection for a certain arrow.
- IDAS must understand the structure of the whole sentence but sometimes that is not a simple matter. If we have an ambiguous phrase such as: *"John saw the woman in the park with a telescope"*, then we usually understand one meaning and ignore the alternative interpretations.
- An empirical study revealed that only **0.53%** of possible sentences considered being grammatical are actually produced [7, p.823]. Note that the capacity to cope with ungrammatical NLE is one of the important requirements of NLE processing.

For artificial system like IDAS the power of natural language to describe the same events in different ways is a great problem. For example, the primitive event: *"Delete a cursor from the screen"* might be described as: *"eliminate a cursor", get rid of a cursor", "remove a cursor from the screen", erase a cursor", "makes a cursor hidden", "set the cursor size to 0", "take away a cursor from the screen"*, etc. Therefore the ED might release IDAS from such problems.

Enquiry Descriptors is especially useful when AD is not simple (e.g. AD ***Mobile Messages*** on Figure 3). And another important point of using ED is that modern technology has completely changed the way that people use the telephone to exchange dialogue with information held on computers. Well developed *"written speech analysis"* does not work with *"verbal speech"* [3]. For example, the first step of Speech Recogniser to parse NLE

"I'm looking for address of insurance company in Bolton" will be filler deleting i.e. *"I'm looking for"*. Finally, initial NLE will be represented as a set of descriptors, which represent the NL description of meaningful fields of AD.

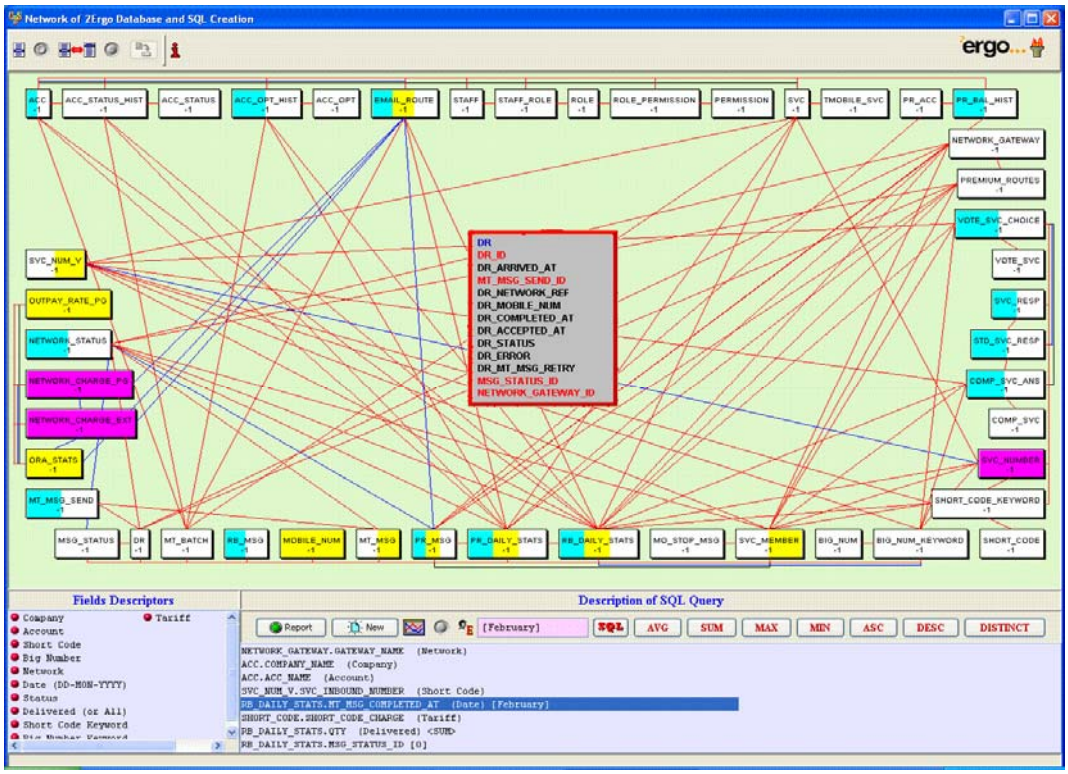


Figure 3. AD "*Mobile Messages"* and example of Enquiry Descriptors

The definition of "meaningful fields" depends on AD objectives. For the considered AD ***Mobile Messages*** is a list of descriptors: {*company, account, network,* etc.}. Between descriptors and meaningful fields exist one-to-one attitude. The procedure for creating ED is very simple (see Figure 3):

- Select desirable descriptors. In the result of selection the corresponding <Table>.<Field> (Descriptor) will be displayed;
- Select field, value for which needs to be assigned, enter value in square brackets and press <Enter>. For descriptor *Date* value [*February*] was defined;
- If some mathematical function need to be involved press corresponding button. To summarize all delivered messages button <***SUM***> has been clicked for selected descriptor *Delivered*;
- Click button SQL to convert ED to SQL-query.

If objectives of using AD changes then set of descriptors need to be extended, which requires effort of KB administrator. But this is the simplest way of extracting data from AD – using **IE**.

Immediate Enquiry is useful for users who are familiar with AD structure and know the meaning of tables and their fields. To create IE, firstly, select table, secondly, select desirable field (see Figure 3). Pair <Table>.<Field> will be displayed. Now user can add a descriptor and do the same procedure as for ED.

Natural User's Enquiry to SQL Query Conversion

The steps of NLE to SQL query are well defined [1]: **(NLE ∨ NLET) → ED → IE → SQL-query.** The final step is quiet complicated because the necessity to access data from many different tables within an AD and join those tables together in a report needs to be implemented. This is extremely important because non-technical users do not know how to join tables to get a more comprehensive view of their data. Quite often a very simple question in English can turn into a very complicated SQL-query e.g. conversion of NLE *"Display all messages amount for all networks in the last month"* gives SQL-query shown on Figure 4.

```
SELECT    NETWORK_GATEWAY.GATEWAY_NAME "Network",SUM(CASE WHEN RB_DAILY_STATS.MSG_STATUS_ID = 0 THEN RB_DAILY_STATS.QTY ELSE 0 END) "Delivered",
          SUM(CASE WHEN RB_DAILY_STATS.MSG_STATUS_ID = 4 THEN RB_DAILY_STATS.QTY ELSE 0 END) "Pending",
          SUM(CASE WHEN RB_DAILY_STATS.MSG_STATUS_ID = 1 THEN RB_DAILY_STATS.QTY ELSE 0 END) "Expired",
          SUM(CASE WHEN RB_DAILY_STATS.MSG_STATUS_ID = 2 THEN RB_DAILY_STATS.QTY ELSE 0 END) "Rejected",
          SUM(CASE WHEN RB_DAILY_STATS.MSG_STATUS_ID = 3 THEN RB_DAILY_STATS.QTY ELSE 0 END) "Undeliverable",SUM(RB_DAILY_STATS.QTY) "Total"
FROM      NETWORK_GATEWAY,RB_DAILY_STATS
WHERE     TRUNC(RB_DAILY_STATS.MT_MSG_COMPLETED_AT) BETWEEN '01-JAN-2006' AND '31-JAN-2006' AND NETWORK_GATEWAY.NETWORK_GATEWAY_ID = RB_DAILY_STATS.NETWORK_GATEWAY_ID
GROUP BY NETWORK_GATEWAY.GATEWAY_NAME
```

Figure 4. Result of NLE to SQL-query conversion

Even the simplest ED like "W*hite thick sliced bread*" cannot be directly converted to SQL-query because AD's data might contain any combination of wrong and correct words and, therefore, four PPR ("*white* $\Rightarrow$ *wht* ", "*thick* $\Rightarrow$ *thk* ", "*sliced* $\Rightarrow$ *slcd* $\oplus$ *sld*", and "*bread* $\Rightarrow$ *brd*") is required [3]. Theoretically, for the considered example there are 16 possible combinations of data, namely**:** (1) "*White thick sliced bread*", (2) "*White thick sliced brd*", … , (16) "wht thk (slcd $\oplus$ sld) brd". Result of such conversion is shown in Figure 5.

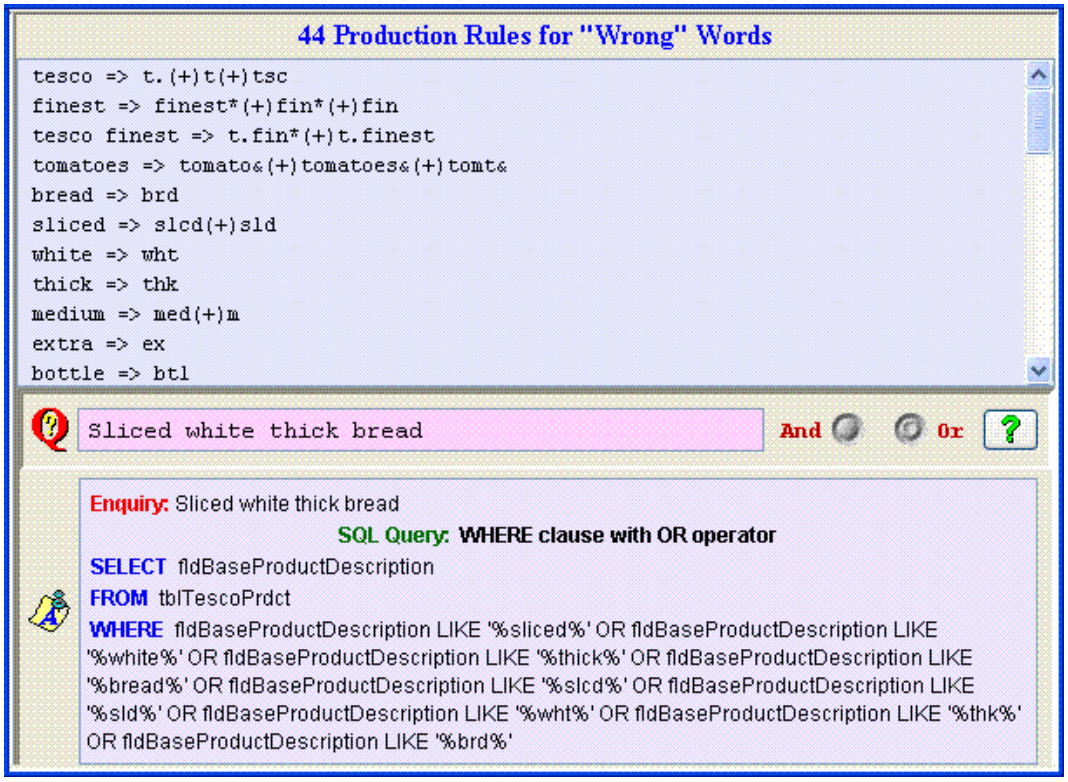


Figure 5. ED to SQL-query conversion using PPR

The idea of joining tables in SQL is that individual rows in one table are attached to some corresponding rows in another table. The criteria for joining rows are decided by the highly skill SQL user. IDAS provides automatic Tables Coupling (TC). The main problem of TC is to select the right tables link from a huge number of possible links. Result of conversion ED from Figure 3 to SQL-query is shown as Figure 6. It is easy to see on TC decision tree shown the amount of possible TC Solutions (TCS). It is important to underline that output produced by SQL-query with different TCS might be different. In such situations a critical question arises: *"What is a criteria of selection of theTCS, which provides the right output?"*. IDAS activities are based on the hypothesis that ***"the right output might be produced by SQL-query with the best TCS"***, where the definition of *the best TCS* is obvious. *Let us call TCS the best if for each pair of tables the shortest link was used*. The given definition follows *the principle of simplicity* described earlier. Red lines on Figure 6 indicate TCS. TC decision tree was created using the *breadth-first method*. *Unipath heuristics rule* has been involved for selection of the best TCS. Two different type of fields are used as *foreign keys* to provide TCS:

- *Primary keys* e.g. ACC.ACC_ID = RB_DAILY_STATS.ACC_ID i.e. primary key ACC_ID from table ACC had been placed into table RB_DAILY_STATS as a foreign key.

- *Value fields*. Sometimes for different reasons, the DB has data redundancy i.e. in different tables there are fields with the same data (data duplication). The names of such fields are not necessarily the same. In that case at the stage of KB creation such field names should be described as synonyms. In the considered example, fields SVC_INBOUND_NUMBER from table SVC_NUM_V and SHORT_CODE from table SHORT_CODE are synonyms. Figure 6 has a double red line which shows the links between them.

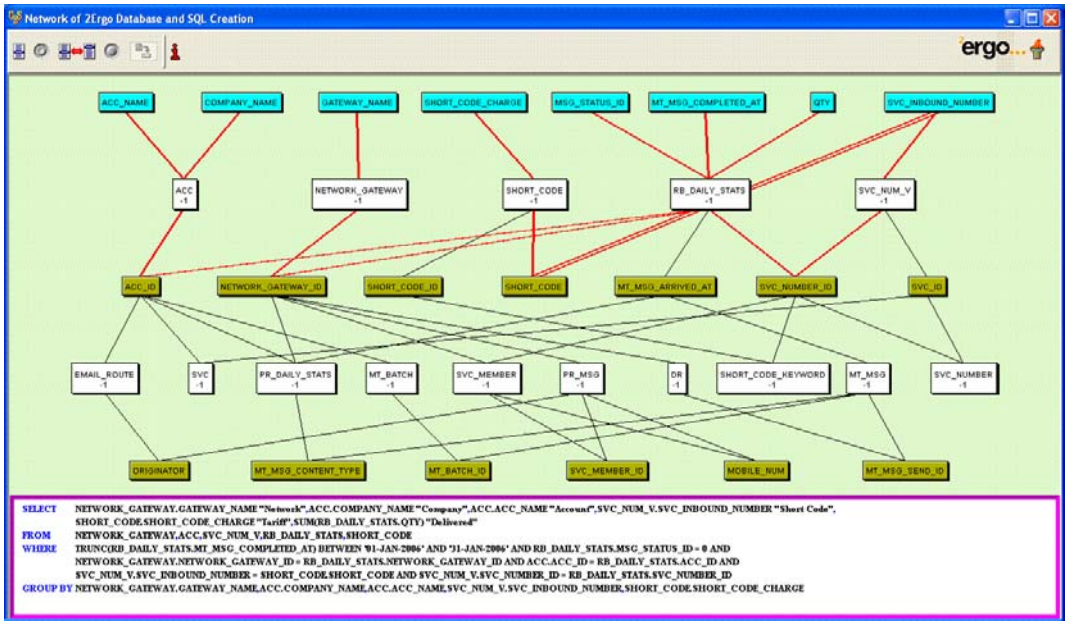


Figure 6. TC Decision Tree and TC Solution

Conclusion

IDAS effectively allows us to place information directly into the hands of business users - eliminating the need for technical support specialists continually to address *ad hoc* requests from end users. To do it properly all four types of enquiries should be provided. IDAS shields the user from the complexity of the underlying technology and itself acts as an intelligent user assistant.

Bibliography

[1] V.A.Lovitskii and K.Wittamore, "DANIL: Databases Access using a Natural Interface Language"*, Proc. of the International Joint Conference on Knowledge-Dialogue-Solution: KDS-97*, Yalta (Ukraine), 282-288, 1997.

[2] G.Coles, T.Coles, V.A.Lovitskii, "Natural Interface Language"*, Proc. of the VIII-th International Conference on Knowledge-Dialogue-Solution: KDS-99*, Kacivelli (Ukraine), 104 -109, 1999.

[3] D.Burns, R.Fallon, P.Lewis, V.Lovitskii, S.Owen, "Verbal Dialogue versus Written Dialogue*, *Proc. of the XI-th International Joint Conference on Knowledge-Dialogue-Solution: KDS-2005*, Varna (Bulgaria), 336-244, 2005.

[4] T.Coles, V.A.Lovitskii, "Text Searching and Mining", J. of *Artificial Intelligence,* National Academy of Sciences of Ukraine, *Vol. 3,* 488-496, 2000.

[5] L.Huang, T.Ulrich, M.Hemmje, E.Neuhold, "Adaptively Constructing the Query Interface for Meta Search Engines", Proc. of the *Intelligent User Interface Conf.,* 2001.

[6] Harris R.J. Monaco G.E., "Psychology of Pragmatic Implication: Information Processing between the Lines", Journal *"Exp. Psychol. General"*, 107, 1978, pp.1-22.

[7] Kitano H., "Challenges of Massive Parallelism", *Proc. Of the 13th International Joint Conference on Artificial Intelligence (IJCAI-93)*, Vol. 1, 1993, pp.813-834.

Authors' Information

Guy Francis *– 2 Ergo Ltd, St. Mary's Chambers, Haslingden Road, Rawtenstall, Lancashire, BB4 6QX, UK, e-mail: guy.francis@2ergo.com*

Mark Liashman *– 2 Ergo Ltd, St. Mary's Chambers, Haslingden Road, Rawtenstall, Lancashire, BB4 6QX, UK, e-mail: mark.liashman@2ergo.com*

Vladimir Lovitskii *– 2 Ergo Ltd, St. Mary's Chambers, Haslingden Road, Rawtenstall, Lancashire, BB4 6QX, UK, e-mail: vladimir@2ergo.com*

Michael Thrasher *– University of Plymouth, Plymouth, Devon, PL4 6DX, UK e-mail: mthrasher@plymouth.ac.uk*

David Traynor *– 2 Ergo Ltd, St. Mary's Chambers, Haslingden Road, Rawtenstall, Lancashire, BB4 6QX, UK, e-mail: david.traynor@2ergo.com*

DISTINCTIVE FEATURES OF MOBILE MESSAGES PROCESSING

Ken Braithwaite, Mark Lishman, Vladimir Lovitskii, David Traynor

Abstract: *World's mobile market pushes past 2 billion lines in 2005. Success in these competitive markets requires operational excellence with product and service innovation to improve the mobile performance. Mobile users very often prefer to send a mobile instant message or text messages rather than talking on a mobile. Well developed "written speech analysis" does not work not only with "verbal speech" but also with "mobile text messages". The main purpose of our paper is, firstly, to highlight the problems of mobile text messages processing and, secondly, to show the possible ways of solving these problems.*

Keywords: *mobile text messages, text message analysis, natural language processing*

Introduction

1. The reasons why is very difficult to use the classical linguistic approach for verbal speech analysis have been considered in [1]. In this paper the problems of Mobile Short Message (MSM) analysis will be discussed. MSM represents plain text message of 160 characters or less and provided by mobile SMS (short message service). The year 2005 saw an explosion in the volume of MSM being sent to mobile phones. Mobile's users choose to send MSM rather than talking on a mobile call because [2]:
   - ♦ They don't have time to chat phone (74%).
   - ♦ To not disturb other patrons on public transportation or at a sporting event or restaurant (53%).
   - ♦ To get work done and send quick notes when on the road travelling for business (32%).
   - ♦ Less disturbing than phone calls (72.5%).
   - ♦ One can reach the other party around the clock (30.4%).

   However, mobile operators need to understand that subscribers give greater priority to the convenience of using the service over the technology and capabilities it offers. Therefore, more effort must be placed on creating user-friendly client interfaces that integrate effectively with the handset features.
2. A wide variety of information services can be provided by SMS, including weather reports, traffic information, inventory management, itinerary confirmation, sales order processing, asset tracking, automatic vehicle location, entertainment information (e.g., cinema, theatre, concerts), financial information (e.g., stock quotes, exchange rates, banking, brokerage services), and directory assistance. SMS can support both *push* (i.e. mobile-terminated (MT)) SM and pull (i.e. mobile-originated (MO)) SM to allow not only delivery under specific conditions but also delivery on demand, as a response to a request.

3. The important distinctive feature of MSM is that the majority of them are bilingual (i.e., using both English words and mobile slang from Tegic's T9 dictionary [3]).
4. We will consider MSM in indissoluble link with Inbound Number (INo) represented by a short code (it is typically a 5 digit number which is accessible by subscribers of any mobile operator) or long code (a usual mobile number– works across all operators).
5. Information services as described above are provided by "Content Providers" who must rent an INo. This can be dedicated to provide a single service or shared to provide multiple services. In they case of multiple services, they are distinguished by the use of a key word that user must provide as the first word of the MSM.
6. The standard 12-key keypad found on many mobile phones today (see Figure 1). On this Figure "Imitator of Mobile" is represented. Alphabetic letters are mapped to keys '2' through '9'. However, this arrangement poses problems for text entry. As three or four letters share the same key, some form of disambiguation is required to determine which letter is intended by the user. There are currently two main methods that are usually used on mobile phones for text entry. They are the multi-tap method and the predictive text entry method. In the multi-tap method, a user taps the key that contains the letter repeatedly until the desired letter appears. The number of taps required depends on the position of the letter on the key. In predictive text input method (e.g., Tegic's T9 [3]), the user presses the key that corresponds to each letter of a word once. The system uses a dictionary of words to determine which of the possible words the key sequence matches. When MSM is received on a particular INo, then for a dedicated INo the MSM is forwarded to the client renting it. If the INo is shared, the MSM needs to be examined to identify the client and the individual service.
7. First we will describe the types of MSM and the problems encountered examining the MSM. The MSM might be represented by:

- **Letter** or **digit**. For example, a number of promotions are quizzes/competitions and sometimes are also interactive, i.e., multiple messages/responses. If the original message to the customer is a question, such as *"How many legs has my dog got?"* then the customer could reply *1*, *2*, *3*, or *4*. Some promotions are multi-choice answers e.g., *'a'*, *'b'*, or *'c'*.
- **Single word** or **number** (e.g. credit card number).
- **Sequence of words** or **numbers.**
- **Combination of words** and **numbers** in MSM.

The main purpose of this paper is to investigate the **bad pairs INo ↔ MSM** and find ways to restore them.

Figure 1. Standard 12-keys keypad

Let's call pair INo ↔ MSM ***bad*** if:

- INo does not exist;
- Type of MSM was not recognised or keyword of MSM was not recognised. Very often the first whitespace-delimited word represents keyword (KW) and allows the identification of the client;
- The pair INo ↔ MSM does not exist because (¬INo & MSM) ∨ (INo & ¬MSM),

where ¬INo and ¬MSM stand for *wrong* INo and *wrong* MSM respectively. Let's call INo and MSM *wrong* if they separately exist but link between INo and KW of MSM does not. The reason of wrong MSM is understandable. For example, a user can tap the 2-key once to get 'a', twice to get 'b' and thrice to get 'c'. If he taped wrongly then instead of desired word *bell* he typed *cell*, or using 6-key instead of *come* was *cone*.

- A special type of MSM (so called ***stop MSM)*** requires synonyms for recognition e.g., *cancel*, *remove*, etc.
- Finally, we would like to underline the most difficult and dangerous problem when INo ↔ MSM exists but

$$((INo^T \neq INo^D) \,\&\, (KW^T = KW^D)) \vee ((INo^T = INo^D) \,\&\, (KW^T \neq KW^D)) \vee ((INo^T \neq INo^D) \,\&\, (KW^T \neq KW^D)),$$

where letters ***D*** and ***T*** mean what user ***d****esired* to type and what was actually ***t****yped*.

This problem takes place because of ambiguity of both INo and KW i.e., one INo might link to several KW and many different INo might use the same KW, and vice versa.

Let's investigate these problems and discuss the results of KW, INo and bad MSM analysis. Our investigation was grounded in real data analysis. As a result of this discussion an algorithm to deduce the correct KW from a bad MSM will be described. Also, the result of using of this algorithm will be shown.

Keywords Analysis

The result of KW analysis and KW ambiguity is shown on Figure 2, namely:

- Total (valid + invalid) KW distribution among letters and mobile's keys (2-9). A KW is *invalid* if it currently is not used on the INo but at the same time the same KW might be valid for another INo. For example, KW *red* is valid for INo 81025 and 80039, and invalid for 89095;
- Displaying the list of KW for selected letter or Inbound No by clicking the corresponding letter or digit;
- For any KW (by clicking when the list of KW is displayed, or just simply typing in KW) the corresponding list of INo is displayed.;
- List of the next (= expected) symbols is displayed for the entered symbol (letter or digit);
- List of ambiguity for both valid and invalid KW is displayed.

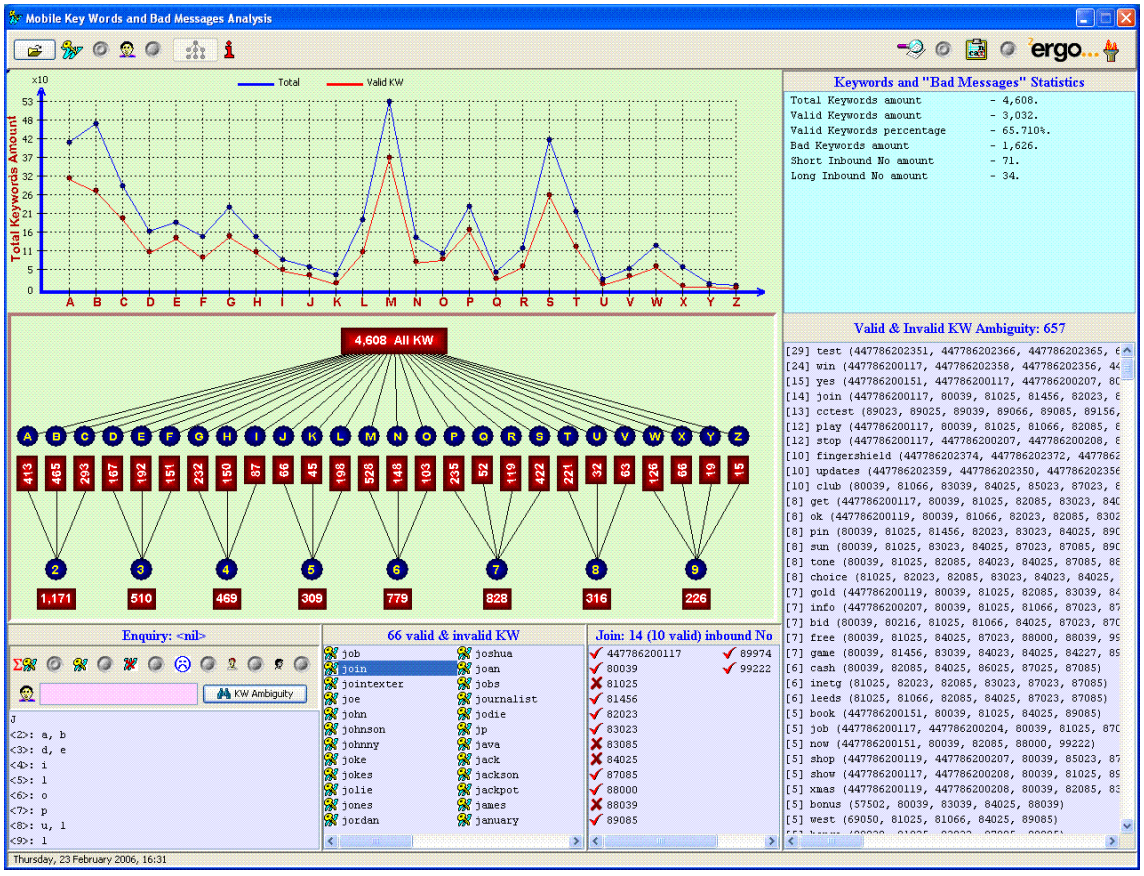


Figure 2. Keywords analysis and KW ambiguity

INo ambiguity is shown on Figure 3.

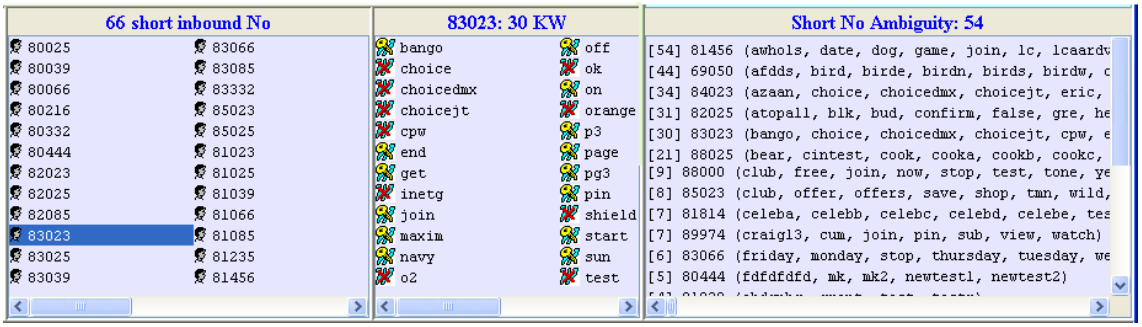


Figure 3. INo ambiguity

To provide such analysis the Knowledge Base (KB) has been created and used for KW, INo and bad MSM analysis, and KW and INo restoration. The main features of KB have been discussed in details in [4]. Here we would like to notice that in our case, under the **KB organisation** we would understand the regularity of data (INo and KW) distribution in memory assuring the storage of various links between them. At any time KB deals only with relatively *small fragments* of the external world. So, the corresponding structures are needed to integrate these fragments separated in time into the integral picture. The structures obtained as a result of integration should contain more information than it had been used for its creation. The organisation of KB should make allowance for such features as:

- associability;
- ability to reflect similar features for different objects and different features for similar objects (where objects are represented by KW and INo);
- heterarchical organisation of information [5]. The idea of heterarchical approach means that a full association of INo and KW represent very complicated net of nodes and unidirectional links between them. The predetermined hierarchy of "super-" and "subclasses" is absent; every node (INo or KW) is a "patriarch" in its own hierarchy if some process of search initiates with it.

Bad Messages Analysis

The main purpose of **Bad M**essages (BdM) is to classify BdM and allocate types of BdMs which might be restored. Several hundred thousand BdMs have been detected and result of this is as follows:

- Wrong KW among valid and invalid KW - 42.12%;
- Wrong KW among valid KW - 20.11%;
- Wrong KW among invalid KW - 22.01%;
- Wrong INo - 39.53%;
- “Stop” MSM - 8.78%;
- Empty MSM - 6.47%;
- Wrong alphabet (e.g. Russian) - 2.65%;
- Mobile slang (from T9 dictionary) - 0.37%;
- Rude MSM - 0.08%.

Remark: *Wrong INo* means literally **wrong** INo, e.g. 22120000, or **unknown** INo. So despite that 39.53% of *wrong INo* it would not be effective to spend more effort in trying to decrease this percentage. In the next session of paper some ideas of KW and right INo restoration will be discussed.

Algorithm of KW and/or INo Restoration

1 **INo recognition**. There are four possible type of INo: (i) ***valid***; (ii***) invalid***, (iii) ***unknown*** when either length of INo is different from short or long INo, or INo does not exist in KB. Remark: Checking existing INo in KB would be sufficient to find out if the INo is known or not. But this operation requires more time than simply checking the length of the INo, and (iv) ***wrong*** INo. Initial analysis of INo does not allow the identification of this type of INo. It would only be possible to do this when KW of the MSM is recognised.

2 **Initial MSM validation**. MSM will be classified as valid if only contains symbols from the Latin alphabet and/or digits are used. Hereafter, only valid MSM will be considered.

3 **Separators elimination** from MSM.

4 **Fillers elimination** from MSM. For example, in MSM: *"I'd like to stop sending messages"* ***I'd like to*** is a filler and will be deleted.

5 **Slang elimination** from MSM using T9 dictionary.

6 S**top** MSM recognition. Remark: In the current version of algorithm MSM *"s␣t␣o␣p"* will not be recognised as a *stop* MSM.

7 **Extracting set of KW** from KB related to **INo**, i.e. $\{KW_{INo}\}$, where $\{KW_{INo}\} \subset \{KW_{KB}\}$. $\{KW_{KB}\}$ represents all existing **KW** in **KB.**

8 **Extracting KW** from MSM, i.e. KW_M. Remark: In the current version of the algorithm only the **first** word of MSM is considered as a KW_M.

9 **Extracting set of INo** from KB related to KW_M, i.e. $\{INo_{KWm}\}$**, where** $\{INo_{KWm}\} \subset \{INo_{KB}\}$.

10 Pair INo $\leftrightarrow$ MSM is accepted if $((INo \in \{INo_{KWm}\} \wedge KW_M \in \{KW_{INo}\}) \Rightarrow$ **IS-Correct(MSM)**$) \mapsto$ **return**(KW_M), where predicate *IS-Correct(MSM)* is **true** when *"MSM is correct"* and **false** (i.e. $\neg$IS-Correct(MSM)) - otherwise. Symbol $\Rightarrow$ stands for word ***then*** and symbol $\mapsto$ means ***lead to***. Returned KW_M is used for further analysis.

11 Pair INo $\leftrightarrow$ MSM represents BdM, if
$(INo \in \{INo_{KWm}\} \wedge KW_M \notin \{KW_{INo}\}) \oplus (INo \notin \{INo_{KWm}\} \wedge KW_M \in \{KW_{INo}\}) \oplus (INo \notin \{INo_{KWm}\} \wedge KW_M \notin \{KW_{INo}\})$, where symbol $\oplus$ means **exclusive or.**

12 After recognition of BdM reason, the attempt to restore BdM is undertaken. To explain this step let us assume that the reason of BdM is:
$INo \in \{INo_{KWm}\} \wedge KW_M \notin \{KW_{INo}\}$.
From this it follows that:
$INo \in \{INo_{KWm}\} \wedge KW_M \notin \{KW_{INo}\} \wedge (KW_M \in \{KW_{KB}\} \oplus KW_M \notin \{KW_{KB}\})$**.**
If $KW_M \in \{KW_{KB}\}$ then attempts to correct INo should be undertaken. The next step will describe the more complicated case of KW_M correction when $KW_M \notin \{KW_{KB}\}$.

13 KW_M **correction.** There are two different approaches to restore KW_M:

(1) The first approach provides searching $KW_i \in \{KW_{KB}\}$ under several conditions:

- the difference in length of words KW_i and KW_M must be less or equal **1;**
- just two different symbols might be in KW_i and KW_M. This rule covers four possible types of misspelling (the word ***attempt*** is used to demonstrate the first three types): (i) ***attepmt***; (ii) ***atempt***; (iii) ***attembt***, and (iv) ***ozlo***. The last type should be considered more attentively. There are two different reasons for this type of misspelling:

  I. Problem of **symbol recognition**. Very often it is simply impossible for the user to distinguish the letter ***'l'*** from the digit ***'1'***, especially when, for example, the previous symbols are letters but for correct KW digit ***'1'*** need to be typed in, e.g. ***oz10***.

II. **Easier typing**. For the user it is easier to press the button **0** once than to press the button **6** three times to enter the letter ***'o'*** in word ***bonus***, because for any reader it is still easy to understand word the ***b0nus***. Another example, when instead of the letter ***'l'*** (pressing the button **5** three times), or ***'i'*** (pressing the button **4** three times) entered digit ***1*** e.g. ***tab1e***.

- **Similarity of words** $\mathbf{KW_i}$ and $\mathbf{KW_M}$ must be more or equal to some **Threshold of Similarity (TofS), i.e.** $\mathbf{Smlrt(KW_i,KW_M) \geq TofS}$. The calculation of $\mathbf{Smlrt(KW_i,KW_M)}$ as a percentage is quite simple**:**

$$\mathbf{Smlrt(KW_i,KW_M) = (ACS_{LR}(KW_i,KW_M) + ACS_{RL}(KW_i,KW_M)) * 2 / (Length(KW_i) + Length(KW_M))*100,}$$

where $ACS_{LR}(KW_i,KW_M)$ and ACS_{RL} stand for **A**mount of **C**ompared **S**ymbols from **L**eft to **R**ight and **R**ight to **L**eft respectively. For example, for considered words: ***attepmt***; ***atempt***; and ***attemppt*** the values of $\mathbf{Smlrt(KW_i,KW_M)}$ are as follows**:**

Smlrt(*attempt*,*attepmt*) = (4+1)*2/14*100=71.43%,

Smlrt(*attempt*,*atepmt*)=(2+4)*2/13*100=92.31%, and

Smlrt(*attempt*,*attemppt*)= (6+1)*2/15*100=93.33%.

Remark: In the result of comparison of words ***attempt*** and ***attemppt*** from *left to right* two sequences remain to be compared from *right to left*: ***t*** and ***pt***. That is why $ACS_{RL}(KW_i,KW_M)$ = 1. The compact description of first approach to restore KW_M might be presented in the following manner:

$$\mathbf{\exists\, KW_i\, ((KW_i \in \{KW_{KB}\}) \wedge (Smlrt(KW_i,KW_M) \geq TofS)) \mapsto return(KW_i),}$$

where quantifier ∃ means *exist*.

To find out an appropriate value for **TofS** thousands of BdMs have been tested for three different values of **TofS – 50.0%, 75.0%,** and **100%**. The decreasing of restored KWs are:

6,370 → (-1,137) → 5,233 → (-709) → 4,524.

That is caused by **type 1** of misspelling (wrong sequence of two letters), because $\mathbf{Smlrt(KW_i,KW_M)}$ is very sensitive to a word's length, e.g. **Smlrt(*node*,*ndoe*)=50.0%, Smlrt(*table*,*tabel*)=60.0%,** and **Smlrt(*axmpridel*,*amxpridel*)=77.78%.** In the current version of the algorithm TofS **= 75.0%** because type 1 misspelling occurs very seldom in short words (i.e. with a length less than 6 characters).

(2) If the previous approach was not success then algorithm is trying to find such $\mathbf{KW_i \in \{KW_{KB}\}}$ that is

(i) an initial part of $\mathbf{KW_M}$, i.e. $\mathbf{KW_i \triangleright KW_M}$,

(ii) $\mathbf{\forall KW_i\, (KW_i \in \{KW_{KB}\} \wedge KW_i \triangleright KW_M)}$ **Select(max(Length(KW$_i$)),** where quantifier ∀ means *from all* and **Select(max(Length(KW$_i$))** stands for *"select KW$_i$ with maximum length", and*

(iii) $\mathbf{(Length(KW_M) - Length(KW_i)) \leq (Length(KW_M)/2)}$, e.g. ***airtext*** ▷ ***airtextww3514*.**

14 **INo correction.** Result of $\mathbf{KW_M}$ correction is shown on Figure 4. To describe the INo correction let us suppose that pair "*81025* ↔ *cash*" has been entered. This pair has been recognised as BdM because

$\mathbf{INo \notin \{INo_{KWm}\} \wedge KW_M \notin \{KW_{INo}\} \wedge KW_M \in \{KW_{KB}\} \wedge INo \in \{INo_{KB}\}.}$ $\mathbf{\{INo_{cash}\} = \{84025, 86025, 87025, 82085, 87085, 87023\}.}$ It would be not acceptable to advise the user: *"Please try to dial 84025, 86025, 87025, 82085, 87085, or 87023"*. Instead a heuristic approach is used and might be describe as follows:

- For each button define a set of ***"direct neighbour"*** buttons (**DrctN**) and a set of ***"diagonal neighbour"*** buttons (**DgnlN**). Given terms easy to explain by example: DrctN(5) = {2, 4, 6, 8} and DgnlN(5) = {1, 3, 7, 8}.
- Find out the wrongly pressed button. For the considered example, Smlrt(*81025*,*84025*)=80%. The same result that we have for INo *86025* and *87025*. Thus it is very likely that the wrongly pressed button was **1**.
- Now the right button should be selected. DrctN(1) = {2, 4} and DgnlN(1) = {5} associated with button 1. First of all the right button is searching among DrctN(1). It is easy to see that only button 4 could be the right button and that is why INo 84025 is displayed (see Figure 5).

Figure 4. KW_M correction

Figure 5. INo correction

The result of testing both KW and INo correction is represented on Figure 6.

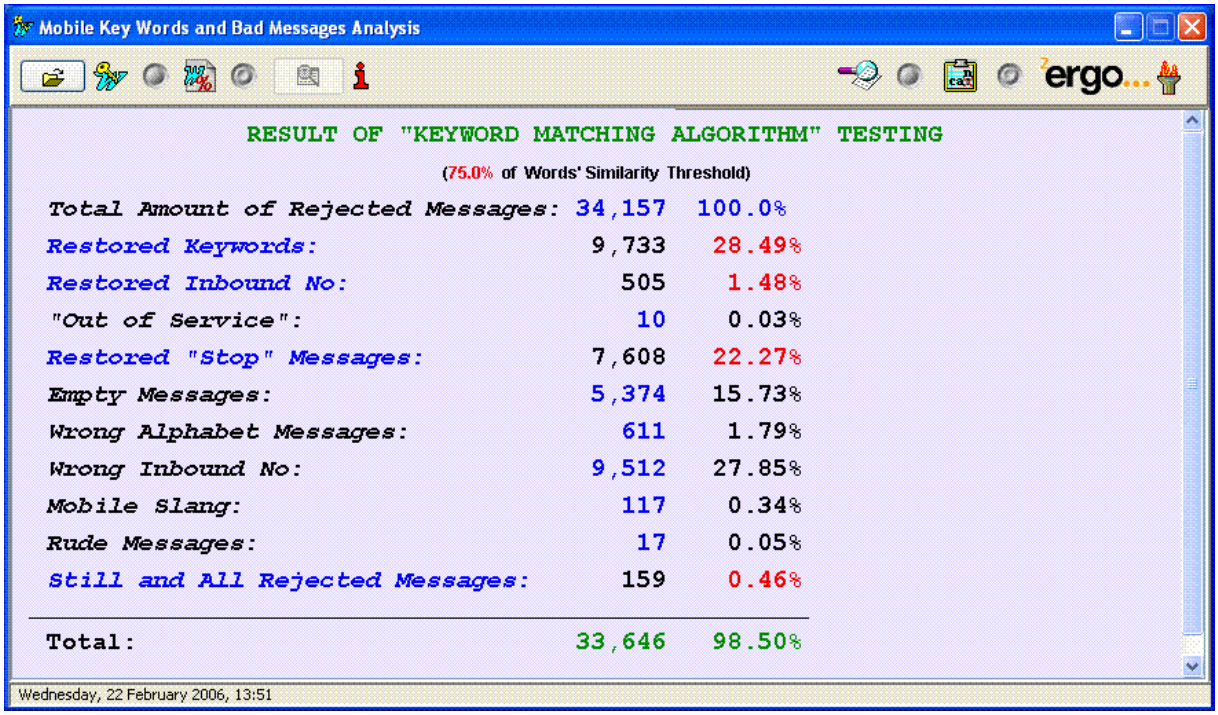


Figure 6. Result of Algorithm Testing

Remark: In Figure 6 the amount of **distinct** *"Still and All Rejected Messages"* is displayed and that is why the initial amount of BdM = 34,157 is more than the total amount of tested and corrected messages (33,646). The described algorithm improved BdM recognition by 52.25%.

Conclusion

The recent development in natural language processing has made it clear that formerly independent technologies can be harnessed together to an increasing degree in order to form sophisticated and powerful information delivery vehicles. Written speech, verbal speech and MSM analysis provide complementary functionalities, which can be combined to meet the modern technologies requirements.

Bibliography

[1] D.Burns, R.Fallon, P.Lewis, V.Lovitskii, S.Owen, "Verbal Dialogue Versus Written Dialogue*, *Proc. of the XI-th International Joint Conference on Knowledge-Dialogue-Solution: KDS-2005*, Varna (Bulgaria), 336-244, 2005.

[2] Opinion Research Corporation, www.orc.co.uk.

[3] Tegic Communication, www.tegic.com.

[4] V.A.Lovitskii and K.Wittamore, "DANIL: Databases Access using a Natural Interface Language", *Proc. of the International Joint Conference on Knowledge-Dialogue-Solution: KDS-97*, Yalta (Ukraine), 282-288, 1997

[5] M.R.Quillian, "Word concepts: A theory and simulation of some basic semantic capabilities", *C.I.P. working paper 79*, Cornegie Inst. of Technol., Pittsburgh, 1965.

Authors' Information

Ken Braithwaite *– e-mail: ken.braithwaite@2ergo.com*

Mark Lishman *– e-mail: mark.lishman@2ergo.com*

Vladimir Lovitskii *– e-mail: vladimir@2ergo.com*

David Traynor *– e-mail: david.traynor@2ergo.com*

2 Ergo Ltd, St. Mary's Chambers, Haslingden Road, Rawtenstall, Lancashire, BB4 6QX, UK

ИССЛЕДОВАНИЕ СТРУКТУРЫ И СВОЙСТВ ОБЪЕКТОВ И ЭЛЕМЕНТОВ СИНТЕЗА ДЛЯ ЗАДАЧИ ОЗВУЧИВАНИЯ ТЕКСТОВОЙ ИНФОРМАЦИИ

Юрий Г. Кривонос, Юрий В. Крак, Николай Н. Шатковский

Аннотация*: Проведен анализ задачи создания систем озвучивания текстовой информации. Описан метод конкатенативного TTS синтеза. Описаны особенности конкатенативных TTS систем. Предложен выбор объектов и элементов синтеза, представлены их структура и свойства. Рассмотрены некоторые фонетические, просодические и акустические характеристики естественной речи (для задачи озвучивания украинской речи).*

Ключевые слова: *озвучивание текстовой информации, синтез естественной речи, объекты и элементы синтеза, просодические характеристики речи.*

ACM Classification Keywords*: H.5.5 Sound and Music Computing: Signal analysis, synthesis, and processing.*

Введение

Проблема реализации речевого диалога человека и технических средств – актуальная задача современной кибернетики. Задача озвучивания текстовой информации и создания озвучивающих систем соприкасается с исследованиями в областях математического моделирования, цифровой обработки сигналов, фонетики, морфологии, словообразования и пр.

Достижение ощутимых результатов синтеза речи стало возможным лишь с возрастанием мощности вычислительной техники, а также с развитием математических методов и программных продуктов записи,

исследования и обработки цифровой звуковой информации. В основу достижений этих результатов положены исследования, проводимые учеными на протяжении 60–90-х годов XX столетия.

Начиная с 1999 года, для расширения доступа к Сети W3C работал над Моделью речевого интерфейса (Speech Interface Framework), которая позволит людям взаимодействовать, используя телефонную клавиатуру, устные команды, прослушивание предварительно записанной речи, синтезированную речь и музыку [W3C].

В середине 80-х была предложена концепция Text-to-Speech (TTS) синтеза. TTS синтез – это компьютерная система, которая любую полученную текстовую информацию, преобразовывает в эквивалентную звуковую речевую информацию, синтезируя новые слова, словосочетания, предложения [Dutoit, 1993].

Общая постановка задачи конкатенативного TTS синтеза речи

Технология TTS синтеза позволяет компьютерам преобразовывать произвольный текст в слышимую речь для доставки текстовой информации людям посредством голосовых сообщений. Ключевая цель TTS приложений в системах связи состоит из представления голосом текстовых сообщений. [Cox, 2000]

В последнее время увеличивается количество попыток ее решения, большинство достигнутых результатов связаны с концепцией конкатенативного Text-to-Speech синтеза.

Популярность данной концепции заключается в том, что в основе такого синтеза лежат естественные, произнесенные диктором, коренным носителем языка, элементы речи. Это приводит к достаточно высокому уровню естественности звучания синтезированной речи.

Сначала обрабатывается входящая текстовая информация (ТИ). Выделяются признаки и параметры конкатенативного синтеза, проводится анализ и сегментация текста на текстовые элементы синтеза (начиная аллофонами, заканчивая сложными сегментами синтеза – в зависимости от конкретного TTS метода) [Кривонос, 2005].

Операции, находящиеся в модуле цифровой обработки сигнала, являются компьютерным аналогом динамического контроля артикуляторних мышц и вибрирующей частоты голосовых связок таким образом, что выходящий сигнал подбирает входящие условия. Для того чтобы сделать это правильно, модуль обработки цифрового сигнала должен некоторым образом учитывать ограничения, поскольку для понимания фонетические переходы важнее чем постоянные состояния [Крак, 2005].

Согласно проведенному анализу необходимые звуковые элементы (эквивалентные текстовым) синтеза речи (заблаговременно, согласно определенной конкретной конкатенативной системы синтеза, записанные реальные речевые сигналы) поддаются обработке методами обработки звуковых элементов. Обработанные речевые звуковые сигналы поступают на блок озвучивания звуковых элементов, где и происходит генерация выходящего звукового сигнала [Кривонос, 2005].

Схематически работа конкатенативных систем TTS синтеза представлена на рисунке.

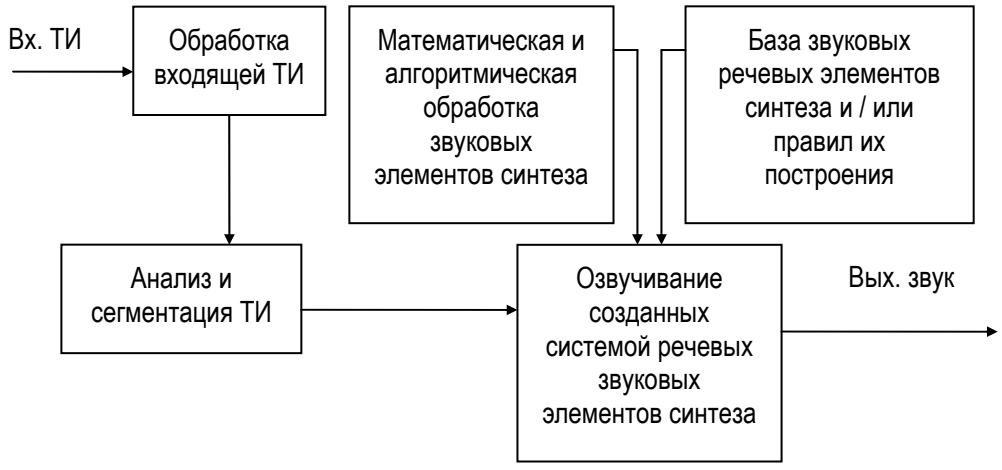


Рисунок. Этапы работы конкатенативной системы озвучивания текстовой информации согласно концепции TTS синтеза

Особенности конкатенативных TTS систем

Системы конкатенативного синтеза оперируют минимальными речевыми данными – конкатенируемыми элементами синтеза. Здесь принципиальным есть выбор элементов синтеза, от которых будет зависеть естественность звучания, разрывность и разборчивость синтезированной речи.

Такие системы имеют ряд особенностей:

1. Удобство получения информации для озвучивания – данные поступают в виде текстовой информации, которую можно произвольно обрабатывать – размечать, сегментировать, структурировать и пр.

2. Относительная простота концепции – происходит лишь сегментация и конкатенация текстовых и, соответственно, звуковых данных.

3. Математические методы обработки стыков конкатенированных речевых сигналов, с одной стороны, имеют общую основу, а с другой – под каждую конкретную систему разрабатываются отдельно, с учетом ее специфики. Это еще больше повышает естественность звучания синтезируемой речи.

4. Высокий уровень естественности звучания синтезированной речи объясняется тем, что в основе синтеза лежат естественные, произнесенные диктором элементы речи.

5. Высокая скорость работы систем конкатенативного синтеза возможна благодаря мощности современного аппаратного обеспечения и развитию программного, что позволяет осуществлять операции конкатенации цифровых звуковых речевых сигналов с их последующим озвучиванием в режиме реального времени.

6. Удобство обработки и доступность элементов синтеза. Поскольку, все элементы синтеза представляют собой цифровые звуковые файлы, это дает возможность обрабатывать (нормализировать амплитудную составляющую, удалять шумы из рабочего сигнала, изменять частотный спектр и т.п.) конкретные сигналы без угрозы изменения всех речевых данных системы в целом.

7. Структура модуля конкатенации определяется размером и размерностью базы данных естественных звуковых речевых элементов, поскольку, непосредственно зависит от естественных данных, тем самым обеспечивая высокую естественность звучания сгенерированной речи. Поэтому при повышении естественности звучания синтезированной речи будет возрастать и размерность элементной базы синтеза [Hess, 1992].

8. Системы конкатенативного синтеза манипулируют речевыми сигналами как совокупностями речевых элементов. Поэтому для повышения уровня естественности звучания сгенерированных звуковых сигналов необходимо такую структуру сегментации / конкатенации естественных речевых элементов, которая бы учитывала и использовала признаки естественности звучания речевых сигналов [Крак, 2005].

9. Поскольку при синтезе речевых сигналов непосредственно используются реальные речевые данные, то для создания и повышения эффективности работы систем сегментации речевой информации необходимо уделить особое внимание исследованию языковедческой теории – разделам фонетики и словообразования [Крак, 2005].

Выбор объектов и элементов синтеза, их структура и свойства

Задачу синтеза естественной речи в ключе конкатенативного TTS синтеза можно рассматривать как задачу озвучивания произвольной текстовой информации, что, с одной стороны, несколько упростит постановку, а с другой – позволит применить большее количество методов цифровой обработки сигналов, учесть многие результаты фонетических исследований[8] на уровне конкретных речевых звуковых сигналов.

Произношение слов изменяется в направлении все большего отождествления их с буквами. Это обусловлено тем, что в памяти человек сохраняет слова и представляет их не столько в звуковой, сколько в графической оболочке – этому способствует процедура приобретения знаний (общеобразовательная и высшая школа), где необходимо больше читать и писать, чем говорить [Ющук, 2004].

[8] В работе используются результаты фонетических и морфологических свойств и характеристик для анализа украинской речи.

Объектом синтеза будем рассматривать слово в связи с тем, что в описанной задаче озвучивания произвольной текстовой (словесной) информации минимальным семантически полным носителем информации является именно слово. Это подсказывают и результаты фонетических исследований [Кривонос, 2006].

Акустические характеристики гласных слова и соотношение гласных под ударением и безударных значительно отличается от соответствующих показателей слов изолированных. Поэтому следует дифференцировать ударение организованное изолированным словом, и ударение слова в предложении, т.е. фразовое ударение [Ющук, 2004].

В данной работе предлагается общая структура представления объектов синтеза, их свойств и характеристик.

1. Общие грамматические характеристики слова, отображающие правила его написания и пр.

2. Наборы гласных, согласных, йотированных, смягченных звуков, мягких знаков, апострофов и их количество, соответствующие слову. Эти характеристики позволят описать (а при построении модели синтеза и использовать) фонетические свойства звуков речи. Так, известно, что:

– произношение гласных зависит, прежде всего, от позиции в слове и их артикуляционных характеристик и в незначительной мере от соседства с теми или другими согласными звуками;

– произношение согласных зависит от звукового окружения и предопределяется их артикуляционными характеристиками.

4. Набор элементов синтеза составляющих слово. Это могут быть как «классические» элементы синтеза (фоны, слоги, дифоны, трифоны), так и какие-то другие, выбранные специально для конкретной системы озвучивания. Элементы синтеза, в свою очередь, обладают рядом свойств и характеристик, которые будут рассмотрены далее.

5. Набор времени звучания элементов синтеза и общее время звучания слова. В естественной речи каждое слово и его составляющие (в задаче синтеза речи это элементы синтеза) имеют строго определенное время звучания и для естественности звучания это следует учитывать.

6. Набор звуковых файлов определенного размера содержащих элементы синтеза. Выделение и использование данных сведений направлено на анализ и оптимизацию работы с оцифрованными звуковыми речевыми сигналами, возможностью упрощения их перекодирования в разные цифровые форматы звуковых данных.

7. Частотные характеристики использования слова в речи и частотные характеристики элементов синтеза для данного слова.

Существует несколько стандартных подходов к выбору концепции формирования элементов синтеза: выбор фонов, дифонов, слогов и трифонов.

Поскольку в естественной речи не все звуки являются вокализированными, то часто используют гибридный дифонно-силябулярный конкатенативный синтез, где часть элементов является дифонами, а часть – слогами [Hess, 1992].

Для конкатенативных систем с практически всеми (кроме фонов) фонетическими единицами в качестве минимальных элементов синтеза характерно присутствие «вспомогательных» единиц, таких как начальные и завершительные полуфоны – акустические эквиваленты начальных и завершительных частиц фонем.

Таким образом, выбор элементов синтеза представляет собой сложную задачу, требующую тщательных языковедческих исследований. Так, следует учитывать большое количество фонетических коартикуляционных свойств элементов речи (следовательно, элементов синтеза речи), в частности таких как:

– произношение гласных в зависимости от их расположения в слове;

– ударение гласных;

– акустические особенности и характеристики гласных изолированных, в слове и в предложении;

– произношение согласных в зависимости от их расположения в слове, смягчение, удвоение и пр.;

– влияние акцентированности и ударения на частотные характеристики речи;

– движение и характеристики частоты основного тона;

– суперсегментные явления;

– структуризацию пары «объект синтеза – элемент синтеза»;

– просодию, средства просодии и просодические характеристики;

– интонационные свойства естественной речи[9] и пр.;

Для учета выше перечисленных свойств в данной работе предлагается следующее общее структурное представление элемента синтеза как составляющего системы озвучивания речи.

1. Текстовый вид элемента синтеза – совокупность текстовых элементов синтеза (это могут быть фонемы, дифоны, слоги, трифоны и пр.) должна формировать полную выборку слов [Кривонос, 2005].

3. Цифровой звуковой сигнал, содержащий элемент синтеза. Записанную звуковую информацию следует хранить с высоким качеством в удобном для последующей обработки формате цифровых звуковых данных. Также необходимо провести фильтрацию избыточной информации сигналов и нормализацию звуковых элементов синтеза. Это принципиально повысит качество естественности звучания и разборчивость синтезируемой речи.

4. Тип элементов синтеза. Согласно предварительным фонетическим и морфологическим исследованиям, необходимо провести строгую типизацию элементов синтеза по таким характеристикам:

– расположение в слове;

– ударность / безударность гласного звука элемента (если присутствует);

– свойства согласного звука элемента (если присутствует);

– сегментные свойства согласных и пр.

5. Время звучания элемента синтеза. На этапах создании и использовании звуковых элементов синтеза следует учитывать время звучания элемента при произношении их диктором в естественных условиях.

6. Звуковой файл, содержащий элемент синтеза. Выделение и использование данных сведений направлено на создание статистических данных о конкретном элементе и базы всех элементов синтеза.

7. Фонетические, морфологические и акустические сведения об элементах синтеза.

Особое внимание следует уделить исследованиям фонетических, морфологических, коартикуляционных, интонационных, просодических свойств речи для их использования при построении систем озвучивания информации.

Заключение

Рассмотрена задача создания систем озвучивания информации, поданной в текстовом виде – посредством конкатенативного Text-to-Speech синтеза. Для этого выполнен обзор основных этапов проблемы, выделение и анализ основных подзадач. Описаны особенности такого подхода относительно решения проблемы – простота концепции, высокий уровень естественности звучания синтезированной речи, высокая скорость работы систем конкатенативного синтеза. Исследованы фонетические свойства и характеристики синтезируемой речи. Предложено структурное представление объектов и элементов синтеза. Проведен анализ их структуры, свойств и характеристик. Акцентирована необходимость анализа интонационных и просодических характеристик украинской речи для задачи синтеза речи. Уделено внимание большому количеству проблем, связанных с фонетическими коартикуляционными свойствами элементов речи.

Библиография

[Dutoit, 1993] Dutoit and Leich H. MBR-PSOLA: Text-to-speech synthesis based on an MBE re-synthesis of the segments database // Speech Communication. – 1993

[Cox, 2000] Cox R.V., Kamm C.A., Rabiner L.R., Schroeter J. Wilponl J.G. Speech and language processing for next-millennium communications services // Proc. Of the IEEE. – 2000. – 88, N. 8.

[Hess, 1992] Wolfgang Hess Speech synthesis – a solved problem? // Signal Processing VI. – 1992. – P. 37–46.

[W3C] http://www.xmlhack.ru/articles/03/02/05/vxml20.html

[9] Теоретическое и практическое значения интонации как формулирующего элемента и структурного компонента звуковой речи является очень важным. Интонацию определяют как «фразовую фонетику», «синтаксическую фонетику», как суперсегментный уровень речи; менее распространенным есть ее определение как ритмомелодики и просодии [Багмут, 2004].

[Багмут, 2004] Багмут А. Українська інтонологія: проблематика досліджень // Українське мовознавство – К.: Національний університет імені Тараса Шевченка, 2004. № 32–33. – С. 36–41

[Крак, 2005] Крак Ю.В., Шатковський М.М. Сегментація мовної інформації для задач автоматичного озвучення // Вісн. Київськ. ун-ту. Сер. фіз.-мат. наук. – К.: Національний університет імені Тараса Шевченка, 2005 – Вип. 4. – С. 254-258.

[Кривонос, 2005] Кривонос Ю.Г., Крак Ю.В., Шатковский Н.Н. Анализ структуры задачи создания систем озвучивания текстовой информации // Компьютерная математика, 2005. – № 3. – С. 87–95

[Кривонос, 2006] Кривонос Ю.Г., Крак Ю.В., Шатковский Н.Н. Структура, свойства, характеристики объектов и элементов синтеза речи // Компьютерная математика, 2006. – № 1. – С. 61–69

[Ющук, 2004] Ющук І. Фонетичні закони й орфоепія // Українське мовознавство. – К.: Національний університет імені Тараса Шевченка, 2004. № 32–33. – С. 16–22

Информация об авторах

Кривонос Ю.Г. *– Институт кибернетики В.М.Глушкова НАН Украины, зам. директора, 40, пр-кт Академика Глушкова Киев-03187, Украина, e-mail: aik@public.icyb.kiev.ua*

Крак Ю.В. *– Национальный университет имени Тараса Шевченко, профессор, 6, пр-кт Академика Глушкова Киев-03187, Украина, e-mail: krak@unicyb.kiev.ua*

Шатковский Н.Н. *– Институт кибернетики В.М.Глушкова НАН Украины, м.н.с., 40, пр-кт Академика Глушкова Киев-03187, Украина, e-mail: nicolo@icyb.kiev.ua*

FINDING OF INFORMATIVE PARAMETERS DESCRIBING BIOMEDICAL POPULATIONS

Mykola Budnyk, Igor Voytovych

Abstract*: Problem of classification one population onto two classes is often used in biological and medical studies. Approach for searching of informative (valuable) parameter, based on integral probability distribution function, has been proposed. It was shown that binary (threshold) decision rules could be generalized by involving of risk zone. Examples of processing of empirical data for clinical group including healthy persons and patients with coronary artery disease, observed by method of magnetocardiography, were presented.*

Keywords*: diagnostic test, generalized parameters, binary decision rule, risk zone, fuzzy logic.*

ACM Classification Keywords*: G.3 Mathematics of Computing – Probability and statistics, J.3 Computer Applications – Life and medical sciences.*

Introduction

Problem of classification of given subject from biological or medical (BioMed) population onto two classes is part of more general problem of Decision Making (DM). This is the simplest example of minimal number of groups. Solution of this problem within particular application area consists in synthesis of appropriate Decision Making Algorithm (DMA). Specific features of BioMed (more generally – living) population is that it is formed by elements of statistical nature. Therefore, characteristics of probability distribution function (PDF) with respect to any experimentally observed parameter are essentially influenced by many another external non-informative factors. As a result, wide blurring, asymmetry, excess, up to non-unimodal form, i.e. differences of empirical PDF from gauss-like form are take place. All that do generating of reliable BioMed DMAs are very difficult, because above factors are, in fact, the disturbed degree of freedom which are equivalent to strong external noises in physical sciences.

The quality of group discrimination is determined primarily by the quality of describing parameters. Therefore, the search for informative (diagnostically significant) parameters, i.e. identification of populations, is crucial. Basic hazards, for instance, of ischemic heart disease (IHD), are posed by sex, age, genetic factor, hypercholesterolemia, arterial hypertension, smoking habits, excessive body weight, and inactive lifestyles. They may cause directed effect that shifts informative parameter towards IHD, which ultimately will bring about asymmetry in PDF. Other PDF's deviations from Gauss form may also be caused by "uncompensated" impact of external factors.

1. Figures of Merit (FOM) of Medical Test

Diagnostic value is determined generally by such 4 indicators as Specificity (Sp), Sensitivity (Sn), Negative prognostic value (NPV), Positive prognostic value (PPV), yet depending on applications (screening, diagnosing or evaluation of treatment effectiveness) a certain index is decisive. Here is considered the simplest and the most practiced case of the one-dimensional parametric (sample) space when division is made with just a single parameter, and the simplest threshold Decision Rule (DR) dividing a group by two - negative and positive classes - depending on the side of a critical value (threshold) which the patient's parameter takes. Here occur the errors of the first kind (missing a target – false negative (FN), α) and second kind (false alarm – false positive (FP), β).

$$\text{Binary DR } (X_{CR}) = \begin{cases} \text{Negative class,} & \text{if } X < (>) X_{CR} \\ \text{Positive class,} & \text{if } X > (<) X_{CR} \end{cases} \quad (1)$$

Assume the objective is to divide a group of persons consisting of samples of healthy individuals N_A in number and patients N_B in number by two – positive and negative classes. Use of threshold DR thus resulted in a(b) true negative (TN) and true positive (TP) persons and c(d) of FN(FP). Therefore, $N_B=b+c$, $N_A=a+d$ and all probabilities are normalized by the total $N=N_A+N_B$. The respective FOMs are as follows:

$$Sp=1-\beta,\ Sn=1-\alpha,\ \alpha=d/N_A,\ \beta=c/N_B,\ NPV=C/(C+\alpha n),\ PPV=Ч/(Ч+\beta/n),\ n=p/(1-p),\ p=N_B/N. \quad (2)$$

Sn and Sp demonstrate a probability of attributing persons to a "true" class, i.e. healthy individuals to class 1 and patients to class 2; they require a priori knowledge of groups and, consequently, are the indicators of value of the **direct** discrimination problem solution (DPS) i.e. the quality of training. NPV and PPV demonstrate a probability that persons attributed to a certain class comprise the "true" group, i.e. class 1 to healthy individuals and class 2 to patients and require a posteriori probability of a correct classification; they are thus the indicators of the value of **inverse** DPS, i.e. the quality of an **exam**.

From (2) it is seen a major failure of inverse DPS by means of classic Bayesian approach being a dependence of prognostics on the prevalence p(2) which is in essence a proportion of a number of persons (a priori probabilities) in training groups $n=N_B/N_A$. Although, both in an exam mode (so called a "blind test") and in an operational mode there is no a priori information **in principle**.

2. Principle of Discrimination of Two Group

Application of a PDF density by contrast with the PDF is an essential failure of the approach set forth in (2). An advantage of integral PDF results from Glivenko-Kanteli theorem, according to which an empirical PDF with probability 1 is approaching an actual PDF. There is no similar theorem for the density, and thus defining empiric PDF density is more complicated task since the precision of its simplest non-parametrical estimates (bar chart, polygon of frequencies) are depending on dividing an empiric parameter range into intervals. With deviation from normality empiric estimates (descriptive statistics) are much dependant on the PDF density form. E.g., whereas estimation of a difference of the average values of the two groups becomes unstable, hence it is necessary to turn to describing a difference of medians. In this connection authors believe that application of integral PDF is a means of **noise regularization** (authors believe it is the external non-informative factors – see Introduction).

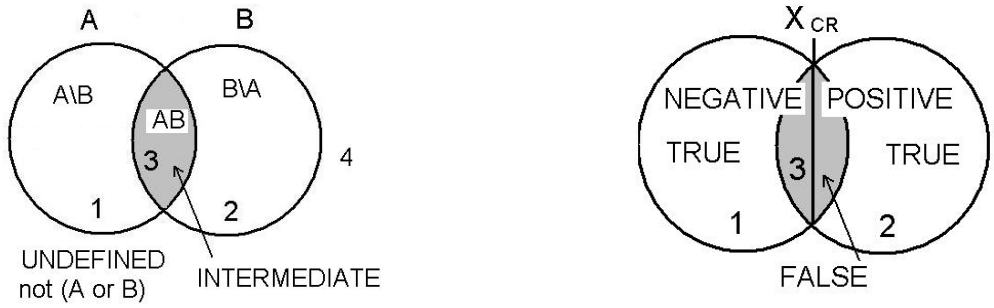


Fig.1 Recognition between two groups according to probability (left) and statistical (right) decision theories.

Fig. 1 shows schematic difference between the probability theory and statistic theory based on binary DR. In general, dividing two groups may result in 4 classes. Expression (3) describes it in terms of the probability theory and (4-5) for binary DR.

$$P(1)=P(A\backslash B)=P(A)-P(AB);\quad P(2)=P(B\backslash A)=P(B)-P(BA);\quad P(3)=P(BA),\quad P(4)=P(\text{not (A or B)})=1-P(A\vee B). \tag{3}$$

$$P(1) = P(TN) = a/N;\quad P(2) = P(TP)= b/N,\quad P(3)=P(False)=P_\alpha+P_\beta,\quad P_\beta = c/N,\quad P_\alpha = d/N. \tag{4}$$

$$P(4) = \begin{cases} 0, \text{ within Discrimination problem, because } P(1)+P(2)+P(3)=1 \\ P(Other) \text{ or } P(Undefined) \text{ within Classification problem} \end{cases} \tag{5}$$

Comparing (4) and (3) we see that DMA based on binary DR brought about the loss of symmetry which occurs in the probability theory, i.e. P(AB)=P(BA) since the probabilities FN and FP of the classification are **different** in general $P_\beta \neq P_\alpha$; because c≠d. It is thus necessary to develop DMA so as to find the threshold X_{OPT} which equals the probabilities of errors of both kinds. It corresponds to the line X_{CR} is changed with some **optimal** line, in general a **curve** X_{OPT} which splits a space of both events so as their resulting spaces would be equal. The obtaining procedure with applying integral PDF instead of differential PDF is shown below.

3. Approach Based on Integral PDF

So, task to be solved consists in unification of data processing so that comparison DM quality between two groups, which are non-homogeneous with respect to non-controlled disturbed factors, have been made. Three methods of discrimination of two groups were compared:

1) traditional method based on differential PDF;
2) method based on non-normalized and non-smoothed integral PDF;
3) method based on standardized (normalized and smoothed integral) PDF.

Experimental data include 151 examined patients formed two groups: 70 healthy persons and 81 CAD pts with unchanged ECG at rest. Well-known, that CAD is the most widespread heart disease and reliable CAD diagnostics is still actual problem. The reason consists in that rest ECG is normal, EchoCG and MRT is mainly limited by morphological heart structure (but did not links with electrophysiology), and various stress tests can be prone to risks.

From the other hand, Magnetocardiography (MCG) is a method of non-invasive recording and analysis of the magnetic field of the heart, arising due to its electrical activity. The advantages of the MCG-mapping consist in: 1) localization of electrical source into myocardium, 2) high sensitivity to various pathological disturbances, 3) revealing of disease at early stages and silent forms, 4) safety [1]. Last decade considerable efforts are directed toward the studying of MCG to CAD diagnostics [1].

MCG observations were performed in Biomagnetic Lab (Institute for Cardiology, Kyiv) by 4-channel magnetocardiograph at unshielded environment [2] according to standard method [1]. Medical analysis was carried out with help of 9 numerical MCG indexes reflecting spatial-temporal structure of magnetic field, see for detail [1,3]. The control group consisted of volunteers with no indications for cardiac diseases found by routine clinical methods [4]. MCG was recorded with a 7-channel magneto-cardiograph CARDIOMAG (Glushkov Institute for Cybernetics, Kyiv, Ukraine) [4]. Additionally, routine clinical examinations (ECG, EhoCG, bicycle test) were conducted. The results of discrimination are presented in Tbl.1 and Fig.2.

Table 1 Results of DPS of two groups based on integral PDF and binary DR

			Non-normalized and non-smoothed PDF						Standardised PDF		
Index	M_{Hel}	M_{CAD}	X_{CR}	SP	SN	NPV	PPV	V_{AVE}	X_{CR}	α	V
Nst	3,68	12,75	7	69%	64%	62%	71%	66,8%	6,37	0,311	68,9%
MR	7,75	11,09	10	67%	67%	64%	69%	66,6%	9,61	0,314	68,6%
IFV	1,9	2,86	2,55	67%	66%	63%	69%	66,2%	2,34	0,343	65,7%
IFH	4,85	6,91	6,2	64%	61%	59%	66%	62,5%	5,84	0,351	64,9%
Y	10,84	12,74	12,1	58%	69%	62%	65%	63,5%	11,86	0,392	60,8%
Nj	3,5	4,16	4	51%	66%	56%	61%	58,6%	3,84	0,414	58,6%

Evaluation of diagnostic valuable of indexes were performed with help of "average value" V_{AVE} and value V (3) if non- or normalized PDFs are used, respectively. Here α is probability of classification errors (within framework of this method errors of both kinds are equal).

$$V_{AVE} = (SP+SN+NPV+PPV)/4, \qquad V = 1 - \alpha. \qquad (6)$$

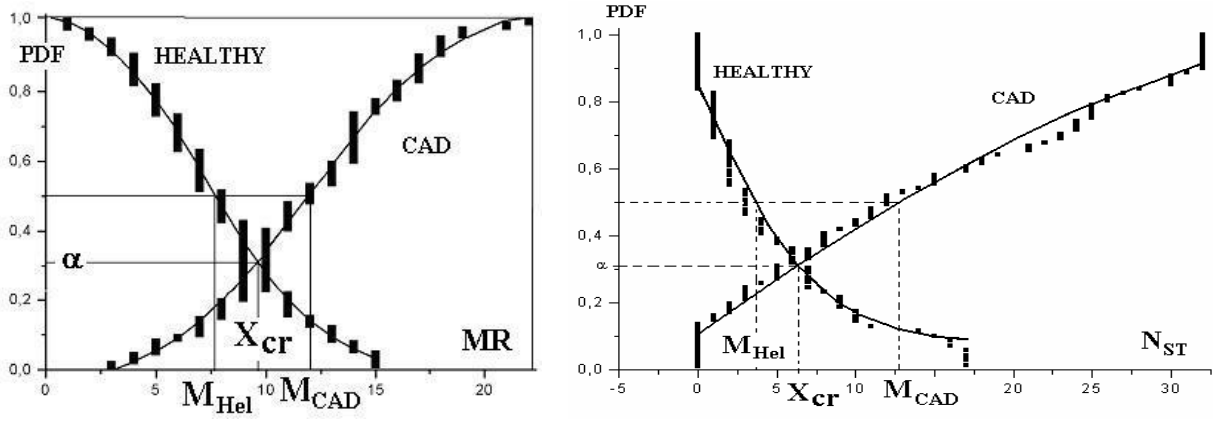


Fig. 2 Discrimination of two groups based on smoothed PDF for quasi-gauss (left) and non-gauss (right) case.

Table1 shows that both FOMs V_{AVE} and V are very close so that ordering of best parameters according to any of them are practically identical. It means that method, based on standardized PDF, is insensitive to any disturbed factors and value V should be considered as adequate and stable FOM for quality (power, reliability) of discrimination problem solution (DPS). Otherwords, method of DPS, utilising normalised PDF, may be used in medical statistics if real data are blurrred, non-gauss, and asymmetrical [5].

This approach being optimal is confirmed by the ROC-analysis (Fig.3). It is seen that there is an optimal value of prevalence when Sp=Sn (point C, Fig.3). Yet the best division of two groups is accomplished for equal groups, i.e. n=1 (point D, Fig.3), although in such case the error probabilities will differ Sp≠Sn. In order to reach maximum with Sp=Sn errors of the 1st and 2nd kind should be equal. Consequently, it is necessary to look for the DPS

technique so as both errors would be equal. From Fig. 4 it is seen that applying the normalized PDF automatically ensure the equality α=β та Sp=Sn (i.e. points C and D are converged). The advantage here is that such technique does not need equal groups in practice since it is done in a pure mathematic way – by normalizing the empiric PDF.

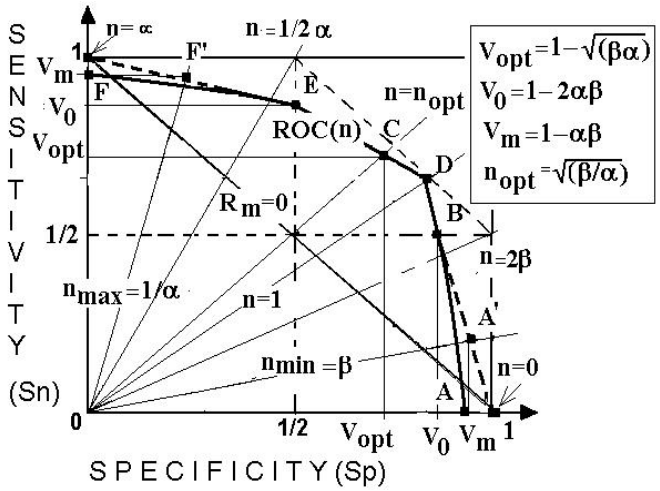


Fig. 3. Parametric dependence of SN from SP, so called ROC curve based on using of non-normalized PDF

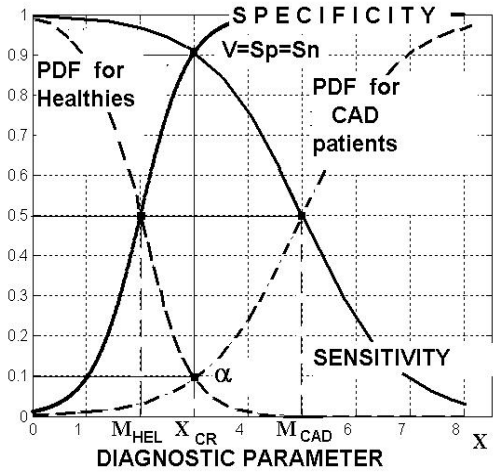
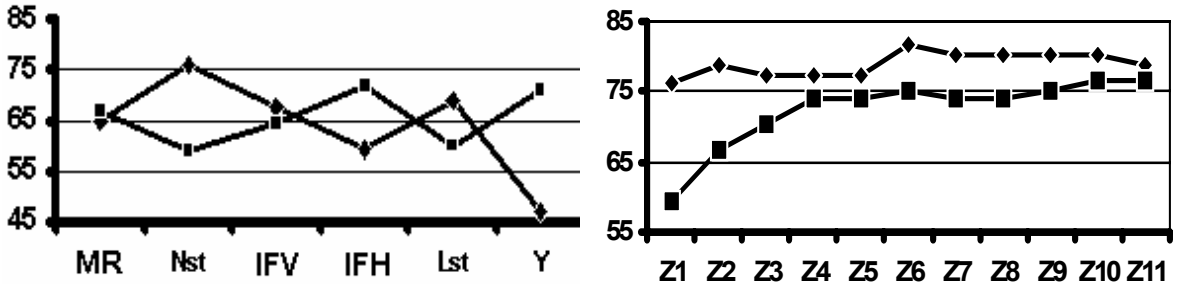


Fig. 4. Dependence of Sn and Sn from threshold of binary DR for normally distributed groups.

4. Generalized Parameters

To implement some DMA in practice, it is necessary to develop a method to DPS allowing to select reliable discriminant parameters having too large (80% and above) FOM, i.e. V according to Sec.1.3. But from Tbl.1 we can see that anyone single parameter has no high value, because V<70%. Low reliability of CAD recognition necessitates the use some generalized (integral) indices instead particular ones. For this purpose, we employed linear discriminant analysis (LDA) and studied the dependence of the discrimination power on various combinations of MCG indices in the LDA space [6]. In such a way so-called cumulative parameters have been introduced into consideration. Cumulative i-th parameter Z_i is sum of first particular parameters, arranged into descending order by significance level of two-population T-test.



A) B)

Fig.5. Dependencies of LDA power for particular (a) and cumulative (b) parameters (♦- healthy, ■ – CAD).

Firstly it was calculated 10 MCG parameters MR, IFH, IFV, N_{ST}, L_{ST}, AFV, AFH, Y, N_T, N_J, characterizing IHD [1]. The best 6 of them are X1-X6 (N_{st}, MR, IFH, IFV, L_{st}, Y) for which p<0.01 is shown in Fig. 5a. From the figure it is seen that the dependence of discrimination power is irregular since different are the force and nature of the impact of external uncontrolled factors on statistic distribution of each parameter. On the other hand, for cumulative parameters discrimination power is gradually increasing for both healthy individuals and IHD patients (Fig. 5b). Consequently, 6 variables suffice for the best division of healthy individuals/IHD patients based on MCG

indexes, since the curve maximum for healthy individuals (81.7%) and local maximum for IHD patients 75.3%) are seen with the same 6 parameters.

Yet ordering of 6 indexes according to the T-test in cumulative order is not optimal if PDF are much different from Gauss form. Therefore all possible combinations C_6^2, C_6^3, C_6^4, C_6^5 were studied. Results are set forth in Table 2.

Table 2. Optimized order in sets of parameters presenting the best division of groups.

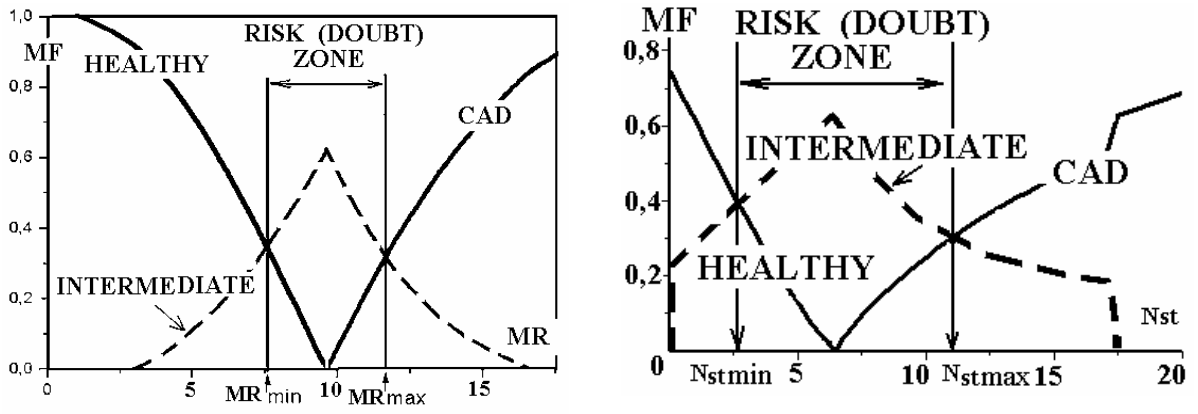
Group	Number of parameters into set				
	1	2	3	4	5
Healthy	N_{ST}	N_{ST}-α_{ST}	MR- Y-N_{ST}	MR-α_{ST}-N_{ST}-Y	N_{ST}-Y-α_{ST}-IFH-MR
Power	76,05	80,28	80,28	80,28	81,69
CAD	IFH	MR-IFV	MR-IFV-N_{ST}	MR-N_{ST}-α_{ST}-IFV	N_{ST}-Y-α_{ST}-IFH-MR
Power	71,6	71,6	70,37	75,3	75,3

It is seen from table 2 that the best power with the set of 5 parameters is increasing for Healthy individuals from 77.46% to 81.7%, for IHD patients – from 74.07% to 75.3%. As a result of combinatory search of about 5 parameters the LDA power corresponding 6 parameters ordered in a nonoptimal way was obtained Cumulative parameter formed by 5 parameters having maximal discrimination power for CAD pts against Healthy (75%), and for healthy against CAD (82%) was founded [6].

Thus non-homogeneity of groups was actually removed from the LDA viewpoint. Identity of set of 5 parameters proves *homogeneity* of both groups in terms of a discrimination task. Here worsening the quality of discrimination caused by the *effect of external uncontrolled factors* is the lowest for both groups simultaneously. Of course, the values of maximal LDA powers are not equal since the forces of effect of the said factors on each group are different.

5. Calculation of Risk (Intermediate) Zone based on Fuzzy Logic

In using a threshold DR the DPS error increases for persons whose parameter value is close to the threshold, i.e. on the border where training statistic samples overlap. It especially affects the quality of DPS for the research methods sensitive to the impact of external non-informative factors [2]. It is necessary in this connection to identify an intermediate zone of the parameter corresponding to the risk of a certain diseases. An example of an IHD risk zone identified by means of MCG parameters, whose PDF is shown in Fig. 2, is set forth in Fig. 6. It is seen that for quasi and for essentially non-Gauss distribution a risk zone is determined unambiguously because of PDF graphs are monotonous.



A) B)

Fig.6. Determination of intermediate zone for parameters MP (a) and Nst (b) in which risk of CAD is predicted.

This approach is based on the transition from the PDF to membership functions (MF) of F applied in fuzzy logic which are defined as follows:

$$F(1)= P(TN) - P_{\alpha}; \quad F(2) = P(TP) - P_{\beta}, \quad F(Intermediate)=F(Risk)+F(Undefined), \tag{7}$$

$$F(Risk)=F(3)=2*P(False)=2*(P_{\alpha} + P_{\beta}), \quad F(Undefined)=F(4)=1- [F(1)+F(2)+F(3)]. \tag{8}$$

Conclusion

In result, method to find numerical informative parameters, i.e. identification procedure describing some complex living object. For example, parameters, containing diagnostic information about the failures of electric processes into the human heart, have been considered. Such parameters have been studied at of patients with coronary artery disease (CAD) by means of method of magnetocardiographic (MCG) mapping. High sensitivity of the MCG leads to essential influences of non-controlled external factors resulting in increasing of blurring of observed groups. That is why selecting and ordering of reliable MCG indexes are too complex task. In order to improve quality of data processing, an approach, based on integral probability distribution function (PDF), have been proposed. In result, single figure-of-merit (FOM), so-called “Significance”, instead 4 ones (sensitivity Sn, specificity Sp, negative (NPV) and positive (PPV) predictive value) have been found. A selection procedure for generalized parameters with LDA involved was proposed and intermediate zone, i.e. the range of parameter values where there is the risk of disease, was determined.

Acknowledgements

Results have been partially completed owing to financial support of the Science and Technology Center in Ukraine (STCU) under the project 2187. Authors also would like to express their frankly gratitude to Prof. V.P. Gladun (Kyiv, Ukraine) for help and discussion.

Bibliography

[1] Diagnostic criteria for chronic coronary artery disease based on registry and analyses of the magnetocardiograms /Budnyk M, Chaikovsky I. Kozlovsky V et al // Preprint 2002-5, Institute for Cybernetics, Kyiv (Ukraine), 2002.

[2] Low-cost 7-Channel Magnetocardiographic System for Unshielded environment / Budnyk M, Voytovych I, Minov Yu et al // Neurology and Clinical Neurophysiology, 2004:112:1-7. http://www.ncnpjournal.com

[3] Evaluation of Magnetocardiography Indices in Patients with Cardiac Diseases / Budnyk M, Kozlovsky V, Stadnyuk L et al // Neurology and Clinical Neurophysiology, 2004:111:1-6. http://www.ncnpjournal.com

[4] Supersensitive MCG system for early identification and monitoring of heart diseases (medical application) / Voytovych I, Kozlovsky V, Budnyk M et al // Control Systems and Machines, 2005, No 3, p.50-62.

[5] Budnyk M., Zakorcheny O., Ryzhenko T., Zholob V., Evaluation of valuable numerical indexes derived from MCG data. In: Proc. Intern. Conference ICAP’2005, Kyiv (Ukraine), 2005, p. 170-171.

[6] Chernysheva D, Budnyk M. Using of discriminant analysis for processing of magnetocardiographic information. In: Computer tools, system and nets. Ed. O.V.Palagin Glushkov Institute for Cybernetics. Kyiv (Ukraine), 2004.

Authors' Information

Mykola Budnyk – *Doctor of Engineering Sciences, Senior Research Scientist; Glushkov Institute for Cybernetics of NAS of Ukraine, Glushkov ave. 40, 03187, Kyiv, Ukraine; e-mail: d220@public.icyb.kiev.ua*

Igor Voytovych – *Prof., Corresponding Member of NAS of Ukraine, Head of Department, Glushkov Institute for Cybernetics of NAS of Ukraine, Glushkov ave. 40, 03187, Kyiv, Ukraine; e-mail: d220@public.icyb.kiev.ua*

TEACHING FRAMEWORK FOR KNOWLEDGE ENGINEERING COURSE

Tatiana Gavrilova, Seppo Puuronen

***Abstract**: The paper presents experience in teaching of ontological engineering. The teaching framework is targeted on the development of skills that will allow facilitating the process of knowledge elicitation, structuring and ontology development for scaffolding students' research. The structuring procedure is the kernel of ontological engineering. The 5-steps ontology designing process is described. Special stress is put on "beautification" principles of ontology creating. The curriculum includes interactive game-format training of lateral thinking, interpersonal cognitive intellect and visual mind mapping techniques.*

***Keywords**: ontologies, knowledge engineering, analytical skills.*

***ACM Classification Keywords**: I.2 Artificial intelligence*

Introduction

Young researchers of computer science and information technologies use a lot of informal rules-of-thumb advice that may help but not a systematic guidelines. Major recommendations deal with library work, citation and other important but not essential components of research activity. Web resources now help to streamline the process of finding, selecting, and entering information from the Web, corporate databases, and reference materials into a paper. We are speaking about not syntax but semantic of study. Our course is aimed at information engineering – structuring and shaping the research framework. This issue has hardly been explored yet.

Students are engaged in various semantically-based activities. Indeed, any conceptual data modeling is a form of informal Semantic Modeling [26]. They do all the "bottleneck" activity including data and knowledge elicitation, structuring, formalizing. But do the do it professionally? They need knowledge engineers' analytical skills.

Knowledge Engineering traditionally emphasized and rapidly developed a range of techniques and tools including knowledge acquisition, conceptual structuring and representation models [1], [24]. But for practitioners it is still a rather new, eclectic domain that draws upon areas like cognitive science. Accordingly, knowledge engineering has been, and still is, in danger from fragmentation, incoherence and superficiality. Still few universities deliver courses in practical knowledge engineering.

This paper describes recent experience in teaching such course. The total number of students that were taught is more than 140.

Teaching Framework

Knowledge Engineering course (KEC) is based on university courses in intelligent-systems development, cognitive sciences, user modeling, and human-computer interaction delivered by authors in 1988-2005. KEC proposes information structuring multi-disciplinary methodology, including the principles, practices, issues, methods, techniques involved with the knowledge elicitation, structuring and formalizing. Emphasis is put not on technologies and tools, but in training of analytical skills. KEC also includes Ontological Engineering that is further development of knowledge engineering towards ontology design and creating.

Students are introduced to major issues in the field and to the role of the knowledge analyst in strategic information system development. Attention is given both to developing inter-personal information communication skills and analytical cognitive creative abilities.

The class feature short lectures, discussions, tests, quizzes and exercises. Lectures are important but the emphasis is put on learning through discussions, simulation, special games, training and case studies. A good deal of the course focuses on auto-reflection and auto-formalizing of knowledge, training of analytical and communicative abilities, discovery, creativity, cognitive styles features, and gaining new insights.

KEC consists of 4 inter-connected modules:

- Getting Started in KE (12 hours),
- Practical KE in depth (12 hours),
- Ontological Engineering (12 hours),
- Business Processes Modelling (12 hours).

Different combination of sub-topics is possible. Fig.1 illustrates the structure.

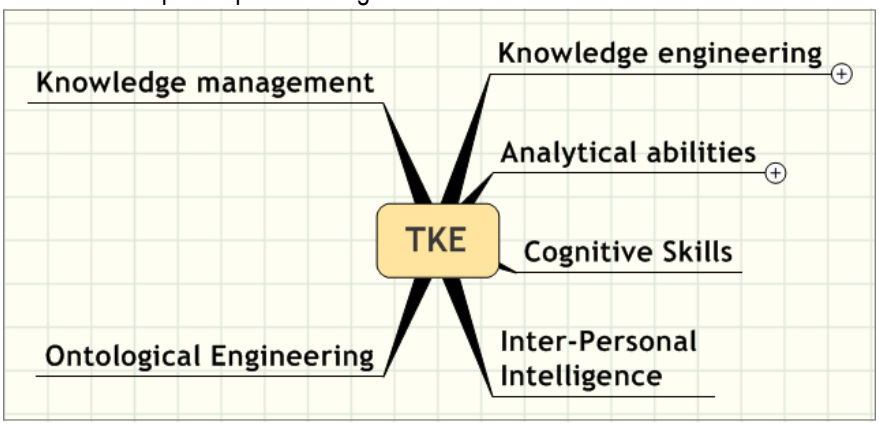


Fig.1. Outline of training on knowledge engineering

The main difference of KEC to existing curricula is cognitive (not technological) bias. Fig.2. shows the main issues covered by tests and practical exercises. Students of IT-departments often under-value the significance of psychological background of categorization, laddering and lateral thinking. But during learning process some of them feel “insight” and become very enthusiastic.

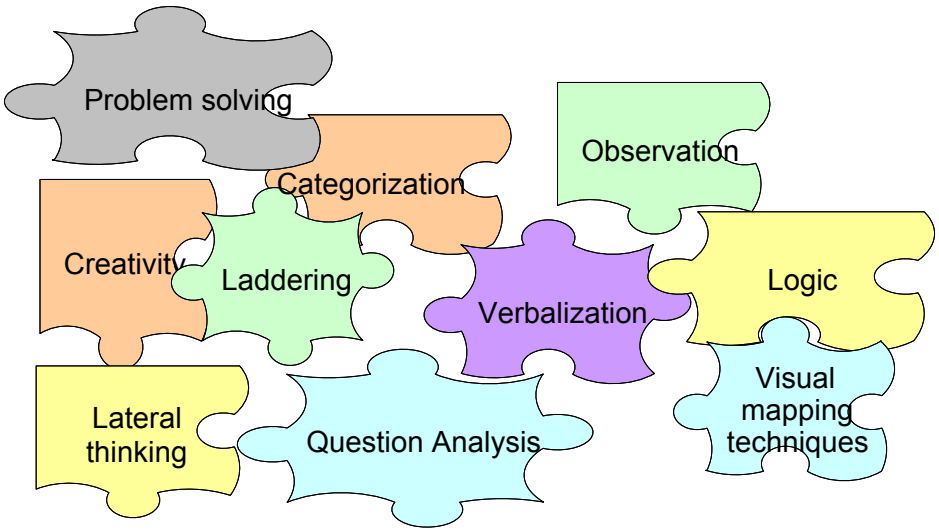


Fig.2. The main topics of practical exercises

The practical methods of knowledge elicitation gain the main interest during teaching as they really work. The training is organized that students study main techniques in pairs “expert – knowledge analysts” with a shift of

roles when needed. Some psychological assessment techniques help students to realize their strong and week points in inter-personal communication and intelligence.

But only knowledge structuring exercises show the importance of obtained analytical skills for the students. Even simple tests from their own research domains are rather difficult at the first classes.

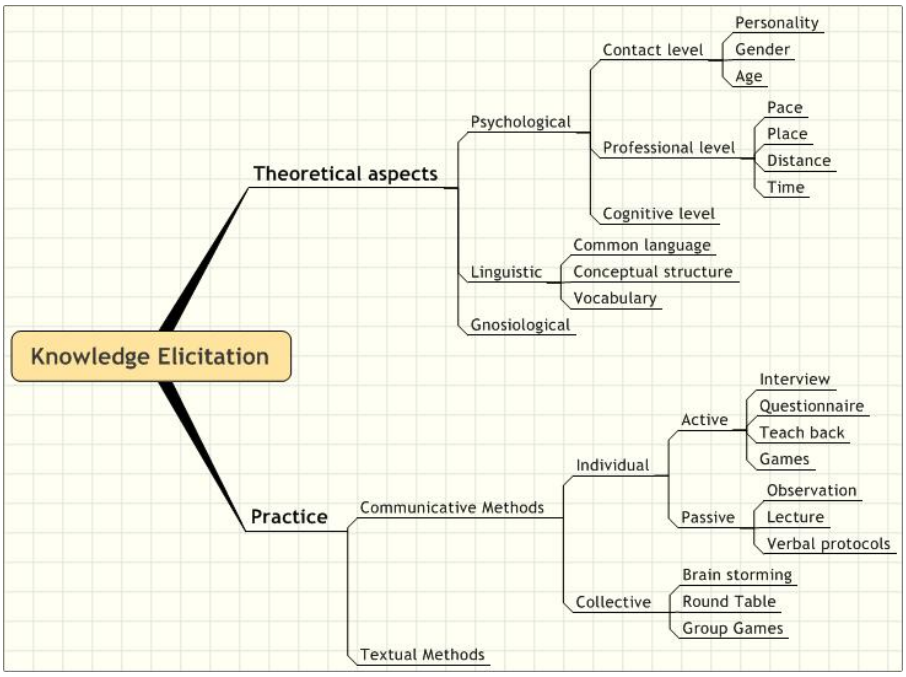


Fig.3. Theory and practice of knowledge elicitation

The study is aimed on semantics not syntax of knowledge engineering. We suppose that systems and languages may be self-studied while general scope and knowledge-stressed approach should be trained thoroughly. Students often under-estimate the role of cognitive styles, verbal skills and logics in information processing. It is supposed to be common sense while it needs to be taught.

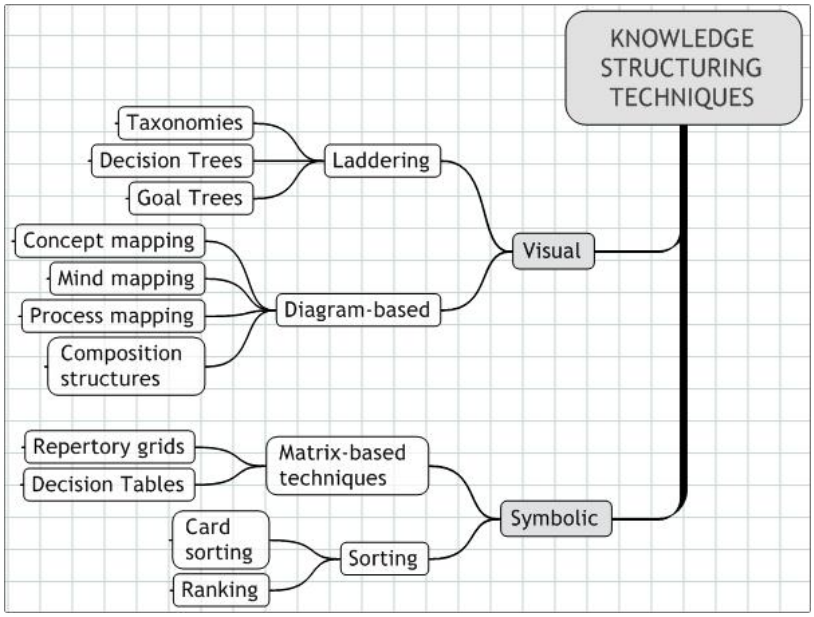


Fig.4. Knowledge structuring techniques

Stress on Ontological Engineering

Ontologies are fashionable now. But our experience in training show that nobody can deal with ontologies without knowledge engineering practice. How to teach ontology design? The theory differs from practical needs…

There are numerous well-known definitions of this milestone term (Gruber, 1993; Guarino and Giaretta, 1998; Jasper and Uschold, 1999; Mizogushi and Bourdeau, 2000; Neches, 1991) but they may be generalized as "Ontology is a hierarchically structured set of terms for describing an arbitrary domain" [9]. In other words, "ontologies are nothing but making knowledge explicit" [12].

Since 2000 a major interest of researchers focuses on building customized tools that aid in the process of knowledge capture and structuring. This new generation of tools – such as Protégé, OntoEdit, and OilEd - is concerned with visual knowledge mapping that facilitates knowledge sharing and reuse [18], [19], [22]. The problem of iconic representation has been partially solved by developing knowledge repositories and ontology servers where reusable static domain knowledge is stored. Ontolingua, and Ontobroker are examples of such projects [20], [21].

Ontology creating also faces the knowledge acquisition bottleneck problem. The ontology developer encounters the additional problem of not having any sufficiently tested and generalized methodologies, which would recommend what activities to perform and at what stage of the ontology development process. The lack of structured guidelines and methods hinders the development of shared and consensual ontologies within and between the specialists. Moreover, it makes the extension of a given ontology by others, its reuse in other ontologies, and final applications difficult [11].

Until now, only few effective domain-independent methodological approaches have been reported for building ontologies [4]; [25], [16]. What they have in common is that they start from the identification of the purpose of the ontology and the needs for the domain knowledge acquisition. However, having acquired a significant amount of knowledge, major researchers propose a formal language expressing the idea as a set of intermediate representations and then generating the ontology using translators. These representations bridge the gap between how people see a domain and the languages in which ontologies are formalized. The conceptual models are implicit in the implementation codes. A re-engineering process is usually required to make the conceptual models explicit. Ontological commitments and design criteria are implicit in the ontology code.

Figure 3 presents our vision of the mainstream state-of-the-art categorization in ontological engineering [12], [13], [28] and may help students and analyst to figure out what type of ontology he/she really needs. We use Mindmanager™ and Cmap™ as they proved to be rather powerful visual tools.

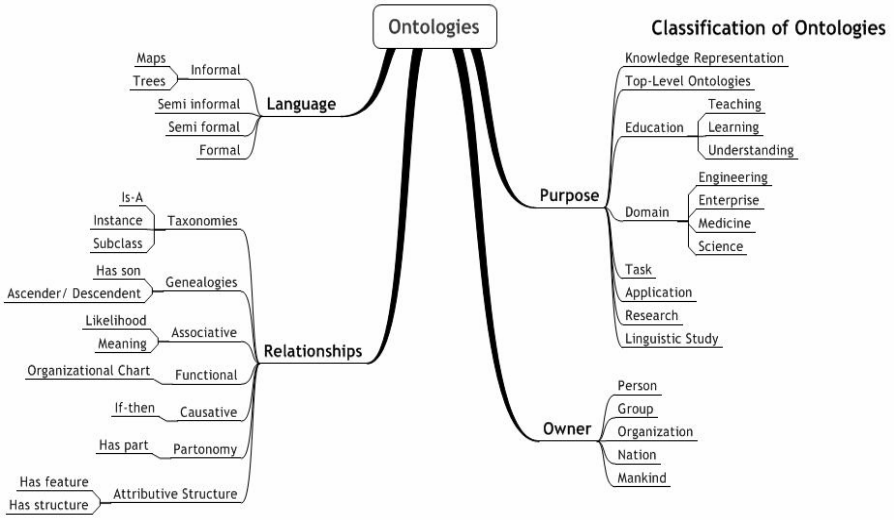


Fig.5. Ontology mind map

We try to simplify a bunch of different approaches, terms and notations for practical use (Fig. 5) and even dare to propose a 5-step recipe, which helps practical ontology design.

Exercises during training help us to evaluate and update it unless it starts to work.

Ontology Design Recipe

The existing methodologies describing ontology life cycle [28], [16], [9] deal with general phases and sometimes don't discover the design process in details. Five simple practical steps were proposed.

Step 1. **Glossary development:** The first step should be devoted to gathering all the information relevant to the described domain. The main goal of this step is selecting and verbalizing all the essential objects and concepts in the domain.

Step 2. **Laddering:** Having all the essential objects and concepts of the domain in hand, the next step is to define the main levels of abstraction. It is also important to elucidate the type of ontology according to Figure 1 classification, such as taxonomy, partonomy, and genealogy. This is being done at this step since it affects the next stages of the design. Consequently, the high level hierarchies among the concepts should be revealed and the hierarchy should be represented visually on the defined levels.

Step3. **Disintegration:** The main goal of this step is breaking high level concepts, built in the previous step, into a set of detailed ones where it is needed. This could be done via a top-down strategy trying to break the high level concept from the root of previously built hierarchy.

Step4. **Categorization:** At this stage, detailed concepts are revealed in a structured hierarchy and the main goal at this stage is generalization via bottom-up structuring strategy. This could be done by associating similar concepts to create meta-concepts from leaves of the aforementioned hierarchy.

Step 5. **Refinement**: The final step is devoted to updating the visual structure by excluding the excessiveness, synonymy, and contradictions. As mentioned before, the main goal of the final step is try to create a beautiful ontology. We believe what makes ontology beautiful is harmony.

Using these tips the students develop several huge practical ontologies to conduct systemically more structured research. This approach is based on developing of a set of ontologies:

- Problem-ontology definition (ontology N1 describing main concepts)
- Ontology of reviewed approaches (ontology N2 describing the history of the problem)
- Experiment framework design (ontology N3 presenting experimental conception)
- Data structure ontology (ontology N4 presenting input and output data)t
- Mathematical modelling and main results ontology design (ontology N5 describing results)

Not all five ontologies are obligatory, but even an attempt to create them is a first step to perform systemic scientific study.

Conclusion

Students and teachers both are knowledge workers. So students enter "the world of ontologies" with interest and begin to use it in their practical research work.

Our experience in training of knowledge analysts and teaching this course in the period of 1999-2005 confirm the unique role of knowledge structuring for developing ontologies quickly, efficiently and effectively. We follow David Jonassen's idea of "using concept maps as a mind tool" [14]. The use of visual paradigm enables students to process and understand greater volume of information.

The course is double-ontological as the development of educational knowledge structures in the form of ontologies provides training and learning support. Teaching ontologies scaffold and improve students' understanding of the courseware and later help to realize substantive and syntactic company knowledge. As such, they can play a part in the overall pattern of learning, facilitating for example analysis, comparison, generalization and transferability of understanding to analogous problems.

Ontological framework scaffolds the student's research activity. But ontological engineering is rather easy for «old» sciences with good structure. Researchers in new, multi-disciplinary and ill-structured disciplines as HCI, cognitive sconces, management, etc. will face a bunch of difficulties in design and development phases. Ontologies also are rather subjective.

Our paper presents one of the first attempts to show the visionary role of knowledge engineering in helping student research. Ontologies are good for better self-understanding of research and then for knowledge sharing.

We also have experienced to teach the modification of this course to the practitioners. It was in-service training of analysts for IT-departments of some companies. The training was a success, as knowledge engineering is a unique set of methods that help everywhere. It is a real 'silver bullet'.

Acknowledgements

This work is partly supported by the grant of Russian Foundation for Basic Research No.04-01-00466.

Bibliography

1. Adeli, H. (1994) Knowledge Engineering. McGraw-Hill, New-York
2. Boose, J.H. (1990) Knowledge Acquisition Tools, Methods and Mediating Representations. In Knowledge Acquisition for Knowledge-Based Systems (Motoda, H. et al., Eds), IOS Press, Ohinsha Ltd., Tokyo, pp.123-168.
3. Eisenstadt, M., Domingue, J., Rajan, T. & Motta, E. (1990) Visual Knowledge Engineering. In IEEE Transactions on Software Engineering, Vol.16, No.10, pp.1164-1177.
4. Fensel, D. (2001) Ontologies: A Silver Bullet foe Knowledge Management and Electronic Commerce. Springer.
5. Gavrilova, T., Voinov, A. (1998) Work in Progress: Visual Specification of Knowledge Bases // Lecture Notes in Artificial Intelligence 1416 "Tasks and Methods in Applied Artificial Intelligence", A.P.del Pobil, J.Mira, M.Ali (Eds), Springer, pp. 717-726.
6. Gavrilova, T.A., Voinov, A., Vasilyeva E. (1999) Visual Knowledge Engineering as a Cognitive Tool / Proc. of Int. Conf. on Artificial and Natural Networks IWANN'99, Spain, Benicassim. - pp.123-128.
7. Gavrilova, T. (2003) Teaching via Using Ontological Engineering // Proceedings of XI Int. Conf. "Powerful ICT for Teaching and Learning" PEG-2003, St.Petersburg, p. 23-26.
8. Gavrilova, T., Kurochkin M., Veremiev V. (2004) Teaching Strategies and Ontologies for E-learning // Int. Journal "Information Theories and Applications", vol.11, N1, pp.35-42.
9. Gómez-Pérez, A., Fernández-López, M., Corcho, O. (2004) Ontological Engineering with examples from the areas of Knowledge Management, e-Commerce and the Semantic Web, Springer.
10. Gruber, T. (1993) A translation approach to portable ontology specifications. Knowledge Acquisition, Vol. 5, pp. 199-220.
11. Guarino, N. & Giaretta, P. (1998) Ontologies and Knowledge Bases: Towards a Terminological Clarification. In Towards Very Large Knowledge Bases: Knowledge Building & Knowledge Sharing, IOS Press, pp.25- 32.
12. Guarino, N., Welty, C. (2000) A Formal Ontology of Properties. In R. Dieng and O. Corby (eds.), Knowledge Engineering and Knowledge Management: Methods, Models and Tools. 12th International Conference, EKAW2000. Springer Verlag, pp. 97-112.
13. Jasper, R. and Uschold, M (1999). A Framework for Understanding and Classifying Ontology Applications. In 12th Workshop on Knowledge Acquisition Modeling and Management KAW'99.
14. Jonassen, D.H. (1998) Designing constructivist learning environments. In Instructional design models and strategies (Reigeluth, C.M. (Ed), 2nd ed., Lawrence Eribaum, Mahwah, NJ.
15. Miller, G. (1956) The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Proccessing Information. The Psychological Review, vol. 63, pp. 81-97.
16. Mizogushi, R. and Bourdeau J. (2000), Using Ontological Engineering to Overcome Common AI-ED Problems. International Journal of Artificial Intelligence in Education, v. 11, pp.1-12.
17. Neches, et al, (1991) Enabling Technology for Knowledge Sharing. AI Magazin, Winter, pp.36- 56.
18. OilEd, Bechhofer, S. and Ng G. Accessed from http://oiled.man.ac.uk/ at December 7, 2004.
19. OntoEdit, AIFB, University Karlsruhe Accessed from http://www.ontoknowledge.org/tools/ at December 07, 2004
20. OntoBroker, Accessed from http://ontobroker.aifb.uni-karlsruhe.de/ at December 7, 2004.
21. Ontolingua, Stanford University. Accessed from http:// www.ksl.stanford.edu/ software/ontolingua/ at December 7, 2004.

22. Protégé, Stanford Medical Informatics. Accessed from http://protege.stanford.edu/ at December 07, 2004.

23. Sowa, J. F. (1984) Conceptual Structures: Information Processing in Mind and Machine. Addison-Wesley, Reading, Massachusetts.

24. Scott, A., Clayton, J.E. & Gibson E.L. (1994) A Practical Guide to Knowledge Acquisition, Addison-Wesley.

25. Swartout, B., Patil, R., Knight, K. & Russ, T. (1997) Toward Distributed Use of Large-Scale Ontologies. In Ontological Engineering, AAAI- 97 Spring Symposium Series, pp.138- 148.

26. The CIO's Guide to Semantics, © 2004 Semantic Arts, Inc. www.semantic-conference.com

27. Tu, S., Eriksson, H., Gennari, J., Shahar, Y. & Musen M. (1995) Ontology-Based Configuration of Problem-Solving Methods and Generation of Knowledge-Acquisition Tools. In "Artificial Intelligence in Medicine", N7, pp.257-289.

28. Uschold, M., Gruninger M (1996). "Ontologies: Principles Methods and Applications", Knowledge Engineering Review,vol1, N1.

29. Wielinga, B., Schreiber, G. & Breuker J. (1992) A Modelling Approach to Knowledge Engineering. In Knowledge Acquisition, 4 (1), Special Issue, pp.23-39.

30. Werthheimer, M. (1959) Productive Thinking, HarperCollins.

Authors' Information

Tatiana Gavrilova *– Professor Saint-Petersburg State Polytechnical University, Politechnicheskaya 29, 195251, St. Petersburg, Russia, gavr_csa@rambler.ru*

Seppo Puuronen – *Professor of University of Jyväskylä, P.O. Box 35, FIN-40014, Jyväskylänyliopisto, FINLAND, Finland, sepi@cs.jyu.fi*

CONNECTION OF NETWORK SENSORS TO DISTRIBUTED INFORMATION MEASUREMENT AND CONTROL SYSTEM FOR EDUCATION AND RESEARCH

Sergey Kiprushkin, Sergey Kurskov, Eugene Sukharev

Abstract: *The development of the distributed information measurement and control system for optical spectral research of particle beam and plasma objects and the execution of laboratory works on Physics and Engineering Department of Petrozavodsk State University are described. At the hardware level the system is represented by a complex of the automated workplaces joined into computer network. The key element of the system is the communication server, which supports the multi-user mode and distributes resources among clients, monitors the system and provides secure access. Other system components are formed by equipment servers (CAMAC and GPIB servers, a server for the access to microcontrollers MCS-196 and others) and the client programs that carry out data acquisition, accumulation and processing and management of the course of the experiment as well. In this work the designed by the authors network interface is discussed. The interface provides the connection of measuring and executive devices to the distributed information measurement and control system via Ethernet. This interface allows controlling of experimental parameters by use of digital devices, monitoring of experiment parameters by polling of analog and digital sensors. The device firmware is written in assembler language and includes libraries for Ethernet-, IP-, TCP- u UDP-packets forming.*

Keywords*: distributed information measurement and control system, network sensors, Ethernet Interface, client-server technology, distance education.*

Introduction

Up-to-date systems of experiment automation are recently built on modules of software-controlled devices or digital measurement hardware, connected to interface bus. In all cases, hardware is connected to computer with interface device.

Integration of distributed system with remote sensors is more efficient, when the network interface used. It can be built on network chip and microcontroller. By using Ethernet interface, it is possible to connect different digital and analog devices, and the connection with servers will be based on TCP/IP networks.

The goal of this work is to develop a network interface for connecting remote sensors and execution units to distributed information measurement and control system for physical experiments.

The Distributed Information Measurement and Control System

The distributed information measurement system (Figure 1) is based on client/server technology and works in the nets on the basis of TCP/IP protocol stack [Gavrilov et al, 2003] – [Kiprushkin et al, 2005].

The system provides the remote access to information and hardware resources of automation equipped working places. The access to physical equipment is provided by the equipment servers (CAMAC server, GPIB server, the server of access to the MCS-196 microcontrollers and others). The communication server integrates the whole distributed systems. Its functions are: communication with user, system monitoring, security, and proper distribution of resources in multiuser mode.

The experiment process is determined and conducted by client software running on a client computer. It is necessary to emphasize, that the managing experiment software are operated not on the remote computer (as when using Web technologies) [Barrie et al, 1996], [Зимин и др., 2002], but on the user one, connected to the system via global network.

The communication server, the equipment servers, and the client software are implemented as Java applications. The data exchange between them is based on TCP stream sockets provided by java.net package, which is included into Java API standard package. The methods of using the input-output ports for the access to the interface controllers are written in C programming language.

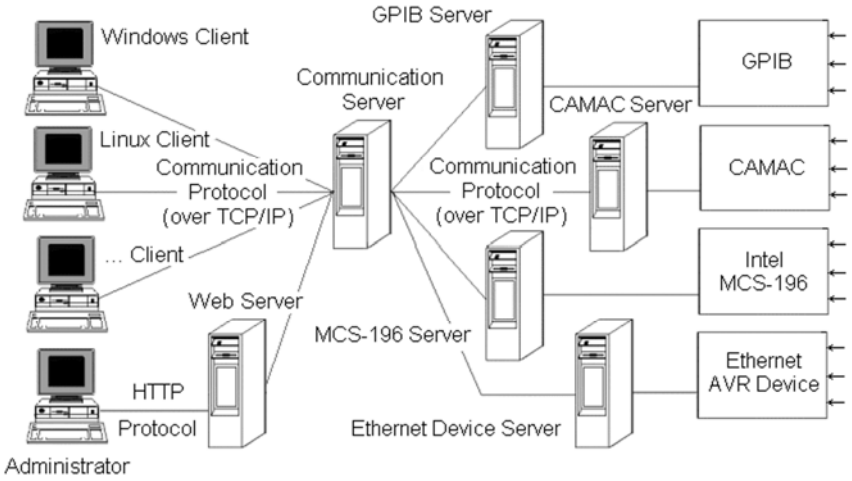


Figure 1. The scheme of the distributed information measurement and control system

Ethernet Interface

There are many specialized processors (network chips) designed for communication over networks. But it is necessary to create a central command unit, which will communicate with devices and control the network chip. This unit can be built on microcontroller.

Choosing network chip, it is necessary to take into account the physical environment and the required transmission rate of data. For communicating over 10 Mbps network, based on twisted pair, Realtek RTL8019AS processor was selected. This chip is compatible with ISA personal computer interface by timings, data and address signals.

By emulating ISA bus with microcontroller, it is possible to gain proper network chip functioning. Atmel AVR microcontrollers are good choice to implement this idea. They perform each instruction per one clock period, so their performance is 16 MIPS for 16 MHz clock rate. This performance is enough for ISA emulation.

Atmel ATMega8535 was used in the described device. It has 8-channel 10-bit ADC, 8 Kb Flash ROM, 512 b EEPROM, 512 b RAM, pulse-width modulator and analog comparator.

The main logic of the device functioning is described below. When the device is turned on, microcontroller firmware program is initiated. By sending the RESETDRV signal, the network chip is resetting. Then microcontroller configures the network chip. Configuration can be made in accordance to the desired aim of operation: e.g. reading data from measurement device and sending these data to specified network address.

The operation modes are:

- Control of the experiment execution through digital or analog devices (relays, step motors, gas injectors, etc.)
- Control of some parameters by polling analog and digital sensors (pressure, temperature, and optical sensors, atom beam sensor, etc.).
- Notify of parameter value, registering by measurement device.

Software

It is possible to present information flow as follows.

Analog value is converted into binary code by ADC. This very value must be received "on the other end of wire" for placing into database. The result is put into TCP packet. TCP protocol provides the reliable transmission of the messages between remote application processes. Then the IP datagram is formed from TCP packet (the level of the internetworking) and is sent to the bottom level – a network interface level.

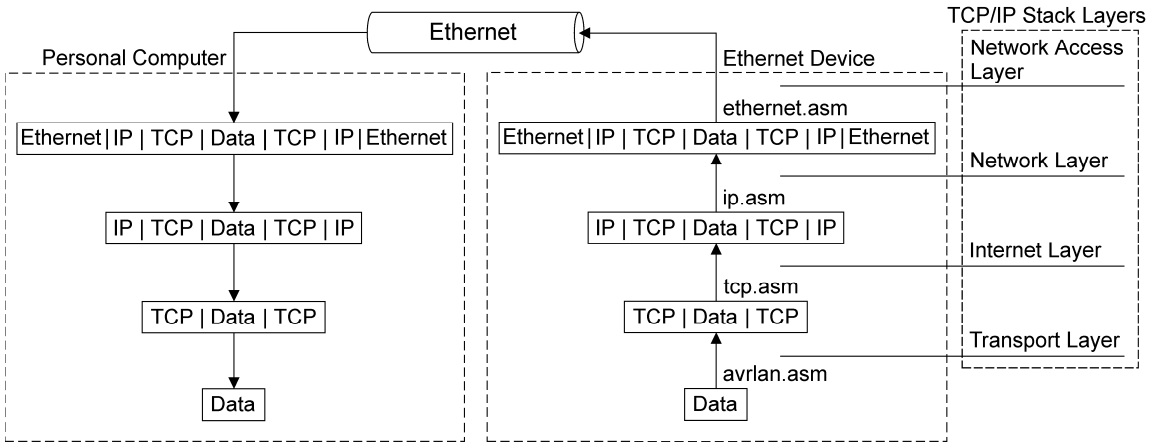


Figure 2. Set of the program modules and sequence of the transmission of the frame, received from remote device by PC

Protocols of this level must provide the integration into global network: TCP/IP network must have a facility of the integration into any other networks, which doesn't depend on internal technology of data communication in these networks. Hence, this level is impossible to define once and for all. An interface facility must be designed for every communication technology. IP-frames to Ethernet encapsulation protocol pertains to such interface facility. Encapsulation of IP-package into Ethernet-frames is described in RFC1042. Then Ethernet-frame is sent via communication media.

The other side receives the frame and performs the re-conversion by correspondent server software. Processing of the frames encapsulation doesn't take much processor time in personal computer. But microcontroller has smaller speed and less memory. That's why it is very important to solve this task by means of optimized algorithm and assembler language.

Set of the program modules and the sequence of the transmission of the frame, received from remote device by personal computer, are shown on Figure 2.

It is possible to use complete 3rd-party libraries and functions for TCP/IP implementation, because the software requires standard interactions only. These libraries are presented in all up-to-date programming environments (Java, .NET, Visual C++, Delphi, LabVIEW, etc.).

There is a question that needs to be answered by a developer of software: what level of TCP/IP stack must be implemented. This choice depends on software, used in personal computer, dataflow and processing speed of microcontroller, as well as requirements of reliability of information delivery.

The more preferred way is to use the lowest possible level of TCP/IP stack, sending data in Ethernet-, or IP-frames.

E.g. measured values of remote temperature sensor can be packed into Ethernet frames directly if qualification of the developer is sufficient to use the Ethernet level. But if the LabVIEW used, then you need to use all modules for package framing (from ethernet.asm to tcp.asm) on microcontroller side. Using the Java language for writing client applications also superimposes the restriction: when TCP-socket is used, you need to use the tcp.asm library in microcontroller. If you use UDP (unreliable delivery protocol), you must encapsulate messages with udp.asm library. This library works at transport level of TCP/IP stack that obliges to use as well as all underlayed protocols. Firmware, designed for the described device, is written on assembler language and includes Ethernet-, IP-, TCP- and UDP-package libraries.

The described device can be used in other networks, based on other protocols, but in this case it is necessary to develop the libraries for generation of correspondent frames.

Internet does not give any warranty for time of package delivery. This reason limits the use of the device in system that imposes hard time restrictions of information delivery. This feature can be eliminated by using a special network, used for undertaking the physical experiment only.

Conclusion

Ethernet interface device and corresponding software were developed and created. It implements access to remote sensor and digital device of the laboratory complex, used for scientific experiments in the field of optical spectroscopy and distant education on Physics and Engineering Department of Petrozavodsk State University.

This interface helps to increase the variety of devices, which can be connected to distributed information measurement and control system without using the computer and software-operated module electronics, as well as different instrument interface like GPIB.

Acknowledgments

We would like to express our gratitude to the laboratories' Head I. P. Shibaev for support of this work as well as students A. V. Mandychev and E. A. Vasilieva.

Bibliography

[Gavrilov et al, 2003] S.E. Gavrilov, S.A. Kiprushkin, S.Yu. Kurskov. Distributed information system with remote access to physical equipment. In: Proceedings of the International Conference on Computer, Communication and Control Technologies: CCCT '03 and The 9th International Conference on Information Systems Analysis and Synthesis: ISAS'03. Orlando, 2003.

[Kiprushkin et al, 2004] S.A. Kiprushkin, N.A. Korolev, S.Yu. Kurskov. Data security in the distributed information measurement system. In: Proceedings of the 8th World Multi-Conference on Systemics, Cybernetics and Informatics: SCI 2004. Orlando, 2004, Vol. 1, pp. 13-16.

[Kiprushkin et al, 2005] S.A. Kiprushkin, S.Yu. Kurskov, N.G. Nosovich Administration of distributed information measurement system. In: Proceedings of the 9th World Multi-Conference on Systemics, Cybernetics and Informatics: WMSCI 2005. Orlando, 2005.

[Kiprushkin et al, 2005] S.A. Kiprushkin, N.A. Korolev, S.Yu. Kurskov. Sharing of Instrument Resources on the Basis of Distributed Information Measurement System. In: Proceedings of the Second IASTED International Multi-Conference on Automation, Control, and Information Technology - Automation, Control, and Applications: ACIT-ACA 2005. Novosibirsk, ACTA Press, 2005, pp. 170-175.

[Kiprushkin et al, 2005] S. Kiprushkin, N. Korolev, S. Kurskov, N. Nosovich. Distributed Information Measurement System for Support of Research and Education in Optical Spectroscopy. In: Proceedings of the Third International Conference "Information Research, Applications and Education": i.TECH 2005. Sofia: FOI-COMMERCE, 2005, pp. 171-179.

[Barrie et al, 1996] J.M. Barrie, D.E. Presti. The World Wide Web as Instructional Tool. Science, 1996, V. 274.

[Зимин и др., 2002] А.М. Зимин, В.А. Аверченко, С.Ю. Лабзов и др. Лабораторный практикум по спектральной диагностике плазмы с удаленным доступом через Интернет. Информационные технологии. 2002, N 3, pp. 37–42.

Authors' Information

Sergey Kiprushkin *– Petrozavodsk State University, Lenin St., 33, Petrozavodsk – 185910, Russia; e-mail: skipr@dfe3300.karelia.ru*

Sergey Kurskov *– Petrozavodsk State University, Lenin St., 33, Petrozavodsk – 185910, Russia; e-mail: kurskov@psu.karelia.ru*

Eugene Sukharev *– Petrozavodsk State University, Lenin St., 33, Petrozavodsk – 185910, Russia; e-mail: eugene-mobile@yandex.ru*

ИНФОРМАЦИОННО-СПРАВОЧНЫЕ СИСТЕМЫ И ДИСТАНЦИОННОЕ ОБУЧЕНИЕ

Андрей Донченко

Abstract*: This article considers the Internet/Intranet information systems as the tool for distance learning. Author considers the model of the 3-tier WEB based information system, the idea of the language for implementing and customized solution, which includes the original language and processor for fast prototyping and implementing small and middle sized Internet/Intranet information systems.*

Keywords*: информационно-справочная система, структура клиент – сервер, дистанционное обучение.*

Введение

В настоящее время все большую актуальность приобретают задачи дистанционного обучения. Важным инструментальным средством их реализации являются малые и средние информационно-справочные системы в рамках Internet/Intranet узлов.

Данный доклад посвящен проблемам дизайна, проектирования и классификации таких систем. Предложена модель архитектуры системы.

Описан оригинальный инструмент проектирования и реализации.

Обзор проблематики и целей

В настоящее время большое распространение получают задачи, связанные с проектированием, обеспечением и поддержкой информационно-справочных систем в рамках Internet/Intranet технологий. Это задачи, связанные с обеспечением и поддержкой доступа к базам данных разного рода посредством стандартных протоколов транспортного уровня (поисковые сервера, справочные системы и т.д.). Существует класс систем, таких, как, например, Oracle - WWW, решающих задачу интеграции информационно-справочных систем с базами данных и прототипирования документов. К сожалению, большинство из них либо ориентировано на конкретную операционную систему (например – Microsoft Peer WEB server), либо требуют специфичного программного обеспечения на серверной и клиентной части.

Важными аспектами проектирования такого рода систем являются следующие.

1. Выбор способа представления и подачи информации (информационная архитектура системы).
2. Пользовательский интерфейс системы.
3. Инструментальные средства, позволяющие прототипировать и поддерживать пользовательский интерфейс, саму систему в рамках выбранной модели представления информации.

Ниже рассмотрена концепция построения и описано инструментальное средство для прототипирования и поддержки класса информационно-справочных систем в рамках Internet/Intranet узла с учетом указанных аспектов.

Информационная архитектура

Информационная архитектура включает в себя два основных аспекта: функциональное взаимодействие с пользователем и способ структуризации предоставляемой информации.

С функциональной точки зрения взаимодействие пользователя и информационно-справочной системы сводится к поиску информации, формированию запросов с последующей навигацией по их результатам.

В общем случае, существуют два типа поисков: полнотекстовый и категориальный. Последовательный просмотр (ознакомление, обучение) является вырожденным случаем поиска, где по умолчанию происходит переход к следующему документу в списке.

Согласно одному из подходов [Wurman Richard Soul, 1996], можно выделить пять способов структуризации информации (LATCH):

1. по географическому положению (L – Location);
2. лексикографический: по алфавиту (A – Alphabet);
3. по времени (T – Time);
4. по категории (C- Category);
5. иерархически (H – Hierarchy): от большего к меньшему(по изменению плотности, цвета и т.п.).

Первым шагом проектирования информационной архитектуры системы является решение о выборе основного и альтернативных способов структуризации предоставляемой информации. Эти способы учитывают особенности принципов работы человека в процессе творческой деятельности – итеративный цикл генерации гипотезы, ее верификация, разбиение проблемы на составляющие ее задачи, наличие ряда методов для каждой задачи, спонтанное переключение от одного метода решения к другому. Эволюция поисковых систем в Internet иллюстрирует вышеприведенные положения. Большинство из них начинались либо как систематизированные списки ресурсов (например, Yahoo, списки полезных ссылок на страничках пользователей), либо как полнотекстовые поисковые системы (Lycos, AltaVista). В данный момент каждый поставщик таких услуг предлагает оба вида поиска как взаимодополняющие друг друга.

Пользовательский интерфейс информационных систем

В настоящее время определились только состав и структура базовых элементов взаимодействия с пользователем на атомарном уровне (так называемые элементы управления, текстовое поле, переключатель, список, просматриватель иерархий). Средства проектирования, построения диаграмм и адекватного представления информационных моделей более высокого уровня (энциклопедия, учебник, справочная система) еще только начинают структурироваться и стандартизироваться. Эти средства базируются с одной стороны, на наработках многочисленных бумажных изданий, давших такие метафоры как содержание, индексы, таблицы иллюстраций, глоссарии и т.д., с другой – на новых метафорах, порождаемых компьютерными технологиями. Яркое тому подтверждение - эволюция гипертекстовых систем. Брешь между этими метафорами и базовыми элементами интерфейса заполняет программист, конструируя оболочки, позволяющие специалистам в предметных областях организовывать информацию.

Помимо упомянутых способов представления информации, связанных с выбором информационной архитектуры и пользовательского интерфейса, отметим и проблемы сопровождения информационных продуктов, поддержки в актуальном состоянии, а также взаимодействия между создателями системы и ее пользователями (например, сообщения об ошибках, сообщения о новых изменениях и дополнениях в системе).

Отсутствие стандартов, поддерживающих информационные метафоры, тормозит взаимодействие различных информационных систем и слияние отдельных информационных массивов. В процессе решения конкретных задач реальной жизни, появляются, шлифуются, стандартизируются отдельные элементы (мета - теги в HTML страницах, форматы гипертекстовых справочных систем – WinHelp или InfoBase). Однако в целом поле деятельности в этой области все еще открыто.

Модель информационной системы

На основе анализа уже существующих информационно-справочных систем можно сформулировать следующие положения, определяющие концептуальную схему модели информационной системы.

- Информация должна быть представлена линейным, однородным набором *информационных статей* (документов) с уникальным идентификатором.
- Каждая статья документа должна принадлежать одному или нескольким *индексированным последовательностям*, построенных на механизмах связных ссылок и представляющих иерархии. Примерами индексированных последовательностей являются содержание, глоссарий, библиография, атлас, и т.д.
- Идентификаторы и пути в иерархиях должны представлять собой методы адресации данной статьи.
- Интерфейс пользователя должен определяться в терминах фреймов – экран разделяется на связанные между собой фреймы (например, сверху, список индексированных последовательностей, ниже слева – содержание, справа – содержимое статьи, или слева – индекс, а справа – список статей по данному индексному гнезду).
- Переход должен осуществляться в зависимости от метода адресации статьи: либо на статью для идентификатора, либо на фреймовое состояние для пути в иерархии.
- Полнотекстовый поиск должен предоставлять средства для формирования запроса (регулярные выражения) и ограничения области поиска (определение индексированных последовательностей, по которым осуществляется поиск) и просмотра/навигации по списку найденных статей.
- Встроенные средства обратной связи (например, кнопка для вызова диалога определения описки, ошибки, предложения по системе и т.п. с дальнейшей отсылкой сообщения для последующей обработки) между пользователем и создателем должны обеспечивать возможность фиксации предложение без перерыва в работе с системой.
- Встроенный механизм сервера рассылки должен осуществлять функцию off-line-нового оповещения зарегистрированных пользователей об изменениях в состоянии системы.

Инструментальные средства прототипирования и поддержки информационно-справочных систем

С учетом сформулированной выше концептуальной модели построения информационно-справочной системы было разработано инструментальное средство прототипирования и поддержки информационно-справочных систем в рамках Internet/Intranet узла. Реализовано как программная система, представляющая собой сервер - базированное решение задачи прототипирования и поддержки гипертекстовых документов, использующих данные в табличном представлении в рамках Internet/Intranet-технологий. На транспортном уровне используется HTTP-протокол. Структура серверной части использует CGI-соглашения для HTTP-сервера. На клиентную часть не налагается никаких ограничений, в качестве клиентного программного обеспечения может быть использован любой WWW-клиент (Microsoft Internet Explorer, Netscape Navigator, Lynx и т.д.), поддерживающий стандарт HTML-2.0 представления гипертекстовых документов.

С функциональной точки зрения взаимодействие компонент инструментальной системы можно представить с помощью рисунка 1.

С точки зрения архитектуры клиент/сервер эта система является трехуровневой, где интерфейс с базой данных – первый уровень, второй -уровень бизнес-правил, третий- уровень интерфейса с клиентным приложением.

На каждом шаге генератор представлений по запросу пользователя на основе описания шаблона документа и табличных данных формирует результирующий документ. Этот документ с помощью стандартного WWW - сервера и HTTP-протокола доставляется к пользователю. Далее, на клиентной части, возможен просмотр и печать результирующего документа с помощью средств, предоставляемых стандартным WWW - клиентом.

Наборы данных системы.

Все основные наборы данных системы представлены как текстовые файлы.

Входными наборами данных процессора шаблонов документов являются собственно шаблон документов и табличные данные, используемые при построении результирующего документа. Программа на языке описания шаблонов документов поставляет процессору шаблонов необходимую информацию для построения результирующего документа. Табличные данные могут быть так же инвертированы и индексированы с целью оптимизации доступа и обеспечения фильтрации кортежей по ключу.

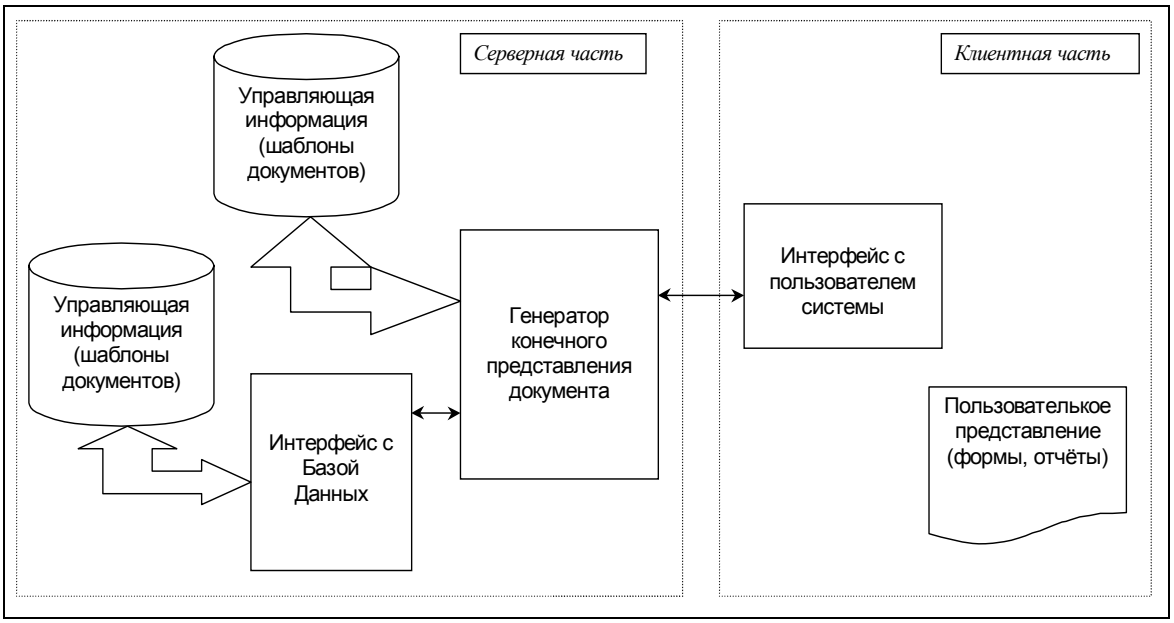


Рис.1. Взаимодействие компонент инструментальной системы.

Результатом выполнения является текстовый документ, отвечающий предъявленным требованиям [T.Berners-Lee, D. Connoly, 1995] и построенный на основе шаблона описания и табличных данных.

Этот документ может быть просмотрен и обработан с помощью любого WWW-клиента, поддерживающего HTML 2.0 в качестве входного языка описания документов.

Рабочие языки системы

Для обеспечения эффективного функционирования система поддерживает два языка. Это собственно язык описания шаблонов документов и язык представления табличных данных.

Язык описания шаблонов документов

Целью языка описания шаблонов документов является возможность параметризации представления результирующего документа на основе таблиц, хранящихся в базе данных на сервере.

Текущее состояние исполнительной системы определяется тройкой значений: текущий оператор языка представления шаблонов документов, текущая таблица данных, номер записи в текущей таблице данных.

Макрокомандой будем называть оператор языка представления шаблонов документов, вызывающий изменение параметров исполнительной системы (открытие новой таблицы, генерация нового фрагмента текста).

Макропеременными будем называть объекты входной программы, связанные с непосредственно с входными данными текущей таблицы (с значениями доменов, номером текущей записи).

Макрофункцией будем называть оператор языка, не приводящий к изменению состояния исполнительной системы и генерирующий значения, непосредственно не связанные с входными данными текущей таблицы.

Макроподстановкой будем называть использование макропеременной, макрокоманды или макрофункции в тексте программы на языке представления шаблонов документов.

Язык описания шаблонов реализован как язык макроподстановок. Программа на нем является шаблоном документа. Сам документ представлен в виде суперпозиции константных, не изменяемых и не зависящих от входных данных, цепочек символов и макроподстановок, вариантных частей, вид которых может изменяться в зависимости от содержимого табличных файлов и входных параметров исполнительной системы.

С функциональной точки зрения все операторы, макропеременные и функции языка разбиваются на три группы:

- управляющие процессом исполнения входной программы; они включают в себя операторы ветвления и цикла;
- обеспечивающие взаимодействие с базой данных; позволяет задать текущую таблицу (возможно – инвертированную), задать текущую запись, получить значения доменов в текущем кортеже и сведения о текущей таблице.

Служебные операторы и макрофункции. Обеспечивают взаимодействие с операционной средой (доступ к параметрам CGI - интерфейса, к переменным среды операционной системы) и функции преобразования данных.

Язык представления табличных данных.

Рассматриваемый язык обеспечивает табличное представление исходных данных во внутреннем, специфичном для данной системы формате. Каждая таблица представлена как отдельный файл, в котором представлены кортежи данной таблицы. Атрибутом является порядковый номер домена в кортеже.

Заключение

В данной работе

- Предложена концептуальная модель представления информации, охватывающая широкий класс энциклопедия - подобных справочных систем, предоставляющая стандартизованный набор механизмов поиска и средств навигации по содержимому. Встроенные в систему механизмы обратной связи с пользователем и оповещения пользователей облегчают поддержку различных этапов жизненного цикла информационной системы. Рассмотрен набор метафор – элементов определения информационных структур - элементов пользовательского интерфейса высокого уровня - позволяющих определять, описывать и реализовывать информационные системы.
- Рассмотрены основные черты разработанного и реализованного комплекса программных средств, предоставляющего следующие возможности в рамках сформулированной модели информационной системы:
  1. Гибкий и простой механизм поддержки взаимодействия между HTTP-сервером и данными в табличном представлении.
  2. Межплатформенную переносимость. Форматы хранения шаблонов текстовых документов и представления табличных данных не зависят от выбора операционной системы. Языковый процессор шаблонов документов так же является переносимым.
  3. Расширяемость множества результирующих документов за счет изменения как непосредственно табличных данных, так и множества шаблонов документов.

На основе рассмотренных модели информационно-справочной системы и инструментального средства был оформлен ряд промежуточных проектов в виде мини-WWW - сайтов, опубликованных в Internet.

Литература

[T.Berners-Lee, D. Connoly, 1995] T.Berners-Lee MIT/W3C, D. Connoly. RFC 1866, Hypertext Markup Language – 2.0, November 1995, Network Working Group.

[Wurman Richard Soul, 1996] Wurman Richard Soul. Information Architects. 1996, Graphis Press Corp. ISBN 3-85709-458-3

[Леонтьев А. Н., 1981] Леонтьев А. Н. Проблемы развития психики. 3-4 изд. М., Изд-во МГУ, 1981.

Информация об авторе

Андрей Донченко - *Bonus Technology Inc., Ph.D., Project Manager/Team Leader; 86D Bozhenko Str., Kyiv, Ukraine, e-mail: andriy.donchenko@gmail.com*

FOCUSING ON DECISION MAKING IN NEW ESP CURRICULUM FOR UKRAINIAN STUDENTS

Elena Baranova, Elena Blyznyukova

Abstract: *The article deals with the problem of training students to become professionals with job-oriented competences which include general competences, capacity for autonomous learning, and ability to make decisions in the field areas. The main focus of the article is Ukrainian new ESP curriculum that takes decision making as one of the most necessary pragmatic competences and study skills.*

Keywords: *English for specific purposes (ESP) curriculum, a decision-making model, problem-based learning*

Introduction

One of the defining features of the modern field of second language teaching for specialists is its cross-disciplinary nature. Growing out of linguistics and psychology, it has been stimulated by theoretical concepts developed in fields such as mother tongue education (process approaches to teaching), discourse analysis (genre theory), artificial intelligence (schema theory) and subject learning (job-related language learning). A rich area of research, one that is particularly relevant to issues of language learning and teaching specialists is that of cognitive science. This area has been linked to the study of language learner strategies [Rubin,1981]. Another aspect of this field with potentially great relevance is the modeling of "expert systems" and processes of decision-making.

Problem Identification

Newly developed Ukrainian ESP (English for specific purposes) curriculum [ESP National Curriculum, 2005] is designed to explore this relevance by combining these areas: looking at the topic of learner strategies through the lens of a decision-making model.

In this article we are making an attempt to show how updated ESP curriculum is going to meet the requirements to specialists training.

Theoretical Analysis of the Problem and Personal Observations

First, we are summarizing a complex of psychological activities that are necessary for decision making. Then we are describing learner strategies as they have evolved over the past two decades. And finally we are framing the learner strategy concept in terms of a decision-making model, relating it to work on teacher decision-making,

pointing out some similarities between teacher and learner decision-making, and highlighting the negotiated nature of decision-making in classroom language learning.

The new curriculum has been developed with the approval of the Ministry of Education and Science of Ukraine:

- as a response to international developments;
- in order to meet the language needs of university students across a range of discipline areas;
- to provide benchmarks for teachers and students in line with the identified levels;
- to provide a standardized basis for course and syllabus design by teachers of English at the faculty level in universities throughout Ukraine.

An enlarged Europe has led to radical changes in education. The creation of the European Higher Education Area by 2010 [Bologna, 1999] sets challenging tasks in terms of greater mobility for students, more effective international communication, better access to information and deeper mutual understanding.

The ESP curriculum was designed to give students an opportunity to develop the competences and strategies needed to function effectively in the study process and in the professional situations they encounter. As a result of new courses higher levels of student language proficiency and their decision making skills will facilitate individual mobility and competitiveness in the job market. So, it is viewed as important to discuss why decision making is important in training career-oriented students.

After exploring the pitfalls of decision making in learning processes students may feel that good decisions are impossible before they become experienced specialists. Yet major decisions must be made – about jobs selection, family/career choice, where to live, what to do with money or how to make money in the first place.

However, the trainers' task is to help students discover whether a decision is good or bad only after they make it and begin to see its consequences. In many cases a person never knows whether his choice is the right one until he takes over the responsibility and uses hand-on experience. If you decide that it is too early to take a risk, you can only guess what career opportunities would have been like if you had accepted the challenge.

A teacher can never guarantee good decisions, but he can guarantee a useful procedure that can increase students' chances of doing better. This procedure is described in the newly developed curriculum that gives recommendations on how to teach English for specific purposes (ESP). Not only does the curriculum give levels and language skills descriptions, but also it characterizes study skills as students' pragmatic competence. Decision making turns out to be one of the most important study skills for future specialists.

Decision making process a complex of psychological activities:

- Selecting the most important changes to make
- Evaluating the relative importance of different options
- Selecting between good options
- Choosing between options by projecting likely outcomes
- Weighing the pros and cons of a decision
- Analyzing the pressures for and against change
- Looking at a decision from all points of view
- Seeing whether a change is worth making

In the updated Ukrainian ESP curriculum it is expected that mentioned above psychological activities should be performed within problem-based learning (PBL).

For the first time the focus on decision making skills has been made in the frame of communicative approach towards language learning to train field specialists.

The template for skills development is based on problem-based learning which is taken as an instructional method characterized by the use of "real world" problems as a context for students to learn critical thinking and problem-solving skills and acquire knowledge of the essential concepts of the job-related course.

By using case-studies within PBL students obtain life-long learning skills which include the ability to find and use appropriate learning resources and develop appropriate strategies. Such approach has proved to be the most effective and constructive for the fast changing social economic situation in Ukraine.

New ESP curriculum that is being currently implemented for teaching English to “field students”, is designed to develop and sustain a set of strategies that enable students to make language decisions in professionally-oriented situations.

However, the complexity of strategy use is becoming more apparent. It has become clear that there are different strategies characteristic not only of different learners, but also of the same learner at different levels, with different language learning goals, engaged in the use of different skills, and so on. As a result of this growing realization, research began on the factors affecting the choice of strategies, rather than just the strategies themselves. This research is summarized by Oxford [Oxford, 1989]. Strategy choice was found to be related to the language being learned, the learning goals, the level of learning (or proficiency of the learner), the learner's self-awareness, age, and sex. Affective variables have been found to play a role: the attitudes, motivational level, and motivational orientation. Personality characteristics play a role as well-learning styles, learning experiences, and methods-as doe’s cultural background (national origin or ethnicity).

Another important factor was investigated by Wenden [Wenden, 1986] with the notion of learner beliefs. The connection was made between a learner's beliefs about language learning and the types of strategies that he or she uses. In addition to these longer term and stable characteristics, there are also a number of short-term factors which play a role in strategy choice: the requirements in the current communication or task in the specific situation of language use (and this of course includes the method by which the researcher attempts to elicit a learner's strategies).

This growing appreciation for the complexity of the issue of learning strategies and the factors that play a role in their use makes it increasingly apparent that a classification and taxonomy of strategies does not adequately represent the intricate ways in which strategies are chosen and used. One of the consequences, or costs, of developing such a classification is that the learner strategies are removed from the context in which they occur. They are categorized and given labels which make sense to the researcher, i.e. explicitly relating strategies which are seen as falling into the same paradigmatic class or category. However such a categorization is not necessarily one which makes sense to the learner, nor one that is used by the learner for accessing possibilities or weighing and making choices about what to do. Such a classification does not give us a sense of why a learner uses a particular type of strategy at a particular time, how this strategies fits with other strategies that are being used, and how the strategy is related to the learner's evolving planning process and beliefs, and how it is evaluated and feeds into further strategy choices.

To determine how and when the learner uses certain strategies, we need to look for relationships among them which are relevant to the learner: those strategies which are considered by the learner to be sub-strategies of others (i.e., the means to achieving others), and those which are related, from the learner's perspective, by patterns of sequencing (i.e., those which naturally stimulate or follow others).

Examining learner actions and behavior in terms of a decision-making model provides an alternative for researching and reporting on learning processes. In the work on teacher decision-making [Woods, 1996], a number of concepts are elaborated that are related to decision-making in the learning-teaching process from the perspective of the teacher. There is every reason to suspect that the other side of the coin-the learner's perspective on decision-making in the process-would provide interesting insights. In fact, there seem to be a number of parallels between learner decision-making and teacher decision-making that are worth exploring. The first consideration is that there is a great deal of similarity in the types of decisions made by teachers and those made by learners: many decisions could be made equally well by either party. For example, it may be a teacher who decides who to call on to answer a particular question, or a learner who decides to put up her or his hand to answer the question. It may be a teacher who assigns a particular exercise for homework of a learner who decides to do it for extra practice. Although there is an important difference-the teacher's decisions are ultimately geared towards actions by another person while the learner's decisions are ultimately geared towards her or his

own actions (i.e., it is the learner who ultimately has to act for learning to take place)-the decisions can involve many of the same actions. In fact, there is often a negotiation that takes place with regard to whose responsibility it is to make which decisions.

Taking into account the teacher-students relations variety new ESP curriculum develops new teacher roles: a facilitator, a process monitor, a consultant, and a language assistant rather than a ready made and rigid example to follow or a totalitarian knowledge supplier.

The focus of this article, though, is not the teacher decision making experience, but learner decision making and the event cognition.

Outcomes of Theoretical Analysis

This issue can be explored in a decision-making model of language learning, where learner strategies are seen as being part of the learner's decision-making process. This perspective implies a more naturalistic accounting of the learner's thinking during the process of learning (or attempting to learn) another language.

According to the mentioned in the ESP curriculum thinking activities we can help students develop the following decision stage model which comprises several steps:

- to notice the situation;
- to interpret it as one in which help is needed (to collect data and analyze it);
- to assume personal responsibility;
- to choose a form of assistance;
- to carry out that assistance.

Within this model different students act in a different way: onlookers who do not help may reach only stage 3 deciding on personal responsibility and decide at this point that it is not their responsibility.

The advantage of this five stage decision model is that it explains why students may fail to help even though they recognize the situation as an emergency.

It is the leaders that usually take over and go through the whole model. However, the leader’s role in a model may vary from high (deciding and telling the group a decision) to low (describing the problem and then joining the group in making the decision).

The event structure framework described above allows us a mechanism for examining these relationships, in particular the means-goals relationship as well as the interconnection between beliefs and choices of actions on the part of the learner.

In a study of learners perceptions about their classes [Allwright and Woods, 1992], we noted several characteristics about learner decision-making that were both similar to and different from teacher decision-making. The first is that, within the interview format that we used to collect data, we found that learners often had goals for their learning and were making decisions about how to achieve these goals; however, they were often much less active decision-makers than the teachers. This is of course not surprising given the usual social-educational norms of the teachers being responsible for the majority of the classroom-related decisions (it is the teacher who holds or announces the plan for the lesson). With the responsibility and the expectation about that responsibility on the shoulders of the teacher, it is natural that the teacher will put more time and effort into thinking through procedures and alternatives than will the learner. As a result, the pattern of decisions produced by the learner reveals that it is often not as complex and hierarchically organized as the teacher's. There are fewer levels of action and sub-actions distinguished and made explicit by the learner than by the teacher.

A second aspect is that students nonetheless have expectations about what will happen in the classroom, and how different things will be done, and who will do what. As with teachers, these expectations depend on previous experiences in language classrooms and on beliefs about language learning. These expectations can be considered a kind of 'implicit plan' that the learner holds for which no advance actions need to be explicitly decided upon until the expectations are not fulfilled, resulting in what Linde [Woods, 1996], terms a 'hotspot.'

At the point when the learner becomes aware of the discrepancy between her or his expectations and that is happening, she or he is more likely to begin to carry out some explicit compensatory planning (even if it is verbalizing a complaint to a classmate).

A number of case studies recommended as classroom techniques in problem-based learning allow examining some aspects of this process, incorporating the intricate interplay of beliefs and different types of motivation in the process as well as strategies combination. The ESP experience shows clearly that a one-shot elicitation of strategies can be very misleading, but a list of the strategies used by a learner can explain what is going on and lead to the most productive decision making.

One of the most important issues that make the foundation for the new ESP curriculum concept is the question of who is supposed to be doing what in the language teaching/learning decision-making process. The question does not have an unambiguous answer and therefore the compilers of the ESP Program came up with the term 'management of language learning.' In this conception, the traditional view that the teacher 'teaches' (tells the learners what to do to learn) and the learners do the 'learning' is seen as a oversimplification of the process. In fact, the actual learning cannot be seen. All we can observe is the actions designed to lead to learning, which can be decided upon by the learner or by the teacher. Whose roles it is to do what is potentially up for negotiation in every class. For example, the exhortation to teach language learning strategies carries with it the possibility that learners will have to learn (i.e., be convinced) that it is their role to make decisions about which actions they should take to learn, a job traditionally done by the teacher and one that they might well resist.

Conclusion

The notion that decision-making processes are negotiated and shared brings us to a new conception of strategy research, one that focuses not on learner strategies but rather on learning strategies and the intricate interplay of learner and teacher in their determination.

ESP recommendations also analyze concepts “individual versus group decision making” and “effective and ineffective decision making”.

Bibliography

[ESP National Curriculum,2005] English for Specific Purposes (ESP) National Curriculum for Universities. Kyiv: British Council, 2005.

[Oxford,1989] Oxford R. Use of language learning strategies: A synthesis of studies with implications for strategy training System, 1989, №17.

[Oxford,1989] Oxford R. Language learning strategies: what every teacher should know. New York: Newbury House, 1990.

[Rubin,1981] Rubin J. Study of cognitive processes in second language learning. Applied Linguistics, 1981, № 11, P.188-131.

[Wenden,1986] Wenden A. Helping language learners think about learning. ELT Journal, 1986, № 40, P. 3-12.

[Woods, 1996] Woods D. Studying ESL teachers' decision-making: Rationale, methodological issues and initial results. Carleton Papers in Applied Language Studies, 1989, № 6, P.107-123.

Authors’ Information

Elena A. Baranova *– Kirovograd Social Pedagogical Institute "Pedagogical academy", Ukraine, e-mail: helenbaranova@yahoo.com*

Elena N. Blyznyukova *– Kirovograd State Teachers’ Training University, Ukraine, e-mail: bllyus@yahoo.com*

ПАМЯТИ

ИННА СОВТУС

04.09.1955 – 09.10.2005

9 октября 2005 года ушла из жизни Инна Кузьминична Совтус, талантливый ученый и педагог, доктор технических наук, профессор, действительный член (академик) Аэрокосмической академии Украины.

В 1976 году Инна Кузьминична закончила Киевский государственный педагогический институт на физмат факультет получив диплом с отличием. И поэтому она после окончания института 10 лет работала в школе учителем математики. В 1985 году поступила на кафедру высшей математики военного института (КВИРТУ), где были замечены ее потенциальные способности к исследовательской работе.

Особо яркий вклад в науке были ее методы и модели нелинейного программировании. Вершина в этом направлении является „модифицированный метод неопределенных множителей" и распространение этого метода на различные сложные структуры соединения элементов в системе, с различным характером резервирования элементов. И. Совтус впервые предложила новый взгляд на структуру сложной системы (технической, экономической и др.) со стохастической связью ее элементов.

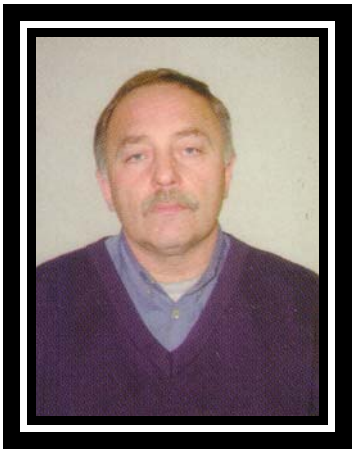
В последнее время ее исследовательские интересы были направлены на проблем оптимизации организационных структур, которые находятся на стыке технических и экономических наук. Полученные графо-аналитичиские результаты с прикладной интерпретацией она считала весьма перспективным в экономико-математических исследованиях, работала над совершенствованием их аналитической обозримости и прикладной направленности. Особая ее исследовательская любовь последних нескольких лет – это «вероятностная теория экономического риска», элементы которой планировались содержанием разделов второй докторской диссертации, которую она планировала защитить на экономические науки.

Публикации И. Совтус (книги, статьи, авторские свидетельства на изобретения и патенты в открытой и закрытой печати) составляют более 150 наименований. Некоторые из них были опубликованы на страницах сборников конференций KDS и Международного научного журнача «Information Theories and Applications».

Ассоциация Создателей и Пользователей
Систем Искусственного Интеллекта

Редакция международного журнала
«Information Theories and Applications»

IN MEMORIAM



ANDREY DANILOV

24.06.1949 – 07.12.2005

On the 7th December 2005 the eminent scientist in the field of computer sciences, research associate in the Department of Control and Informatics in Technical Systems in St. Petersburg State University of Aerospace Instrumentation, expert in the learning center of the Federal State Enterprise «Admiralty Shipyard», consultant in the Committee on Information, Press and Telecommunication of the Government of the Leningrad Region, member of many international projects for development and using of information technologies in the field of education and training of the staff, doctor of technical sciences, Andrey Dmitrievich Danilov pass away.

The scientific community has lost talented scientist, a grand person and friend.

Andrey Dmitrievich Danilov has published more than 60 works in the field of controlling of technical and information systems, knowledge technologies and distance learning.

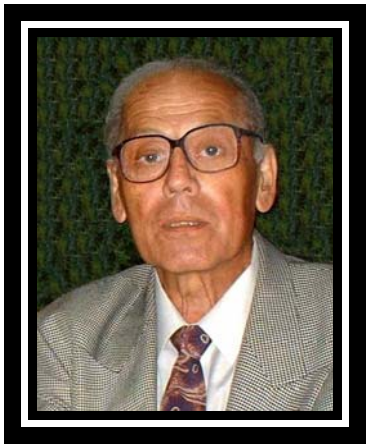
A lot of these publications have been published on the pages of the International Journal «Information Theories and Applications»; he was one of the most active participants of the conferences, which has been organised by the Journal.

The Members of the Institute on Information Theories and Applications FOI ITHEA and Editorial Board of the International Journal «Information Theories and Applications» feel deep regret of bereavement of our colleague and friend Andrey Dmitrievich Danilov.

Institute on Information Theories and Applications FOI ITHEA

International Journal «Information Theories and Applications»

ПАМЯТИ



ВАЛЕРИЯ НИКОЛАЕВИЧА КОВАЛЯ

23.02.1937 – 15.03.2006

15 марта 2006 года ушёл из жизни выдающийся учёный в области компьютерных наук, создатель первой в Украине супер-ЭВМ «СКИТ», заведующий отделом теории цифровых математических машин и систем Института кибернетики имени В.М. Глушкова Национальной Академии Наук Украины, доктор технических наук, профессор ВАЛЕРИЙ НИКОЛАЕВИЧ КОВАЛЬ.

Научный талант, трудолюбие и исключительная воля к достижению стратегических целей позволили Валерию Николаевичу получить широкую известность как специалиста в области системного анализа многомерных процессов, цифровых систем обработки информации, теоретических основ построения ЭВМ нового класса – интеллектуальных решающих машин. Выдающимся итогом его плодотворной жизни стала разработка супер-ЭВМ «СКИТ», которая входит в список наиболее высопроизводительных ЭВМ стран СНГ. В трудные годы становления нового независимого государства – Украины ВАЛЕРИЙ НИКОЛАЕВИЧ сумел найти возможности и ресурсы, сплотить коллектив для работы над столь грандиозным проектом.

ВАЛЕРИЙ НИКОЛАЕВИЧ КОВАЛЬ ушёл от нас в расцвете творческих сил, не реализовав всех своих замыслов. Но он оставил яркий след в отечественной науке и в наших сердцах...

Коллектив Института кибернетики имени В.М. Глушкова Национальной Академии Наук Украины, Ассоциация Создателей и Пользователей Систем Искусственного Интеллекта, редакция международного журнала «Information Theories and Applications» испытывают глубокое сожаление по поводу ухода из жизни ВАЛЕРИЯ НИКОЛАЕВИЧА КОВАЛЯ – нашего лидера и товарища – и выражают соболезнование семье покойного.

Институт кибернетики имени В.М. Глушкова
Национальной Академии Наук Украины

Ассоциация Создателей и Пользователей
Систем Искусственного Интеллекта

Редакция международного журнала
«Information Theories and Applications»