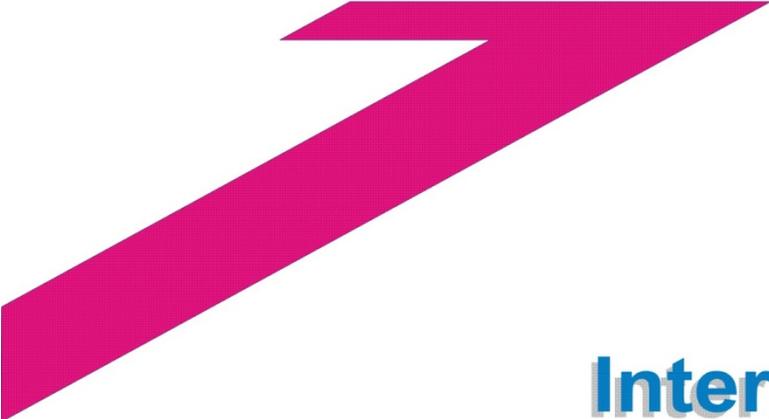


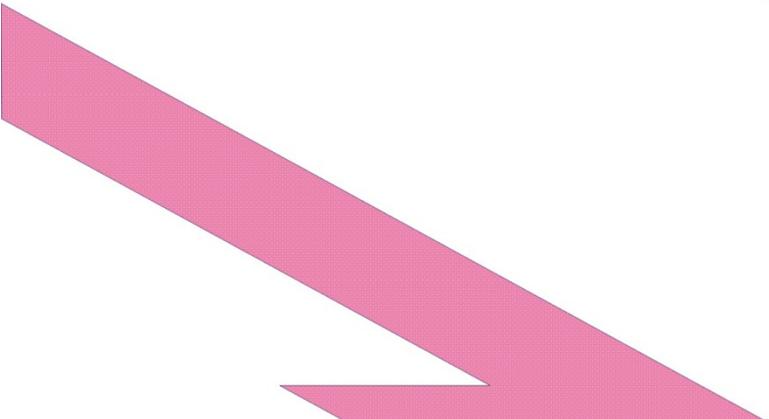


I T H E A



International Journal

INFORMATION TECHNOLOGIES
&
KNOWLEDGE



2014 Volume 8 Number 2



International Journal
INFORMATION TECHNOLOGIES & KNOWLEDGE
Volume 8 / 2014, Number 2

EDITORIAL BOARD

Editor in chief: **Krassimir Markov** (Bulgaria)

Abdelmgeid Amin Ali	(Egypt)	Larissa Zaynutdinova	(Russia)
Adil Timofeev	(Russia)	Laura Ciocoiu	(Romania)
Aleksey Voloshin	(Ukraine)	Levon Aslanyan	(Armenia)
Alexander Kuzemin	(Ukraine)	Luis F. de Mingo	(Spain)
Alexander Palagin	(Ukraine)	Natalia Ivanova	(Russia)
Alexey Petrovskiy	(Russia)	Nataliia Kussul	(Ukraine)
Alfredo Milani	(Italy)	Natalia Pankratova	(Ukraine)
Arnold Sterenharz	(Germany)	Nelly Maneva	(Bulgaria)
Avram Eskenazi	(Bulgaria)	Nikolay Lyutov	(Bulgaria)
Axel Lehmann	(Germany)	Orly Yadid-Pecht	(Israel)
Darina Dicheva	(USA)	Rafael Yusupov	(Russia)
Ekaterina Solovyova	(Ukraine)	Rumyana Kirkova	(Bulgaria)
George Totkov	(Bulgaria)	Stoyan Poryazov	(Bulgaria)
Hasmik Sahakyan	(Armenia)	Tatyana Gavrilova	(Russia)
Iliia Mitov	(Bulgaria)	Vadim Vagin	(Russia)
Irina Petrova	(Russia)	Vasil Sgurev	(Bulgaria)
Ivan Popchev	(Bulgaria)	Velina Slavova	(Bulgaria)
Jeanne Schreurs	(Belgium)	Vitaliy Lozovskiy	(Ukraine)
Juan Castellanos	(Spain)	Vladimir Ryazanov	(Russia)
Julita Vassileva	(Canada)	Volodimir Doncheko	(Ukraine)
Karola Witschurke	(Germany)	Martin P. Mintchev	(Canada)
Koen Vanhoof	(Belgium)	Yuriy Zaychenko	(Ukraine)
Krassimira B. Ivanova	(Bulgaria)	Zhili Sun	(UK)

International Journal "INFORMATION TECHNOLOGIES & KNOWLEDGE" (IJ ITK)
 is official publisher of the scientific papers of the members of
 the ITHEA International Scientific Society

IJ ITK rules for preparing the manuscripts are compulsory.
 The **rules for the papers** for IJ ITK are given on www.ithea.org

Responsibility for papers published in IJ ITK belongs to authors.

International Journal "INFORMATION TECHNOLOGIES & KNOWLEDGE" Volume 8, Number 2, 2014
 Edited by the **Institute of Information Theories and Applications FOI ITHEA**, Bulgaria, in collaboration with:
 Institute of Mathematics and Informatics, BAS, Bulgaria; V.M.Glushkov Institute of Cybernetics of NAS, Ukraine;
 Universidad Politecnica de Madrid, Spain; Hasselt University, Belgium;
 St. Petersburg Institute of Informatics, RAS, Russia; Institute for Informatics and Automation Problems, NAS of the Republic of Armenia.

Printed in Bulgaria
Publisher ITHEA®

Sofia, 1000, P.O.B. 775, Bulgaria. www.ithea.org, e-mail: info@foibg.com

Technical editor: Ina Markova

Издател: ИТЕА®, София 1000, ПК 775, България, www.ithea.org, e-mail: info@foibg.com

Copyright © 2014 All rights reserved for the publisher and all authors.

© 2007-2014 "Information Technologies and Knowledge" is a trademark of ITHEA®

© ITHEA® is a registered trademark of FOI-Commerce Co.

ISSN 1313-0455 (printed)

ISSN 1313-048X (online)

КЛАССИФИКАТОР ДЛЯ СТАТИЧЕСКОГО ОБНАРУЖЕНИЯ КОМПЬЮТЕРНЫХ ВИРУСОВ, ОСНОВАННЫЙ НА МАШИННОМ ОБУЧЕНИИ

Евгений Путин, Адиль Тимофеев

Аннотация: Обсуждаются оригинальные подходы к построению системы обнаружения компьютерных вирусов. Обосновывается актуальность разработки и реализации эффективного классификатора как для известных, так и для ранее неизвестных вирусов (вирусов нулевого дня). Дается общий подход к построению классификатора для статического обнаружения вирусов с использованием машинного обучения с учетом анализа последних исследований в данной области. Описываются результаты новых исследований, и обосновывается выбор итоговой агрегационной модели – стекового классификатора.

Ключевые слова: классификатор, обнаружение, компьютерные вирусы, машинное обучение

ACM Classification: E4. Coding and Information Theory

Введение

В настоящее время использование глобальной сети Интернет является неотъемлемой частью нашей повседневной жизни. Через браузеры можно скачивать различный контент, в том числе программное обеспечение. Сегодня многие компьютерные системы становятся уязвимы к «зловредным» программам, т.е. программам, нацеленным на нанесение вреда конечному пользователю, или компании. Зловредные программы могут быть категоризованы на несколько групп:

- Вирусы – компьютерные программы, которые размножают себя и внедряются в файлы пользователя или операционной системы;
- Черви – саморазмножающиеся компьютерные программы, которые способны посылать себя на другие компьютеры по локальной сети или Интернету;
- Трояны – программы, которые маскируют себя под желанную функциональность, но на самом деле реализуют другие «невидимые» операции, такие как неавторизованные доступ.
- Шпионы – программы, установленные в компьютере без осведомления пользователя для того, чтобы собирать о нем нужную информацию.

Каждая из указанных групп имеют свою уникальную специфику, но всех их объединяет операционная система Windows (в нашем случае 32-х битная), а значит и единый формат представления исполняемых файлов. Поэтому в дальнейшем под компьютерным вирусом будем просто понимать любую программу (win32) которая причиняет вред.

На Рис. 1 представлена гистограмма роста компьютерных вирусов [1]

Гистограмма проверяемости файлов на заражённость и инфицируемость представлена на Рис. 2 [2].

Как видно из рис. 2, самый проверяемый тип файлов Win32 Exe, что не удивительно из-за множества уязвимостей в Windows OS.

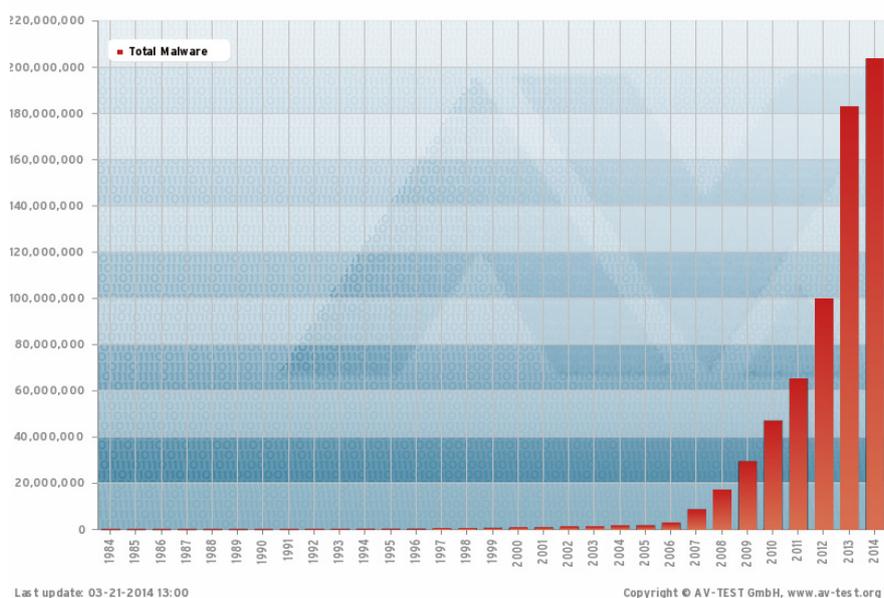


Рисунок 1. Гистограмма роста количества вирусов во времени

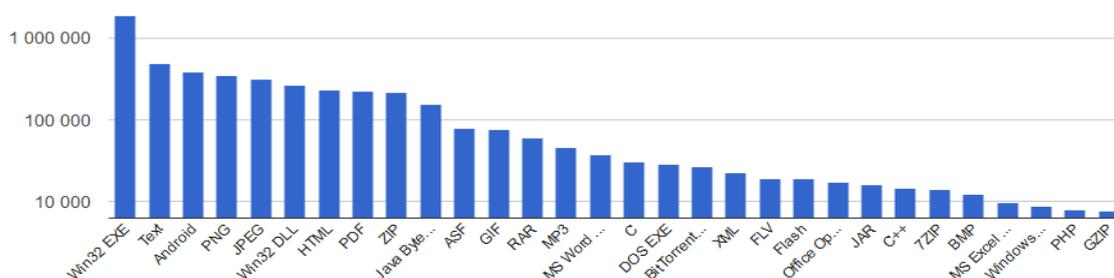


Рисунок 2. Гистограмма проверяемости файлов на зараженность и инфицируемость (по оси Y - количество проверяемых файлов за последние 7 дней (30.03.14), по оси X – типы проверяемых файлов).

1. Подходы к обнаружению вирусов и актуализация

Методы обнаружения вирусов могут быть разделены на 2 класса :

1. Методы, основанные на сигнатурах.
2. Методы, основанные на выявлении аномалий.

В большинстве современных антивирусов центральное место занимает сигнатурный подход. Он даёт 100% точность обнаружения на уже известных вирусах. Но сигнатурный анализ бесполезен на тех вирусах, которые не известны антивирусу, т.е. не известны их сигнатуры. Будем называть такие вирусы, вирусами нулевого дня (zero-day viruses). Аномальный подход наоборот позволяет обнаруживать вирусы нулевого дня. Чаще всего аномальные методы строят классификаторы или решающие правила. Существует немало таких методов [3], но в последнее время наибольший потенциал представляют методы, использующие машинное обучение.

Сигнатурные методы обнаруживают вирус при помощи поиска сигнатур уже известных вирусов в специальном словаре – базе сигнатур вирусов.

Аномальные методы обнаруживают вирус, используя знания, правила и спецификации о нормальном поведении той или иной программы.

Преимущества и недостатки сигнатурного анализа:

1. Позволяет определять конкретный вирус с высокой точностью и малой долей ложных срабатываний.
2. Беззащитен перед полиморфными вирусами.
3. Требуется регулярного и крайне оперативного обновления базы сигнатур.
4. На неизвестные вирусы требуются эксперты для ручного анализа вирусов и выделения сигнатур.
5. Неспособен выявить какие-либо новые вирусы, атаки.
6. На разные версии одного и того же вируса необходимы разные сигнатуры.
7. С учетом того, что база сигнатур огромна, сигнатурный анализ очень ресурсоемкая операция.

Преимущества и недостатки аномального анализа:

1. Возможность обнаружения ранее неизвестных вирусов (вирусов нулевого дня (zero-day viruses)).
2. Высокая вероятность ложных срабатываний, т.е. таких при которых доброкачественная программа была распознана как вирус.
3. Высокая сложность обучения системы.
4. Лечение неизвестного вируса практически всегда является невозможным.
5. Для уже обученной системы анализ выполняется сравнительно быстро (нужно лишь извлечь признаки).

И сигнатурные и аномальные методы внутри себя могут использовать три различных подхода для обнаружения вирусов:

1. Статический подход. Используя этот подход, подозрительная программа анализируется статически (т.е. без запуска самой программы) как обычный файл.
2. Динамический подход. При этом подходе, подозрительная программа анализируется динамически, т.е. во время ее выполнения в реальном времени.
3. Гибридный подход. Объединение статического и динамического подходов в разных частях анализа «зловредной» программы.

Ниже на рис. 3 приведена «карта ума» (mind map) по подходам к обнаружению вирусов.



Рисунок 3. Карта ума по методам обнаружения вирусов.

В современном мире разрабатываются таргетированные вирусы и совершаются таргетированные атаки, т.е. такие атаки, которые нацелены на конкретную компанию, организацию, страну.

К примеру, по опубликованным данным газеты New York Times [4] известный компьютерный червь Stuxnet, который по некоторым предположениям был специально разработан против компьютерной сети ядерного проекта Ирана, использовал 4 уязвимости нулевого дня и 3 известных. Более детальный обзор

вируса Stuxnet и уязвимостей нулевого дня можно найти в последнем [5] Также очень хорошо цифровая актуализация описана в отчете Лаборатории Касперского [6].

Из всего вышесказанного следует необходимость развития, усовершенствования и создания новых методов обнаружения вирусов в области аномального анализа, в том числе и в статическом подходе.

Обнаружение вирусов при помощи методов машинного обучения (МО) представляет огромный потенциал в статистическом обнаружении вирусов, потому что все методы МО призваны обобщать решение конкретной задачи на общий класс аналогичных задач, и показывают практически лучшие результаты во всех задачах классификации и прогнозирования.

2. Общий подход к созданию классификатора

В начале исследований формируется и определяется выборка вирусов и чистых фалов.

Создание классификатора для обнаружения вирусов условно можно разделить на три стадии:

1. Формирование признакового описания файла (features extraction). Результатом этой стадии является вектор, содержащий признаки характеристики рассматриваемого объекта. В задаче построения классификатора статического обнаружения вирусов признаками могут выступать следующие объекты:

- Строки – исполняемый файл рассматривается как обычная строка или последовательность строк. Признаки – числовые характеристики строк (например, частота нулей в подстроке);
- Структурные элементы Portable Executable файла. Эта специальная информация встроена во все файлы Win32 и Win64. Она необходима для загрузчика операционной системы Windows и для самого приложения. Подробная документация доступна (см. например,[7]). Признаки, извлеченные из структурной информации PE файлов могут быть следующими: сертификат, date/time stamp, файловый указатель – позиция внутри файла информация компоновщика, тип CPU, логическая информация (выравнивание секций, размер, секции кода, отладочные флаги), информация об импорте – список тех DLL, которые использует исполняемый файл, и экспорте – те функции которые предоставляет другим приложениям, таблицу релокаций (перемещений) директории ресурсов – иконки, кнопки и прочее.
- N-граммы на уровне байт. Сегменты последовательных байт из разных мест внутри исполняемого файла длины N. Каждая N-грамма рассматривается как признак. К примеру, количество байтовых биграмм – $256 \times 256 = 65536$ штук. Таким образом, будем рассматривать 65536 признаков.
- N-граммы на уровне опкодов. Опкод (opcode – operation code) – специфичный для CPU операционный код, который выполняет специальную машинную команду (например, mov, push, add).

2. Выбор признаков (feature selection). В течение этой фазы вектор, созданный на стадии 1 вычисляется, а избыточные и нерелевантные признаки выбрасываются из рассмотрения. Выбор признаков имеет много преимуществ: увеличение выполнения обучающейся модели за счет сокращения количества необходимых операций и, как следствие, увеличение скорости обучения, повышение обобщающей способности за счет сокращения размерности пространства признаков, удаление “выбросов”, лучшая интерпретируемость и т.п. Задача этой стадии заключается в том, чтобы из уже имеющихся признаков выбрать наиболее значимые (информативные). Существует несколько подходов к выделению информативности признаков. Наиболее популярными являются корреляционные и фильтровые методы [4].

3. Построение математической модели, классификатора, который использует разреженный вектор, полученной стадии 2. Для построения классификатора могут использоваться следующие математические модели:

- Деревья решений (decision trees, DT);
- Случайный лес (random forest, RF);
- Градиентный бустинг (gradient boosting machines, GBM);
- Логистическая регрессия и ее оптимизации (logistic regression, LR);
- Метод опорных векторов (support vector machines, SVM);
- K-ближайших соседей (K-nearest neighbor, kNN);
- Adaboost;
- Наивный Байес (Naive Bayes, НБ, NB);
- Нейронные сети (НС, NN).

3. Анализ исследований в области создания классификатора статического обнаружения вирусов

По результатам анализа проделанных ранее исследований можно сделать вывод, что не во всех работах присутствуют описанные выше фазы. Например, часто выбор признаков опускается. При этом авторы полагают, что большое количество признаков существенно не ухудшит результирующую модель, так как существуют устойчивые к «передозированным» признакам модели.

Следует сказать, что анализ и сравнение результатов исследований является сложной задачей, поскольку разные авторы используют различные обучающие множества. Кроме того, сравнение моделей происходит по различным метрикам.

В своей работе 2001 года [8] автор предложил следующий метод:

1. Для каждого исполняемого файла рассматриваются три различных признака:

- Список DLL, использующихся внутри исполняемого файла;
- Список системных вызовов DLL;
- Количество различных системных вызовов внутри каждой DLL.

2. Выбор признаков, осуществлялся специальным индуктивным алгоритмом Ripper [9] для нахождения паттернов в данных DLL.

3. В качестве математической модели выступал Наивный Байесовский (NB) классификатор, который использовался для нахождения паттернов в строковых данных, а N-граммы последовательностей байт были использованы, как вход для Мультиномиального Наивного Байеса.

Обучающее множество состояло из 4266 файлов, из которых было 3265 вирусов и 1001 чистых файлов. Точность классификации для такого множества, подхода и модели была 97,11%.

В работе за 2006 год [10] использовался N-граммный по байтам подход. Их обучающее множество состояло из 1971 чистых файлов и 1651 зловредных. Выбор признаков проходил с использованием Information Gain. Были выбраны 500 наиболее часто встречаемых N-грамм. Ученые пробовали множество различных моделей (Naïve Bayes, SVM, KNN и др.), но лучшая из них – Adaboost дала 0.98 точности с 0,05% ложных срабатываний.

В работе [11] авторы использовался близкий к подходу, описанному в [8], а именно:

1. Из каждого исполняемого файла извлекались:

- Структурные признаки формата PE: вся информация о заголовках PE и секциях;
- Список всех используемых DLL;
- Список всех функций внутри каждой DLL.

2. Все эти признаки были отфильтрованы по Information Gain (IG) и были выбраны 20 наиболее информативных признаков.

3. Рассматривалось несколько моделей (SVM, NB, DT), причём лучшей оказалась DT (C4.5 реализация) которая показала результат в 99,6% точности с 2,7% ложных срабатываний.

Исследователи использовали открытую вирусную коллекцию VX heaven [12] (которая на 2010 год насчитывала ~ 230 тысяч PE вирусов, червей и агентов) и собранную коллекцию чистых файлов в 10592. При этом много внимания уделялось выбору значимых признаков и построению моделей на различных подмножествах признаков, в том числе уменьшению размерности (пространства признаков), что дало чуть худший результат, чем ранее приведенный.

В фундаментальной работе [13] исследовался подход, основанный на опкодах. В качестве обучающего множества была взята VX heaven коллекция. В работе [13] дается детальный анализ использования и выделения опкодовых N-граммов. Перебором различных N лучший результат достигается на $N = 2$. При этом производилось построение множества моделей (RF, DT, NB, KNN, SVM) с оптимизациями, и сравнение моделей. Была выбрана лучшая модель, а именно SVM, с полиномиальным ядром для биграмм дала 95,90% точности с 0,03% ложных срабатываний.

В работе [14] использовался строковый подход. Для большей масштабируемости подхода использовались ансамбли SVM с бэггингом (Bagging, Bootstrap Aggregating). Обучающее множество состояло из 39838 исполняемых файлов, из которых 8320 было чистых файлов и 31518 вирусов, червей, троянов и агентов файлов. Идея подхода состояла в том, чтобы извлекать из файлов интерпретируемые строки и использовать их в качестве прецедентов для обучения ансамбля SVM с последующей агрегацией. К примеру строка "`<html> <script language = 'javascript'> window.open('readme.eml')`" всегда присутствует в червях "Nimda". Авторы работы [12] извлекли 13448 строк из всего обучающего множества, а для селекции признаков воспользовались алгоритмом Max-Relevance, который отранжировал признаки, выбрав таким образом 3000 значимых признаков. Итоговой результат с примененной моделью – 92,22% точности.

4. Краткое изложение авторского исследования

За основу в проведенных исследованиях была взята VX Heaven коллекция вирусов, в которой насчитывается ~ 230 тысячи PE вирусов. Чистые файлы собирались с различных версий операционной системы Windows (95, XP, Vista, 7), свободного проекта Cygwin и другого свободного программного обеспечения взятого с сайта download.com. Выбирались исключительно файлы формата Portable Executable, т.е. файлы с расширением exe, dll, sys. Всего было собрано 17746 чистых файлов.

Таким образом все обучающее множество представляло 248 тысяч исполняемых файлов.

Целью работы было:

- Исследовать существующие подходы статического анализа обнаружения вирусов с применением машинного обучения;
- Провести собственные независимые эксперименты;
- Разработать собственный классификатор, способный агрегировать различные признаковые описание одних и тех же файлов;

- Сравнить результаты приведенных исследований с собственными наработками.

Заметим, что у каждого из ранее описанных подходов есть свои принципиальные недостатки:

- У подхода с выделением строк недостатком в том, что на разные классы вирусов приходится разные строковые описатели, а в целом необходимо формировать общее представление о всех рассматриваемых объектах. По этой причине он дает, худший результат, чем остальные подходы, но лучше может классифицировать конкретный тип зловредных программ;
- У подхода с N-граммами байт, хотя он дает хорошие результаты, очень слабая интерпретация с точки зрения исследователя. Например, трудно заключить, почему последовательность 0000 0000 0000 0001 более значима чем 0000 0000 0000 0010 и совершенно неочевидно, какой будет точность при увеличении исходного множества исполняемых файлов;
- У подхода с N-граммами опкодов хорошая интерпретация (за счет того что есть потенциально опасные последовательности опкодов которые известны), но не самый лучший результат по точности и дизассемблированию кода;
- У подхода со структурными признаками, хорошие результаты. Однако выделенную структурную информацию можно подделать. Если разработчик вируса пользуется стандартными средствами разработки (к примеру, Visual Studio), то структурная информация вирусного файла не сильно отличается от чистого, так как по умолчанию используются стандартные средства компиляции и линкера.

Из приведенного следует, что единственно верного решения нет, и имеет смысл рассматривать комбинации подходов. В предлагаемом подходе рассматривается комбинация N-грамм опкодов и структурных признаков PE файла.

Следует отметить, что весь статический анализ очень чувствителен к запакованным исполняемым файлам и применению обфускации на бинарном уровне. Запакованный исполняемый файл – это исполняемый файл внутри которого находится подпрограмма запаковывающая его и делающая недоступным правильное чтение его структурных элементов. Также бессмысленно дизассемблировать запакованный файл, потому что сначала будет дизассемблироваться сам пакер, а это не то, что требуется. Применение обфускации на бинарном уровне портит дизассемблирование файла, но не отражается на структурных элементах. Поэтому в исследовании рассматриваются только незапакованные исполняемые файлы. Для этих целей была специально разработана программа, которая и распаковывает все файлы.

Условно исследование можно разделить на 3 части:

1. Исследование структурного подхода и построение классификаторов для него.
2. Исследование N-грамм для опкодов и построение классификаторов для них.
3. Агрегация классификаторов с 1 и 2 стадии в один общий стековый классификатор.

Для структурного подхода рассматривались 173 признака – это вся информация о заголовках PE: Optional, File, Dos headers, отдельно поля DataDirectory и 8 основных секций: text(code), data, idata, edata, reloc, rsrc, debug, bss и tls. Поля секций которых не было в исполняемом файле обнулялись. Затем, выбрасывая из рассмотрения признаки с малым количеством уникальных значений (<10), строилось множество моделей более менее устойчивых к передозировке признаками. Признаки, у которых был большой разброс в значениях, факторизовались, т.е. происходила перенумерация с присвоением. Лучшие результаты в 99,6% и 99,2% точности показали модели RF и GBM, соответственно. Также проводились исследования связанные с уменьшением размерности, в частности с использованием PCA

Далее строился агрегированный классификатор. На вход ему подается матрица из предсказаний других классификаторов. Суммарно было построено 20 классификаторов (по 10 на каждый подход) и поэтому столбцов у матрицы предсказаний было 20.

Суть агрегированного классификатора в том, чтобы стабилизировать и сгладить предсказания разных классификаторов натренированных на разных признаковых подмножествах описывающих одну и ту же задачу. Такой прием в машинном обучении называется Stacking или StackedGeneralization [15]. Он позволяет улучшить обобщающую способность классификатора, в некоторых случаях улучшая общую точность до 5%, во многом благодаря сглаживанию и уменьшению количества ложных срабатываний. С учетом того, что данные с предсказаний классификаторов распределены близко к линейному, то чаще всего для стекового классификатора выбирается линейная модель, что и происходит в нашем случае.

В итоге стековый классификатор дал общую точность в 99,87% с 0,01% ложных срабатываний.

Заключение

Итоговый стековый классификатор, как и ожидалось, дал лучшую общую точность и меньшее число ложных срабатываний, за счет оптимального объединения построенных на предыдущих шагах классификаторов.

В планах на будущее:

- Провести исследование с целью адаптации классификатора статического обнаружения вирусов с использованием машинного обучения на случаи запакованных, обфусцированных файлов. Такая цель может быть достигнута более глубокой предфильтрацией исходных файлов, к примеру можно выделить общие сегменты (байт, опкодов) присущие чистым файлам и соответственно вирусам и рассматривать в качестве признаков распределение этих общих сегментов внутри каждого файла;
- Провести исследование с большим количеством используемых моделей и автоматической селекцией признаков. К примеру, нейронные сети, а также глубокие нейронные сети представляют огромный потенциал в автоматической селекции признаков;
- Провести исследование с целью распознавания конкретных классов вирусов.

Работа выполнена при поддержке грантов РФФИ № 14-08-01276 и № 12-08-01167-а.

Библиография

[1] <http://www.av-test.org/en/statistics/malware/>

[2] <https://www.virustotal.com/en-gb/statistics/>

[3] Nwokedi Idika, Aditya P. Mathur "A Survey of Malware Detection Techniques" 2007

[4] <http://www.nytimes.com/2011/01/16/world/middleeast/16stuxnet.html?pagewanted=all>

[5] http://go.eset.com/us/resources/white-papers/Stuxnet_Under_the_Microscope.pdf

[6] http://media.kaspersky.com/documents/business/brfwn/ru/Advanced-persistent-threats-not-your-average-malware_Kaspersky-Endpoint-Control-white-paper-ru.pdf

[7] <http://msdn.microsoft.com/en-us/library/gg463119.aspx>

[8] M. G. Schultz, E. Eskin, E. Z., and S. J. Stolfo, "Data mining methods for detection of new malicious executables," in Proceedings of the IEEE Symp. on Security and Privacy, pp. 38-49, 2001.

-
- [9] W. Cohen "Fast effective rule induction." Proc. 12th International Conference on Machine Learning, pp. 115-23, San Francisco, CA: Morgan Kaufmann Publishers, 1995.
- [10] J.Z. Kolter and M.A. Maloof. "Learning to detect and classify malicious executables in the wild." The Journal of Machine Learning Research, 7:2721–2744, 2006.
- [11] Usukhbayar Baldangombo, Nyamjav Jambaljav, and Shi-Jinn Horng "A Static Malware Detection System Using Data Mining Methods", International Journal of Artificial Intelligence & Applications; Jul2013, Vol. 4 Issue 4, p 113
- [12] vxheaven.org
- [13] Igor Santos, Felix Brezo, Xabier Ugarte-Pedrero, Pablo G. Bringas "Opcode Sequences as Representation of Executables for Data-mining-based Unknown Malware Detection" Published in Information Sciences: an International Journal Volume 231, May, 2013 Pages 64-82
- [14] Y. Ye, L. Chen, D. Wang, T. Li, Q. Jiang, and M. Zhao. "Sbmds: an interpretable string based malware detection system using svm ensemble with bagging" Journal in Computer Virology, 5 (4):283–293, 2009.
- [15] Georgios Sigletos Georgios Paliouras Constantine D. Spyropoulos "Combining Information Extraction Systems Using Voting and Stacked Generalization", Journal of Machine Learning Research 6 (2005) 1751-1782

Сведения об авторах



Тимофеев Адиль Васильевич – главный научный сотрудник лаборатории речевых и мультимедийных интерфейсов Санкт-Петербургского института информатики и автоматизации Российской академии наук, Профессор кафедры информатики математико-механического факультета Санкт-Петербургского государственного университета, доктор технических наук, профессор, Заслуженный деятель науки РФ, 199178, Россия, Санкт-Петербург, 14-я линия, д. 39, СПИИРАН, e-mail: tav@iias.spb.su



Путин Евгений Олегович – студент V курса математико-механического факультета Санкт-Петербургского государственного университета, 199178, Россия, Санкт-Петербург, 14-я линия, д. 29, математико-механический факультет СПбГУ, e-mail: putin.evgeny@gmail.com

Classifier for static detection of computer viruses, based on machine learning

Evgeniy Putin, Adil Timofeev

Abstract: Original approaches to building a system for detection of computer viruses are discussed in this work. The actuality of developing and implementing an effective classifier for both known and previously unknown viruses (zero-day) is grounded. A general approach to construction of a classifier for static virus detection using machine learning, based on analysis of recent research, is outlined in the paper. The results of new research are presented. The choice of the final aggregation model - Stack classifier - is grounded.

Keywords: classifier, detection, computer viruses, machine learning

ПРОГРАММНАЯ СИСТЕМА ДЛЯ РЕШЕНИЯ ЗАДАЧИ СИНТЕЗА SYNTHESIS 1.0

Александр Докукин

Abstract: В статье представлено описание программной системы, предназначенной для решения введенной ранее задачи синтеза в рамках алгебраической теории распознавания. В основе алгоритмов системы лежат теоретические результаты, также описанные ранее. Приводятся оценки скорости работы и качества решения задачи для реальных данных больших объемов.

Keywords: алгебраическая теория распознавания, задача синтеза, программная система.

ACM Classification Keywords: I.5 Pattern Recognition — I.5.0 General.

Введение

Термин задача синтеза используется в данной статье для обозначения определенного нового типа задач в рамках теории распознавания. А именно, задачи поиска в генеральной совокупности объекта, обладающего в наибольшей степени выраженным свойством, причем свойство это описано только набором прецедентов, т.е. объектов, обладающих или не обладающих этим свойством. При этом, предполагается, что набор прецедентов неполон, т.е. в генеральной совокупности есть неизвестные объекты, и, может быть, противоречив.

Такая общая постановка возникла при решении частной прикладной задачи — синтеза белков с заданным профилем структуроформирующих свойств. Задача решается в предположении, что элементарной единицей, отвечающей за обладание данным отрезком белка структуроформирующими свойствами, является пятибуквенная аминокислотная последовательность [Nekrasov, 2004]. Эта гипотеза получила, в том числе, косвенное подтверждение, поскольку позволила разделить представителей двух классов (имеющих и не имеющих структуроформирующие свойства) методами распознавания с точностью 89% [Senko et al, 2011].

Строгая формальная постановка задачи, а также методы её решения и их обоснование опубликованы ранее [Dokukin, 2013a; Dokukin, 2013b]. В данной статье мы кратко опишем только основную идею метода, понимание которой необходимо для прикладного применения соответствующих алгоритмов. Основной же целью данной статьи является описание программной системы, реализованной на основе полученных ранее результатов.

Метод синтеза

Общая идея решения задачи синтеза заключается в следующем. Если нет возможности оценить выраженность некоторого свойства объекта явно, но имеются прецеденты его наличия или отсутствия, можно обучить на этом наборе прецедентов алгоритм распознавания, и использовать для оценки выраженности свойства оценку принадлежности объекта одному из классов, вычисляемую алгоритмом.

Для увеличения скорости работы, а также для упрощения интерпретации получаемых результатов был разработан особый метод логических закономерностей [Dokukin, 2013a]. Поскольку мы говорим о распознавании пятибуквенных слов в определённом алфавите, логические закономерности приобрели

форму шаблонов слов, т.е. слов с фиксированными буквами в некоторых позициях и пропусками в остальных. В местах пропусков допустимы любые буквы.

Набор закономерностей строится по набору прецедентов, этот этап называется этапом обучения. Качество набора закономерностей можно оценить, распознав на его основе независимую выборку. Этот этап решения задачи, очевидно, будем называть распознаванием.

Последний, третий этап заключается собственно в синтезе, т.е. поиску по заданному шаблону слова с заданными свойствами. Здесь можно выделить два принципиальных случая, которые были названы элементарной и общей задачей синтеза. Элементарная задача заключается в поиске слова по шаблону той же длины, что и прецеденты. При этом, задан целевой класс, соответствующий или наличию, или отсутствию некоторого свойства. Общая задача заключается в поиске слова по шаблону, длина которого существенно превышает длину прецедентов. При этом, на шаблоне задан профиль целевого свойства, т. е. размечены участки его наличия или отсутствия.

Универсальный подход к решению задачи синтеза заключается в независимом восстановлении каждой пропущенной буквы по всему набору шаблонов содержащих её. Например, если имеется шаблон АВ-С*D--Е и нас интересует восстановление центральной буквы, необходимо выбрать наилучшего кандидата, удовлетворяющего набору шаблонов АВ-С*, В-С*D, -С*D-, С*D--, *D--Е. Влияние букв в других пропусках не учитывается. Показано, что при определенных, легко проверяемых, условиях на набор закономерностей такой подход даёт оптимальное решение.

Программная система

В описываемой программной системе реализованы алгоритмы для решения всех трёх описанных этапов, а именно: обучения, распознавания и синтеза. При этом, система предоставляет визуальные средства управления, представления промежуточных результатов и т.д.

Для работы с программой необходимо либо создать новый проект, либо загрузить уже существующий. Ключевой характеристикой проекта является набор прецедентов, по которым производится обучение алгоритмов синтеза. То есть, проект предназначен для работы с одним типом задачи. При этом, система позволяет одновременно работать с несколькими проектами.

Основное окно системы разделено на три части. Самая левая содержит дерево проектов и позволяет переключаться между ними, а также между отдельными их свойствами.

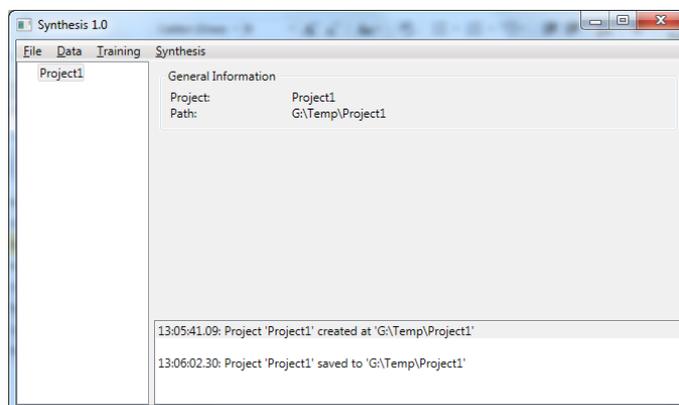


Рисунок 1. Общий вид проекта

Справа сверху расположено окно свойств. Это окно содержит описание текущего элемента дерева проектов. Если выбран проект, то в окне кратко перечисляются основные свойства проекта. Иначе, отображается подробная информация, соответствующая текущему этапу работы.

Наконец, справа внизу расположен журнал проекта, в котором отражаются основные этапы работы с ним. Журнал содержит точные временные отметки, что позволяет оценить продолжительность отдельных этапов.

Исходные данные предоставляются в виде набора из двух текстовых файлов, каждый из которых содержит все прецеденты одного из классов. Формат данных таков: первые пять символов каждой строки соответствуют пятибуквенному прецеденту, остальные символы — комментарий.

Таким образом, принципиальными ограничениями на тип исходной информации является фиксированная длина прецедентов и бинарность целевого профиля. Кроме того, мощность алфавита ограничена допустимым набором аминокислот. Если данные соответствуют этим требованиям, но не имеют отношения к белкам или к их структуроформирующим свойствам, применение всех алгоритмов по-прежнему допустимо. Например, можно работать не со структурообразующими свойствами самими по себе, а с определенной пространственной структурой. Бинарный признак при этом может быть задан в виде "Определенная структура" и "Все остальные". Обучающая выборка должна быть приведена в соответствие данному признаку.

Исходные данные представлены в свойстве "Training Data" проекта.

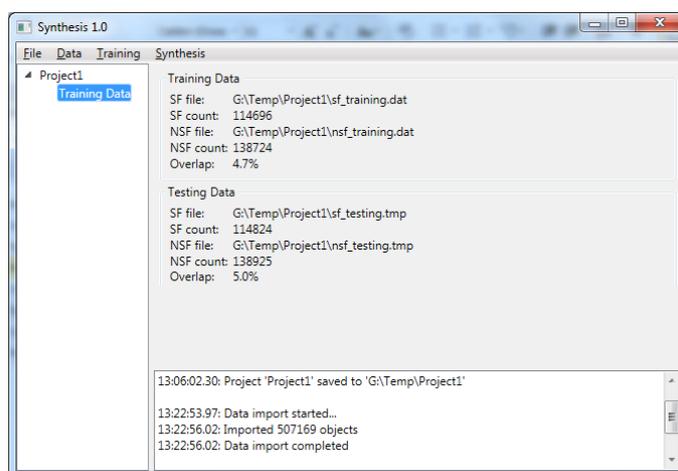


Рисунок 2. Отчёт об исходных данных

В этой версии описание исходных данных состоит только из отчёта. Отчёт содержит две одинаковые секции, посвящённые обучающим (Training Data) и тестовым (Testing Data) данным. Они содержат данные о расположении файлов для обоих классов (SF и NSF file), количество объектов в каждом классе (SF и NSF count) и процент совпадений прецедентов в двух классах (Overlap).

Обучение алгоритма для текущего проекта производится из меню "Training → Train". Ключевой особенностью разработанного метода является высокая скорость работы. Обработка выборки из пятисот тысяч объектов занимает около двадцати секунд на обычном персональном компьютере. Результаты обучения будут представлены в свойствах "Training Results" проекта.

Отчёт включает статистику числа найденных закономерностей по рангам (Rank N Count), а также статистику нерегулярных закономерностей (Irregular N Count), т.е. закономерностей, которые не являются уточнениями закономерностей меньшего ранга.

Template	Rank	Information Value
--L--	1	13868.17
--A--	1	4558.55
--S--	1	3839.81
--G--	1	7875.44

Template	Rank	Information Value
--H--	1	6812.16
--M--	1	6787.81
--W--	1	6636.25
--C--	1	6123.89

Рисунок 3. Результаты обучения

Кроме того, на этом этапе в удобном виде представлен полный набор найденных закономерностей для каждого из классов (SF и NSF Regularities). Таблицы содержат шаблон закономерности (Template), её ранг (Rank), значимость (Information Value) и регулярность (Is Regular). Таблицы могут быть отсортированы по любому из этих полей.

Проверка результатов обучения является важной частью оценки качества полученного решения. Здесь под проверкой понимается проверка качества распознавания независимой выборки, т.е. обученный алгоритм распознавания приписывает некоторый класс неизвестным ему объектам, и проверяется доля верных ответов алгоритма. Как уже упоминалось выше, наилучший достигнутый результат с небольшими (до нескольких тысяч объектов) обучающими выборками составил 89% [Senko et al, 2011]. Разработанный метод с большими "грязными" выборками (допускающими пересечение классов), дал результат на уровне 78% [Dokukin, 2013a].

Проверка также занимает незначительное время. Для полумиллиона объектов это менее десяти секунд. Отчёт об обучении появляется в свойствах "Testing Results".

Class	Assigned	Correct	Total	Correct / Real	Correct / Assigned
SF	129920	95127	114824	0.83%	0.73%
NSF	123829	104132	138925	0.75%	0.84%
Total				0.79%	0.79%

Рисунок 4. Отчёт о распознавании независимой выборки

Здесь тоже присутствуют два одинаковых блока для классов SF и NSF. В них отображается количество объектов, отнесённых алгоритмом к классу (Assigned); количество объектов, отнесённых корректно (Correct); общее количество объектов (Total) и две оценки качества: доля правильно угаданных среди объектов класса, и доля правильно отнесённых. Кроме того, присутствует информация, усреднённая по классам.

Обученный алгоритм можно использовать для решения нескольких задач синтеза одновременно. Для этого нужно импортировать соответствующие шаблоны слов и целевые признаки. Шаблоны описываются текстовыми файлами специального формата:

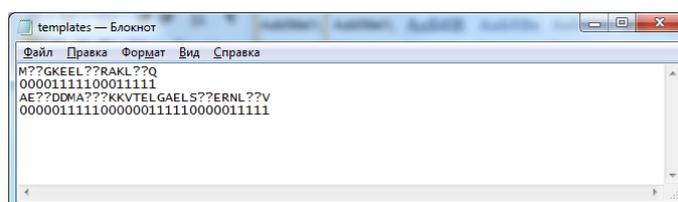


Рисунок 5. Пример файла шаблонов

Здесь, первая строчка задает собственно шаблон, а вторая — профиль целевого признака. Для добавления новых шаблонов в проект надо воспользоваться меню “Synthesis → Import Profiles...”.

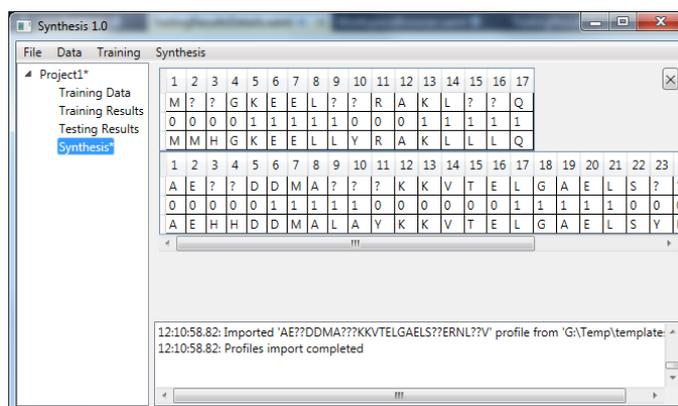


Рисунок 6. Представление шаблонов задач синтеза в программе

Загруженные шаблоны представляются в программе в виде таблицы. Первые две строчки соответствуют загруженным данным, третья — решению задачи, которая решается практически мгновенно. В программе используются обобщённые алгоритмы на случай длинных последовательностей букв синтеза [Dokukin, 2013b]. Таким образом, длина искомых шаблонов может многократно превышать длину обучающих прецедентов.

Шаблоны можно удалять, нажав на соответствующий крестик. В ближайших версиях программы запланировано добавление и редактирование шаблонов вручную, без посредничества текстовых файлов.

Выводы

Нетрудно видеть, что использованные при обучении данные породили SF-шаблон “--L—” с огромным отрывом по значимости, поэтому большинство случаев недостатка информации разрешается в пользу

этой буквы. Однако, в некоторых случаях (см. шаблон 2, позиция 10) окружение существенным образом влияет на решение. Класс NSF дает более разнообразные решения за счёт наличия нескольких лидирующих закономерностей.

Как уже говорилось выше, на данную ситуацию можно влиять, сужая выборку до интересующего типа последовательностей. Мы надеемся с помощью представленной программы обнаружить такие содержательные постановки.

Кроме того, уже сейчас запланировано развитие алгоритмов для случая нескольких классов. Их реализация будет сразу же внедрена в программу, что, как мы надеемся, ещё более повысит её практическую ценность.

Благодарности

Статья опубликована при поддержке ITHEA ISS (www.ithea.org) и ADUIS (www.aduis.com.ua), проект ITHEA XXI.

Литература

[Dokukin, 2013a] Dokukin A. "On a Logical Regularities Based Method of Definite Quality Object Synthesis", International Journal "Information Models & Analyses", 2013, Vol. 2, No. 3, pp. 212–216.

[Dokukin, 2013b] Dokukin A. "On the Formalization of Synthesis Task in Pattern Recognition", Proceedings of the 11th International Conference "Pattern Recognition and Image Analysis: New Information Technologies" (PRIA-11-2013), 2013, Vol. 1, pp. 84–86.

[Nekrasov, 2004] Nekrasov A. N. "Analysis of the Information Structure of Protein Sequences: A New Method for Analyzing the Domain Organization of Proteins", Journal of Biomolecular Structure and Dynamics, 2004, Vol. 21, Iss. 5, pp. 615-623.

[Senko et al., 2011] Senko O. V., Nekrasov A. N., Ryazanov V. V., Dokukin A. A. "Prediction of Structure forming Properties of Protein with Help of Pattern Recognition Methods", Proceedings of 8-th Open German-Russian Workshop "Pattern Recognition and Image Understanding" OGRW-8-2011, pp. 38–40.

Информация об авторе



Alexander Dokukin – Computing Centre of Russian Academy of Sciences, researcher,
40 Vavilova St., Moscow, Russia, 119333; e-mail: dalex@ccas.ru.

Major Fields of Scientific Research: Algebraic Approach to Pattern Recognition.

Software System for Solving Synthesis Problem "Synthesis 1.0"

Alexander Dokukin

Abstract: *The software system for solving previously introduced synthesis task in pattern recognition theory is described. The algorithms used in the system are based on theoretical results described earlier. Estimates of its speed and quality in real-world tasks of large scale are presented.*

Keywords: *algebraic recognition theory, synthesis task, software system*

ИНФОРМАЦИОННАЯ ПОДДЕРЖКА ПОДГОТОВКИ ПРОЕКТОВ УПРАВЛЕНЧЕСКИХ РЕШЕНИЙ ПО ОРГАНИЗАЦИОННОЙ ДЕЯТЕЛЬНОСТИ

Нина Баканова

Аннотация: В статье приводится описание разработки функции информационной поддержки работ по подготовке проектов управленческих решений по исполнению документов входного потока. Разработка функции базируется на методе контент-анализа текста документа на совпадение с терминами справочника распределения должностных полномочий в организации.

Ключевые слова: системы электронного документооборота, организационное управление, информационная поддержка управленческой деятельности, контент-анализ.

ACM Classification Keywords: H.4.2 Types of Systems - Decision support

Введение

В соответствии с нормативами обработки документов, для каждого документа, поступающего в документооборот организации, должно быть подготовлено указание по исполнению решений. Указание содержит управленческое решение по исполнению документа и список исполнителей-подразделений, в компетенции которых решение вопросов указанных в документе. Процесс исполнения может включать в себя неоднократную передачу поручений по служебной иерархии, с целью подключения к работе специалистов, соответствующей компетенции. Подготовка указаний по исполнению решений выполняется на каждом уровне управленческой иерархии, которые проходит документ. Высокая трудоемкость работы в крупных организациях определяется:

- объемом входного потока документов (50-100 тысяч в год);
- объемом поступающих документов (до 100 страниц текста), содержащих отраслевую специфику;
- требованием оперативности решения вопросов, указанных в документах входного потока;
- иерархической структурой нормативов распределения должностных полномочий;
- необходимостью выдачи пояснений по подготовленным проектам указаний.

Для ускорения работ и сокращения рутинных трудозатрат руководителей выполняется предварительный анализ поступившего документа и подготовка проекта указаний, который возложен на советников и помощников руководителей. Подготовленный проект содержит текст и список исполнителей (3 – 4 подразделения). Работа состоит в определении проблематики документа, выборе подразделений – исполнителей для решения вопросов на основании нормативов распределения должностных полномочий, установки иерархии исполнителей (ответственный, соисполнитель).

Подготовка проекта указаний является ответственным участком работы, включает анализ документа, подготовку решения по исполнению, определение всех участников в чьи должностные полномочия входят вопросы, затронутые в документе [Ларин, 2002]. Работа требует высокого профессионализма, хорошего знания проблемной области, сферы ответственности исполнителей.

Структурная разработка функций поддержки

Информационная поддержка работ по подготовке проектов указаний базируется на анализе текста документа методом контент-анализа для поиска совпадений с терминами специального справочника, подготовленным на основании нормативов распределения должностных полномочий в организации. Последующий частотный анализ встречаемости вопросов, относящихся к задачам тех или иных подразделений, позволяет выдать результат анализа в последовательности уменьшения показателя частоты. Структурная разработка функции информационной поддержки проводилась с учетом следующих положений:

- ответственность за принятие решений в организации несет Руководитель;
- все документы, адресованные в организацию, докладываются Руководителю;
- проект указания готовят советники и помощники Руководителя;
- Руководитель контролирует, корректирует и утверждает, подготовленные проекты указаний;
- документы с указаниями (резолуции) поступают к исполнителям-подразделениям;
- подготовка проектов указаний может выполняться на следующем иерархическом уровне.

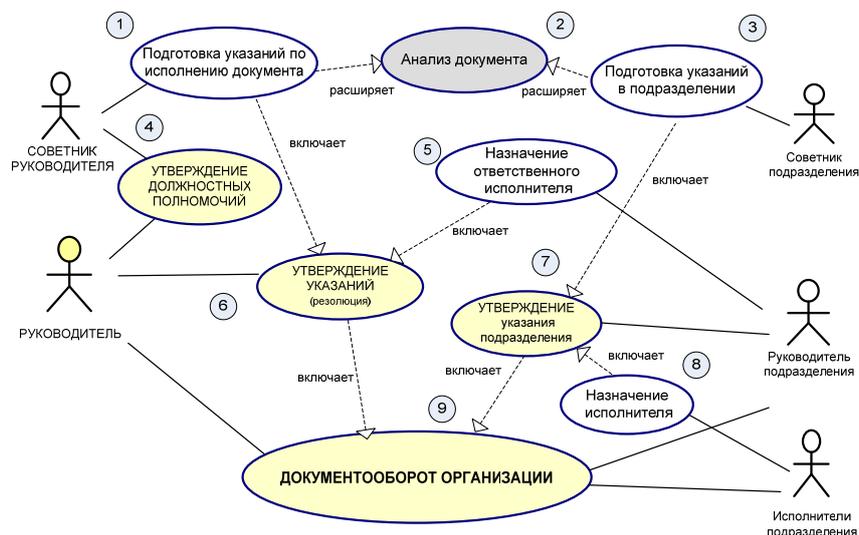


Рисунок 1. Подготовка проекта указаний по исполнению решений

На Рисунке 1 представлена схема взаимодействия подразделений организации в процессе подготовки и утверждения проекта указаний по исполнению управленческих решений в виде диаграммы «вариантов использования» (ВИ) объектно-ориентированного метода (ООП) разработки информационных систем [Гради Буч, 1999].

На диаграмме показаны участники процесса и используемые ими функции системы документооборота. С левой стороны – участники высшего уровня структурной иерархии организации, обеспечивающие начальный анализ документа и подготовку концептуальных решений. С правой стороны - участники, выполняющие функции исполнения решений на следующих уровнях иерархии. Подготовленное указание, включающее список исполнителей, после утверждения Руководителем (6, 7) поступает в систему документооборота. Вариант использования «Анализ документа» (2) расширяет функции (1) и (3) «Подготовка проекта указаний», которые используются на различных уровнях иерархии. Процессы (1) и (3) имеют различные нормативные наборы данных, поэтому на схеме показаны дважды для демонстрации функциональных различий.

Разработка алгоритма выполнения функции

Проведенная структуризация функции информационной поддержки подготовки проекта указаний по исполнению решений позволила выделить следующие задачи, необходимые в процессе разработки алгоритма:

- подготовка текста документа, создание карты анализа;
- анализ текста поступившего документа на совпадение с терминами нормативов о распределении функциональных обязанностей по подразделениям;
- сохранение данных о проведенном анализе для получения объяснений по каждому назначенному исполнителю, что немаловажно при распределении работ в организации;
- корректировка проекта указаний на основе личного опыта Советника по проблемам, связанным с выполнением предыдущих аналогичных заданий (анalogии можно выбирать по тематическим направлениям с использованием синонимов и ассоциаций);
- перенос подготовленного проекта в карточку документа с сохранением результатов анализа до окончания исполнения документа.

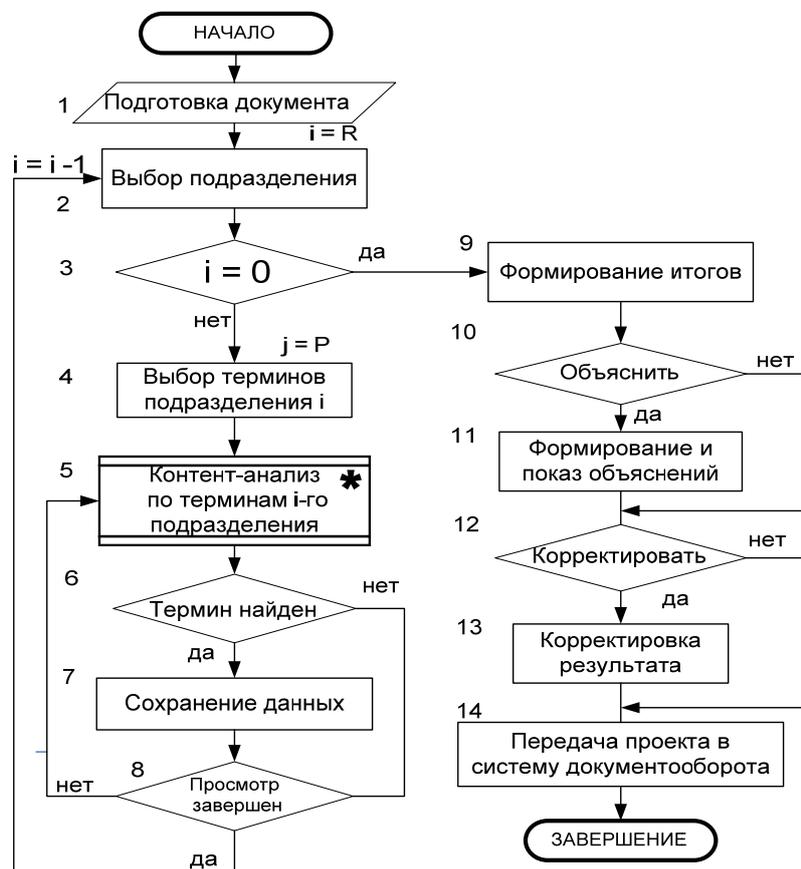


Рисунок 2. Блок-схема алгоритма подготовки проектов указаний

Анализ текста поступившего документа на соответствие со справочником должностных полномочий проводится по информационному массиву, представленному в виде тезауруса, включающего парадигматические отношения (синонимы, вышестоящие, нижестоящие). При совпадении терминов подсчитывается количество совпадений по тематике подразделения и вносится отметка о совпадении в карту представления документа. Количество совпадений с тематическими направлениями подразделения выдается как результат работы алгоритма и является основанием для формирования списка

исполнителей. Список исполнителей сортируется в порядке убывания значений, ограничивается по количеству подразделений. Блок – схема алгоритма представлена на Рисунке 2.

В результате работы алгоритма выдается список исполнителей, содержащий первые 4 элемента (названия подразделений), показывается количество совпадений. Блок формирования пояснений (11), при выборе подразделения выделяет в тексте все совпадения со справочником должностных полномочий по данному подразделению. Предусмотрены функции добавления текста в проект указания и корректировки исполнителей. Работа алгоритма завершается передачей подготовленного проекта в систему документооборота.

Результат разработки

Разработанный программный модуль осуществляет поиск фрагментов текстового массива на совпадение с терминами расширенного классификатора функциональных обязанностей (тезауруса) до ближайшего подчиненного уровня должностной иерархии. Найденное количество фрагментов текста подсчитывается и вычисляется суммарный весовой коэффициент по отношению к конкретному исполнителю. Результаты подсчета ранжируются и выдаются как проект назначения исполнителей. Ответственным исполнителем алгоритм указывает исполнителя с большим весовым коэффициентом. Режим редактирования проекта позволяет изменить ответственного исполнителя [Баканова, 2011].

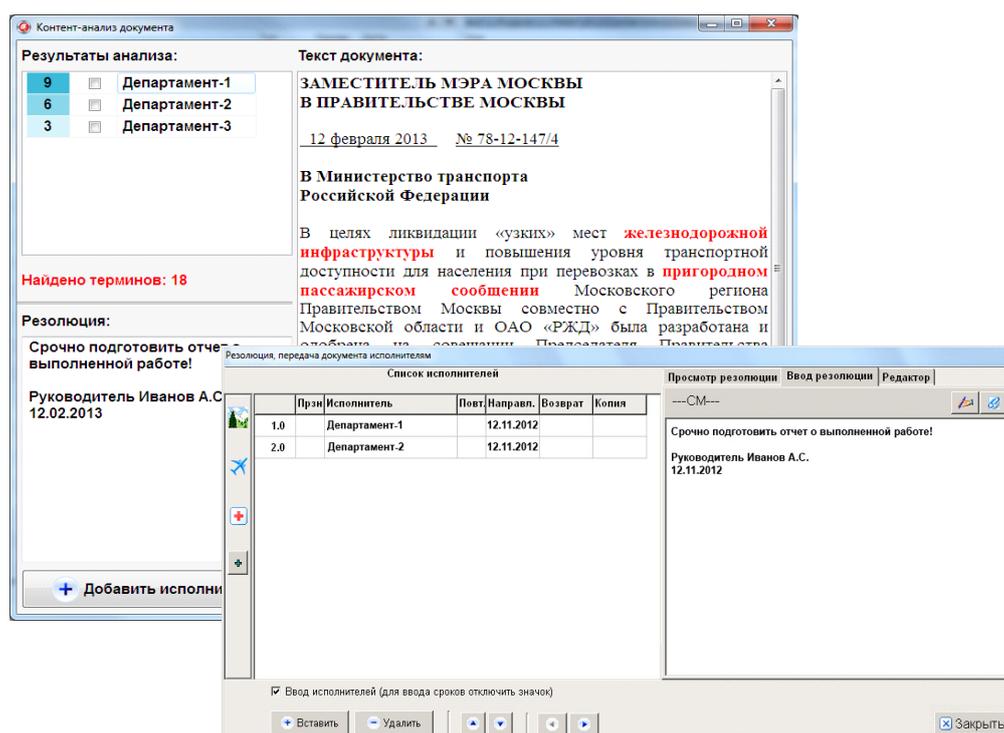


Рисунок 3. Функции интерфейса подготовки проекта указаний

На Рисунке 3 показаны функции интерфейса подготовки проекта указаний. Результаты анализа представлены в виде списка исполнителей, для каждого из которых указано количество терминов, найденных в тексте документа. Чем больше терминов найдено в тексте для данного исполнителя, тем более темным фоном отмечено их количество. В нижней части экрана указано общее число найденных в тексте документа терминов по всем исполнителям. Программный модуль позволяет получить объяснение результатов (весовых коэффициентов), выделяя в тексте выбранные термины цветом и показывая результат подсчета встречаемости терминов. Разработка интерфейса проводилась с учетом

исследования ментальных репрезентаций при подготовке проектов управленческих решений [Баканов, 2009].

Заключение

Исследование процедур поддержки управленческой деятельности и информационного потенциала баз данных позволяет создавать методы и алгоритмы, для выполнения работ по совершенствованию управленческой деятельности [Петровский, 2009].

Выполненная разработка показала возможности прагматического анализа текста деловых документов, поступающих в организацию.

Благодарности

Работа выполнена при поддержке гранта РФФИ 14-07-00216_а «Разработка принципов реализации информационной поддержки принятия управленческих решений, на базе информационного потенциала систем организационного управления».

The paper is published with financial support by the project ITHEA of the Institute of Information Theories and Applications FOI ITHEA (www.ithea.org) and the Association of Developers and Users of Intelligent Systems ADUIS Ukraine (www.aduis.com.ua).

Литература

[Баканов, 2009] Баканов А.С. Особенности психологического подхода к моделированию человеко-компьютерного взаимодействия // Вестник ГУУ. 2009. №6. С. 15–1

[Баканова, 2011] Баканова Н.Б. Интеграция систем организационного управления и интеллектуальных сервисов поддержки принятия решений – журнал «Искусственный интеллект и принятие решений», 2011 № 3, стр. 17-25 ISSN:2071- 8594

[Гради Буч, 1999] Гради Буч. Объектно-ориентированный анализ и проектирование, второе издание - М.: Издательство «Бином», СПб., «Невский диалект», 1999 г.

[Ларин, 2002] Ларин М.В. Управление документацией в организациях. – М.: Научная книга. 2002.

[Петровский, 2009] Петровский А.Б. Теория принятия решений. – М.: Издательский центр «Академия», 2009.

Сведения об авторах



Баканова Нина Борисовна – к.т.н., доцент, Институт прикладной математики им. М.В. Келдыша РАН, Миусская пл., д.4, Москва, 125047, Россия; e-mail: nina@keldysh.ru

Information Support of Preparation Project Management Decisions on Organizational Performance

Nina Bakanova

Abstract: *This article describes the design features of informational support project preparation of management decisions on the execution of documents input. Development of functions based on the method of content analysis of the document into coincidence with the terms of office directory distribution organization.*

Keywords: *Docflow system, content analyses*

ПСИХОЛОГИЧЕСКИЙ ПОДХОД К ИЗВЛЕЧЕНИЮ ЗНАНИЙ В ПРОЦЕССЕ ЧЕЛОВЕКО-КОМПЬЮТЕРНОГО ВЗАИМОДЕЙСТВИЯ С СИСТЕМОЙ ЭЛЕКТРОННОГО ДОКУМЕНТООБОРОТА

Арсений Баканов

Аннотация: В настоящей статье описывается способ извлечения и визуализации экспертных знаний в процессе человеко-компьютерного взаимодействия с системами электронного документооборота, системами документационного обеспечения управления. В работе приведено описание проведенных экспериментов, приводятся полученные результаты экспериментов. Описываемый подход к извлечению знаний, ориентирован на использование в системах документационного обеспечения управления и предполагает использование алгоритмов и правил для выявления эвристик в процессе принятия решений при решении управленческих задач.

Ключевые слова: Системы электронного документооборота, Извлечение экспертных знаний, Представление знаний.

Классификация ключевых слов ACM: H.4.2 Types of Systems - Decision support

Введение

Статья посвящена исследованию проблем, связанных с взаимодействием человека с информационными системами, в частности вопросам извлечения и визуализации экспертных знаний в процессе взаимодействия с системами документационного обеспечения управления. Важность и актуальность проводимых исследований обусловлена рядом причин, среди которых в первую очередь необходимо отметить востребованность в связи, с созданием в РФ электронного правительства и переходу к безбумажному документообороту в Федеральных органах исполнительной власти РФ. В настоящей статье описывается оригинальная методика неявного извлечения экспертных знаний в процессе человеко-компьютерного взаимодействия. Методика ориентирована на использование в системах документационного обеспечения управления и предполагает разработку алгоритмов и правил для выявления эвристик в процессе принятия решений при решении управленческих задач. Также в данной статье предлагается способ визуализации извлеченных экспертных знаний, позволяющий представить извлеченные знания как структурированную совокупность символов, образов и массивов текстовой информации.

В настоящей статье, кратко представлены результаты исследований, проводимых с 2012 года по настоящее время в Институте психологии РАН. Проводились исследования двух типов: в исследованиях первого типа первоначально предъявлялся стимульный материал (в виде вопросов, ключевых слов, графических образов), а затем испытуемому предъявлялась текстовая информация, после прочтения которой, испытуемый принимал решение о сортировке (т. е. определял департамент или подразделение организации, куда документ должен быть направлен для исполнения), отвечал на вопросы по тексту и проходил психологическое тестирование; в исследованиях второго типа, эксперимент начинался с

предъявления текстовой информации, после прочтения которой, испытуемый принимал решение о сортировке, отвечал на вопросы по тексту и проходил психологическое тестирование.

Испытуемым предъявлялись тексты, как прошедшие предварительную обработку – т.е. с выделенными опорными словами и визуализированной структурой, так и тексты без предварительной обработки.

В ходе исследований, использовалась установка контроля движения взора испытуемого (www.smivision.com). С ее помощью были получены экспериментальные данные: траектория взора, диаметр зрачка испытуемого, скорость перемещения взора, как в процессе чтения текста, так и в процессе принятия решения.

Системы электронного документооборота

Основной задачей системы электронного документооборота (СЭД) — является обеспечение процесса создания, управления и распространения электронных документов. Первоначально системы электронного документооборота использовались исключительно для учета и контроля прохождения документов, однако со временем к СЭД стали предъявляться все новые и новые требования, в связи с этим СЭД стали охватывать все более широкий спектр задач. В настоящее время СЭД фактически представляют собой системы организационного управления.

Информационные системы организационного управления в процессе функционирования накапливают большие информационные массивы [Баканова, 2007], сохраняемые в виде баз данных. В информационных массивах содержится информация, отражающая различные аспекты деятельности организации. Одним из таких аспектов деятельности организации является процесс принятия управленческих решений. Поэтому расширение возможностей использования информационного фонда систем организационного управления, за счет создания информационной поддержки принятия решений, является одной из актуальных задач повышения эффективности СЭД.

Обобщение и анализ информации, относящейся к различным прикладным задачам, открывает более широкие возможности ее использования. Исследование этих возможностей для целей, связанных с реализацией поддержки принятия решений в СЭД, представляет собой интересные малоизученные задачи. Одной из таких задач является выявление информационных источников для реализации сервисов поддержки управленческой деятельности. Ее решение позволит перейти к определению сервисов, необходимых для поддержки управленческой деятельности [Баканова, 2007]. Достоинством созданных сервисов будет оперативная и актуальная информация, полученная из выявленных источников данных. Анализ дополнительных возможностей использования накопленного информационного фонда и создание на его основе сервисов поддержки управленческой деятельности являются важной составной частью задачи повышения эффективности использования СЭД, а также качества управления в организационных структурах.

Для решения указанной задачи, при проектировании СЭД требуется выявить потенциал прикладных подсистем в части накопления данных, полезных для использования в управленческой деятельности. При этом, возможно, потребуется расширить функциональные задачи подсистем [Ташев и др., 2013] для формирования и сохранения информации, которая может быть использована в процессах принятия решений. Использование прикладных (технологических) информационных подсистем в качестве источника данных для организации поддержки принятия управленческих решений определяет новые подходы к разработке СЭД с целью создания востребованных сервисов поддержки управленческой деятельности. Функционал сервисов базируется на алгоритмах, обеспечивающих лицо, принимающее решение (ЛПР), оперативно подготовленной информацией, а также на методах выявления и

идентификации требуемых данных, содержащихся в программных комплексах систем организационного управления. Для эффективного функционирования системы поддержки принятия решений (ППР) требуется знание предметной области [Luger, 2002]. В целях успешной реализации системы, было принято решение ограничить предметную область транспортной тематикой. Такое решение было принято по примеру разработчиков создавших приложение SHRDLU [Winograd, 1972] работающее в минимальной предметной области.

Взаимодействие человека с системами электронного документооборота

В организационном управлении субъектом управления всегда выступает человек. Для принятия управленческих решений, направленных на выполнение стратегических и тактических задач, поставленных и оперативно возникающих перед организацией, руководитель постоянно должен принимать и отвергать сотни разнообразных альтернатив [Баканова, 2007]. Управленческое решение – это некоторое воздействие на управляемый объект, определяющее действия, необходимые для проведения изменений в его состоянии.

Проблемам принятия решений посвящено множество научных работ, среди которых можно отметить работы таких ученых как Luger, Newell, Simon, Ларичев, Петровский. Среди исследований, касающихся вопросов информационного взаимодействия человека с техникой вообще, необходимо отметить работу коллектива авторов: Т. В. Атанасовой, Т.Н. Савченко, Г.М. Головиной и др., в которой описаны психологические механизмы взаимодействия человека с интеллектуальной информационной средой обитания [Атанасова Т.В. и др., 2010]. С позиций когнитивной психологии [Величковский Б.М., 2006], возможности человека по приему и переработке информации описываются с помощью различных функциональных моделей структуры памяти пользователя [Baddeley, 2009].

В процессе взаимодействия с информационной средой, к человеку по различным информационным каналам (с использованием различных телекоммуникационных устройств) могут поступать значительные объемы информации для переработки и последующего принятия решения. Количество каналов информации, по которым информация поступает к человеку, продолжает стремительно увеличиваться, как и количество поступающей информации. В процессе взаимодействия с СЭД, человеку приходится учитывать значительное количество различных факторов, а также решать задачи многокритериального выбора. Для человеческой системы переработки информации многокритериальные задачи представляют собой особо сложный класс задач [Петровский А.Б., 2004]. Наличие многих критериев приводит к нагрузке на человеческую систему переработки информации, заставляя человека использовать различные, зачастую оригинальные эвристики для того, чтобы решить поставленную задачу [Ларичев О. И., 1987].

Описание проведенных экспериментов

В соответствии с целью работы было проведено экспериментальное исследование взаимодействия человека с интеллектуальной информационной средой на примере взаимодействия человека с системой поддержки принятия решений. В ходе исследований моделировалась работа лица принимающего решение. Работа ЛПР заключалась в ознакомлении с документом – чтении текста предъявленного на мониторе компьютера, с последующим принятием решения о сортировке (decision making), т. е. определении департамента или подразделения организации, куда документ должен быть направлен для дальнейшей обработки.

Необходимо отметить, что в крупные управленческие организации ежедневно поступают сотни документов и ошибка ЛПР при сортировке чревата не только проволочками при исполнении документа, но и ненадлежащим исполнением, в случае если документ будет направлен в непрофильное подразделение. Для работы с документами (в том числе входящими), в крупных управленческих организациях традиционно используют системы электронного документооборота [Баканова Н. Б., 2007]. В состав такой системы электронного документооборота может быть включен программный модуль, реализующий функции системы поддержки принятия решений. Данный модуль осуществляет предварительную обработку текста документа, выделяет опорные слова цветом и визуализирует структуру документа. В рамках проводимых экспериментов представляло научный интерес исследовать процесс принятия решений в задаче о сортировке документа, как «экспертами», т. е. людьми, имеющими опыт работы в качестве ЛПР, так и «новичками», т. е. людьми, не имеющими опыта работы в качестве ЛПР. В процессе исследования, решения принимались как с помощью системы поддержки принятия решений, так и без помощи системы поддержки принятия решений. В качестве испытуемых в исследованиях участвовали:

- 1) Эксперты – лица имеющие опыт работы в качестве ЛПР.
- 2) Новички – лица, не имеющие опыта работы в качестве ЛПР – студенты высших учебных заведений г. Москвы.

Исследования проводились с использованием оборудования отслеживающего траекторию движения глаз пользователя в процессе считывания информации, а также в процессе принятия решений (www.smivision.com). Параллельно с проведением экспериментов, проводились исследования когнитивного стиля импульсивный/рефлексивный [Холодная, 2002], а также стиля саморегуляции поведения (ССПМ) по методике В. И. Моросановой [Моросанова, Индина, 2011]. Испытуемым предъявлялся некоторый документ содержащий текст, после прочтения, которого необходимо было принять решение о выборе из некоторого числа альтернатив (принять решение о сортировке документа). Испытуемым последовательно предъявлялись тексты как прошедшие предварительную обработку системой поддержки принятия решений, так и без обработки. Обработка заключалась в определении структуры текста и проведении контент-анализа с использованием специально разработанного словаря (тезауруса). В результате обработки визуализировалась структура текста, а опорные слова выделялись цветом [Баканов А.С., 2009]. Последовательность предъявления текстов (с обработкой и без обработки) на мониторе компьютера, менялась, чтобы нивелировать привыкание испытуемого, после прочтения текста испытуемый принимал решение. Экспериментатор наблюдал и фиксировал (в том числе и в формате *.mpeg) траекторию движения глаз испытуемого, как в процессе чтения текста, так и в процессе принятия решения. На этапе принятия решения испытуемым предъявлялись (также на мониторе компьютера) вопросы двух типов. К вопросам первого типа относились вопросы, предъявляемые в следующем порядке:

- 1) Выберите альтернативы (из некоторого списка) к которым относится данный документ/текст.
- 2) Выберите альтернативы (из числа выбранных ранее) к которым данный документ/текст относится в наибольшей степени.
- 3) Выберите только одну альтернативу (из выбранных ранее).

Таким образом, в процессе ответов на вопросы первого типа в приведенном выше порядке выявлялась структура ментальных репрезентаций пользователя, сформировавшаяся по прочтении конкретного документа/текста см. Рисунок 1.

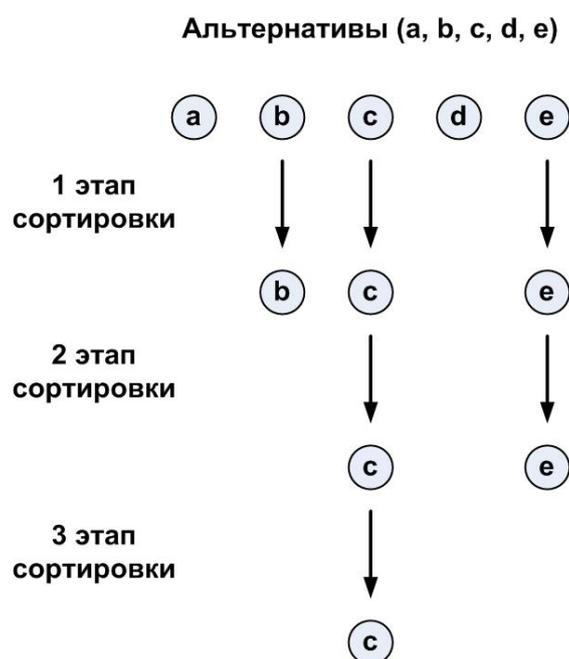


Рис. 1

Необходимо отметить, что существует достаточное количество определений термина ментальная репрезентация, мы же будем использовать следующее определение: ментальная репрезентация - субъективный образ объективной реальности, отражение внутреннего и внешнего мира в сознании человека. Или применительно к данному исследованию субъективный образ прочтенного документа/текста в сознании испытуемого. При ответе на вопрос (также предъявляемый на экране монитора) второго типа испытуемый должен был количественно оценить степень, в которой данный документ/текст относится к выбранной альтернативе. Траектория движения взора, скорость движения взора, а также диаметр зрачка испытуемого в процессе принятия решения фиксировалась экспериментатором.

Кроме траектории взора, экспериментатор в процессе проводимого исследования фиксировал время в течении которого испытуемый читал предъявленный текст, а также время в течении которого испытуемый принимал решение после прочтения текста как с выделенными цветом (опорными) словами, так и без выделенных цветом опорных слов. В настоящее время исследования продолжаются. К настоящему времени в исследованиях приняло участие более 70 человек.

В ходе проведения экспериментов удалось выявить, что большинство испытуемых читали предлагаемый текст дважды. Причем во второй раз взор испытуемого перемещался от одного опорного слова (фрагмента текста) к другому опорному слову (фрагменту текста) в независимости были ли они выделены в процессе обработки текста или нет. В процессе движения от одного опорного слова к другому, взор испытуемого „перескакивал” через строки и абзацы, иногда возвращаясь к отдельным фрагментам текста. Можно предположить, что таким образом испытуемый пытался запомнить наиболее важные (для него) слова или фрагменты текста. Такие последовательности слов можно представить в виде:

$$A(X_1, X_2, X_3, \dots, X_n)$$

$$B(Y_1, Y_2, Y_3, \dots, Y_m)$$

и т.д.

где $X_1, X_2, X_3, \dots, X_n, Y_1, Y_2, Y_3, \dots, Y_m$ опорные слова (невыделенные либо выделенные в процессе обработки) и фрагменты текста; A, B – некоторые предикаты.

Последовательность опорных слов также можно представить в виде:

$$A_1(X_1, X_2), A_2(X_2, X_3), \dots, A_{n-1}(X_{n-1}, X_n)$$

$$B_1(X_1, X_2), B_2(X_2, X_3), \dots, B_{n-1}(X_{n-1}, X_n)$$

и т.д.

Представляя предикатное описание графически, получаем последовательность опорных слов в виде семантической сети рис.2.

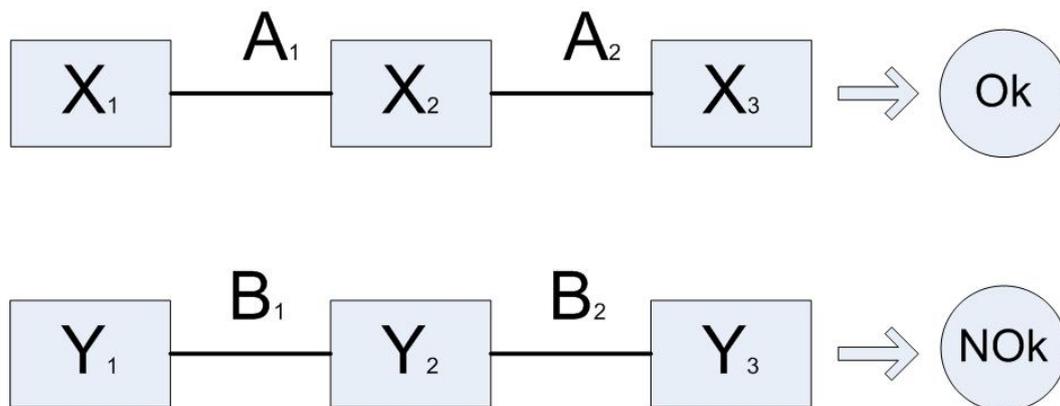


Рис. 2

Символами «Ok» и «NOK» на Рисунок 2 показано решение эксперта задачи о сортировке, т.е. решение о том, что данный документ/текст относится (или не относится) к выбранной альтернативе.

Краткие выводы

Обобщая все вышеизложенное можно сделать следующие выводы:

- 1) Представляя отношения дугами графа, мы визуализируем алгоритм рассуждения эксперта о предметной области.
- 2) Двигаясь по дугам графа мы приходим к результату – решению эксперта в задаче о сортировке.
- 3) Расширяя множество $X_1, X_2, X_3, \dots, X_n, Y_1, Y_2, Y_3, \dots, Y_m$ опорных слов с помощью специализированного словаря мы можем получить совокупность продукционных правил „ЕСЛИ-ТО”.

Благодарности

Автор благодарен доценту к.ф.н. Леандру Литову, Заведующему секцией Атомная Физика, Физический Факультет, Софийский Университет „Св. Климент Охридски”, Болгария, за предоставленный доступ к компьютерным ресурсам грид-структуры CERN.

Библиография

- [Baddeley, 2009] Baddeley, A.D., Eysenck, M., Anderson, M.C. Memory. Hove: Psychology Press. 2009.
- [Luger, 2002] George F. Luger. Artificial Intelligence. Structures and Strategies for Complex Problem Solving, 4th Edition, Addison Wesley. 2002.
- [Winograd, 1972] Understanding Natural Languages. New York: Academic Press, 1972
- [Атанасова и др., 2010] Атанасова Т., Савченко Т.Н., Головина Г.М., Баканов А.С. Интеллектуальная информационная среда обитания и субъективное восприятие качества жизни // Методы исследования психологических структур и их динамики. Труды ИП РАН. М., 2010.
- [Баканов, 2009] Баканов А.С. Особенности психологического подхода к моделированию человеко-компьютерного взаимодействия // Вестник ГУУ. 2009. №6. С. 15–18.
- [Баканова, 2007] Баканова Н. Б. Использование программно-технических комплексов для повышения эффективности контроля в системах документооборота // «Электросвязь». 2007. № 6. С. 51–53.
- [Величковский, 2006] Величковский Б.М. Когнитивная наука: Основы психологии познания. В 2 т. Т. 1. М.: Смысл, 2006.
- [Ларичев, 1987] Ларичев О.И., Петровский А.Б. Системы поддержки принятия решений. Современное состояние и перспективы развития. // Итоги науки и техники. Серия Техническая кибернетика. М. ВИНТИ, 1987. т.21, с.131-164.
- [Моросанова, Индина, 2011] Моросанова В.И., Индина Т.А. Регуляторные и личностные основы принятия решений. – СПб.; М.: Нестор-История, 2011, - 282 с.
- [Петровский, 2004] Петровский А.Б. Многокритериальное принятие решений по противоречивыми данным: подход теории мультимножеств. // Информационные технологии и вычислительные системы, 2004, №2, 56-66.
- [Ташев и др., 2013] Ташев Т., Баканова Н., Ташева Р. Исследование верхней границы пропускной способности коммутационного узла при входящем трафике типа "горячей точки". International Journal "Information Technologies & Knowledge", Vol.7, No. 2, 2013, pp. 182-189.
- [Холодная, 2002] Холодная М. А. Когнитивные стили: О природе индивидуального ума. Учебное пособие – М.: ПЕР СЭ, 2002. -304 с.

Информация об авторах



Баканов Арсений Сергеевич – к.т.н., Институт психологии Российской академии наук, Москва, Россия; e-mail: arsb2000@pochta.ru

Основные области научных исследований: Инженерная психология, Системы принятия решений, Компьютерные технологии

Psychological Approach to Extracting Knowledge in Human-Computer Interaction with Electronic Docflow System

Arseniy Bakanov

Abstract: This article describes the method of extraction and visualization of knowledge in the process of human-computer interaction with electronic docflow system. The paper describes the experiments, the results of experiments are given. The described approach to extracting knowledge, focused on the use of docflow systems, and involves the use of algorithms and rules to identify heuristics in decision-making in solving management problems.

Keywords: Docflow system, Extracting knowledge, Visualization knowledge.

ДВУХУРОВНЕВЫЙ ГЕНЕТИЧЕСКИЙ АЛГОРИТМ РЕКОНФИГУРАЦИИ ПРОГРАММИРУЕМЫХ ЛОГИЧЕСКИХ ИНТЕГРАЛЬНЫХ СХЕМ

Алексей Городилов

Аннотация: В статье рассматривается задача реконфигурации программируемой логической интегральной схемы после возникновения отказов логических элементов. Задача является актуальной для таких областей, как космонавтика и управление ответственными промышленными объектами, в которых используются высоконадежные отказоустойчивые системы. В статье рассматриваются отказы логических элементов в конфигурируемых логических блоках. Для оптимальной реконфигурации необходимо разместить логические модули на программируемой логической интегральной схеме, не используя отказавшие логические элементы, при условии минимизации занимаемой площади, длины связей и величины задержек. Первым этапом в решении поставленной задачи является выбор компактно расположенных логических элементов, которые будут задействованы в реализации схемы. Компактность здесь понимается как минимальность количества информации, необходимой для описания используемого множества логических элементов. Поставленная задача не может быть решена точно и эффективно, но может быть решена приближенно с помощью различных эвристических методов, в том числе генетических алгоритмов. В статье описывается несколько известных подходов к решению задачи и предлагается оригинальный двухуровневый способ кодирования для построения генетического алгоритма, учитывающего особенности задачи и использующего стандартные генетические операторы. Предложенный алгоритм реализован на практике в виде компьютерной программы, позволяющей сравнивать его работу с некоторыми другими алгоритмами. Выполнено сравнение различных эвристических алгоритмов с точки зрения точности находимого решения, оценивается скорость работы предложенного алгоритма. Можно сказать, что предложенный генетический алгоритм в среднем работает дольше других эвристических алгоритмов, но находит более точное решение, что ведет к уменьшению времени реконфигурации и, тем самым, к увеличению коэффициента готовности программируемой логической интегральной схемы.

Ключевые слова: программируемые логические интегральные схемы, реконфигурация, надежность, генетический алгоритм.

ACM Classification Keywords: I.2 Artificial Intelligence: I.2.8 Problem Solving, Control Methods, and Search - Heuristic methods, B.8 Performance and Reliability: B.8.1 Reliability, Testing, and Fault-Tolerance

Введение

Современные цифровые устройства содержат множество транзисторов и линий соединений, отказ которых может привести к нарушению работы всего устройства, поэтому для обеспечения бесперебойной работы оборудования в высоконадежных системах (например, в области космонавтики и управления ответственными промышленными объектами) требуются соответствующие методы проектирования, диагностирования и восстановления. Для решения задачи восстановления могут быть использованы программируемые логические интегральные схемы (ПЛИС), которые предоставляют широкие

возможности логической реконфигурации. ПЛИС являются универсальным базисом для проектирования цифровых устройств любого уровня сложности. По статистическим данным ведущих фирм-производителей объем производства ПЛИС постоянно увеличивается [Уваров, 2007].

При реконфигурации основной задачей является оптимальное выполнение стадий размещения и соединения с учетом частичной или полной неработоспособности отказавших элементов. Оптимальность здесь понимается в смысле минимизации занятой площади, длины соединений и величины задержек. Для решения сформулированной задачи оптимизации нами предлагается использовать генетический алгоритм.

Задача реконфигурации

Каждый производитель ПЛИС имеет свою собственную архитектуру, но в целом все они являются вариациями архитектуры, показанной на рис. 1 [Zeidman, 2002].

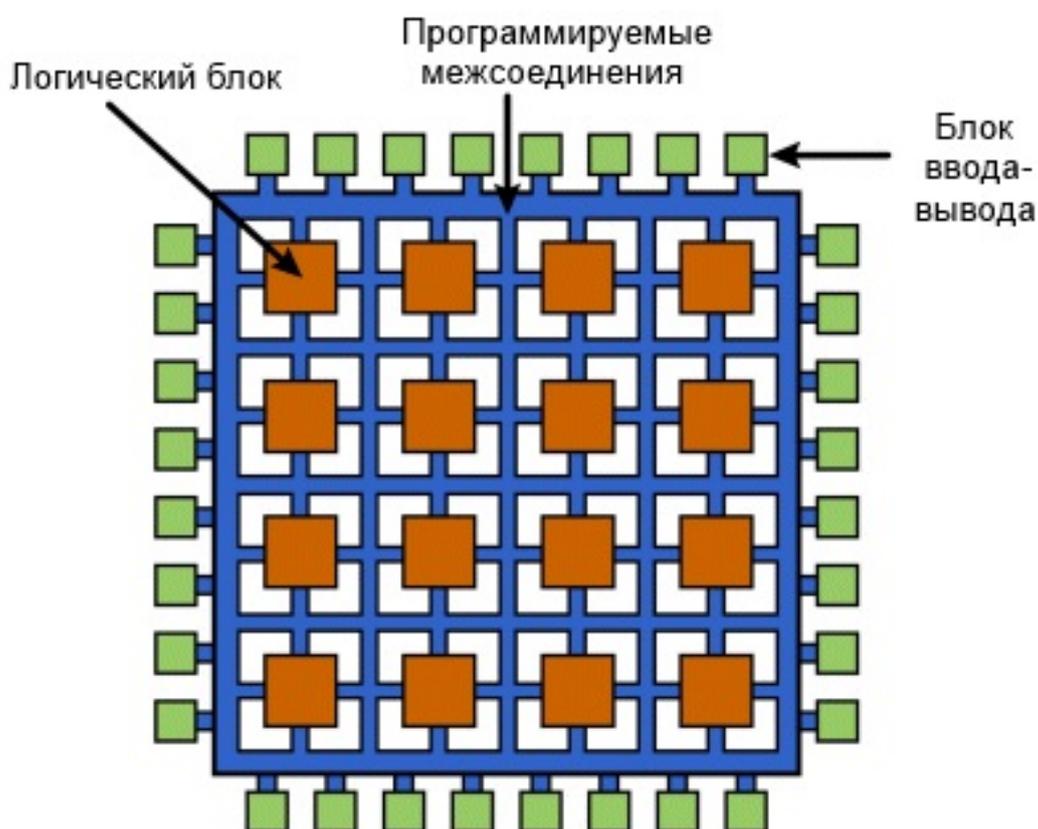


Рис. 1. Общая схема ПЛИС типа FPGA

Основными составляющими в ПЛИС типа FPGA являются ячейки конфигурационной памяти, конфигурируемые логические блоки (КЛБ), блоки ввода-вывода и программируемые межсоединения.

В данной статье рассматриваются отказы логических элементов (ЛЭ) в КЛБ. Некоторые виды отказов могут приводить к полному выходу из строя элемента и невозможности его дальнейшего использования. Такие элементы должны быть исключены и в результате реконфигурации их функции будут реализованы другими элементами. Однако, некоторые отказы приводят к тому, что элемент сохраняет способность

выполнять какую-либо полезную работу. Например, к таким видам отказов относятся однократные константные отказы для некоторых видов логических элементов: элементов с избыточным базисом [Турун, 1999, Турун, 2011], 8-1 мультиплексоров и таблиц перекодировки (LUT) [Турун, 2013]. В случае такого отказа КЛБ может быть задействован в схеме, но с учетом его ограниченной функциональности.

Согласно [Цыбин, 2006], большая доля отказов приходится на ячейки памяти. Однако, такие отказы тоже можно рассматривать как однократные константные отказы на входе логических элементов: если ячейка памяти вышла из строя, она генерирует на выходе какое-либо постоянное значение (0 или 1). Отказы в соединительных линиях также могут быть описаны константными отказами в логических элементах (обрыв соединения означает константу 0, замыкание на питание – константу 1).

По результатам диагностирования должна быть получена информация о состоянии каждого логического элемента. Поскольку элементы в ПЛИС образуют строго упорядоченную матричную структуру (см. рис. 1), эти данные можно представить в виде таблицы, значения которой обозначают состояние соответствующих логических элементов (отказов нет, полностью неработоспособен, либо частично работоспособен, с указанием его ограниченных возможностей). Такую таблицу будем называть таблицей состояний. Задача реконфигурации состоит в поиске новой схемы размещения и соединения ЛЭ, для чего, в первую очередь, нужно выбрать из доступного множества элементов те, которые будут использованы в схеме.

Нужно отметить, что традиционная архитектура обычно содержит систему локальных соединений нижнего уровня и глобальные линии связи, которые в совокупности обеспечивают возможность установления связи между любой парой логических элементов. Однако для снижения затрат конфигурационной памяти в FPGA может быть реализована трехуровневая структура, в которой прямая связь между некоторыми удаленными элементами может быть недоступна [Цыбин, 2006]. Кроме того, соединение между логическими элементами имеет характеристики (например, количество межсоединительных переключателей в цепи, влияющее на величину задержки сигнала), значение которых нужно минимизировать. В общем случае можно считать, что более предпочтительным является такой выбор ЛЭ, при котором они будут располагаться наиболее компактно.

Множество используемых ЛЭ в системах конфигурации ПЛИС описывается набором прямоугольников (координатами их левого верхнего угла и размерами). Прямоугольники не должны пересекаться и не должны содержать неработоспособных элементов. Чем меньше количество таких прямоугольников в описании, тем короче будет конфигурационный файл, и тем быстрее будет выполнена реконфигурация. Кроме того, элементы внутри одного прямоугольника располагаются максимально компактно, поэтому минимизация количества прямоугольников ведет и к оптимизации расположения элементов, то есть желаемой минимизации характеристик соединений.

Рассмотрим упрощенный случай, когда каждый элемент может находиться только в одном из двух состояний: полностью работоспособном (будет обозначаться «1» в таблице состояний) или полностью неработоспособном (будет обозначаться «0»). Назовем прямоугольный фрагмент таблицы состояний, полностью состоящий из «1» покрывающим прямоугольником. Пусть N – количество ЛЭ, необходимых для реализации схемы устройства (или, по крайней мере, той части устройства, которая обеспечивает выполнение основных и наиболее важных функций). Тогда первый этап в решении задачи реконфигурации заключается в поиске для заданной таблицы состояний минимального числа покрывающих прямоугольников, содержащих в сумме не менее N элементов. Поскольку количество логических элементов в FPGA может быть очень велико, пространство поиска получается большим.

Подходы к решению

Тривиальный подход к решению задачи заключается в переборе различных наборов покрывающих прямоугольников. Однако, общее их количество в худшем случае пропорционально числу элементов в матрице. Таким образом, количество наборов, то есть перебираемых вариантов, будет расти экспоненциально с ростом числа элементов. Следовательно, тривиальный алгоритм является неэффективным и не может использоваться на реальных данных.

Поскольку точных эффективных алгоритмов решения поставленной задачи не известно, на практике применяют различные эвристические алгоритмы. Одним из них является следующий «жадный» алгоритм:

1. Установить общее количество искомых элементов: $Need \leftarrow N$.
2. Найти покрывающий прямоугольник R , содержащий наибольшее количество «1» N_R .
3. Установить все элементы таблицы в прямоугольнике R в значение «0».
4. Уменьшить количество искомых элементов: $Need \leftarrow Need - N_R$.
5. Если найденных элементов не достаточно ($Need > 0$), перейти к шагу 2.

Данный алгоритм работает быстро, но не всегда находит точное решение, то есть не всегда найденное количество покрывающих прямоугольников оказывается минимальным.

Для решения указанной проблемы в данной статье предлагается использовать генетические алгоритмы, также являющиеся разновидностью эвристических алгоритмов. Их использование в рассматриваемой задаче представляется оправданным, поскольку генетические алгоритмы хорошо зарекомендовали себя для задач, где пространство поиска велико и не монотонно [Гладков, 2006].

Известны примеры применения генетических алгоритмов в задачах проектирования и диагностирования ПЛИС. Так, в [del Solar, 2006] ГА используются для задач размещения и связи. Однако, рассматриваемая в данной статье задача реконфигурации существенно отличается наличием неработоспособных элементов в матрице, что существенно увеличивает сложность задачи размещения. В [Yakimets, 2011] рассматриваются альтернативные подходы к разработке систем ПЛИС на основе ГА. Предлагается использовать несколько генотипов на одном (или нескольких) кристаллах ПЛИС и разрабатывать частично корректные детерминированные автоматы. Их использование уменьшает временные затраты на эволюцию, однако усложняет методику генерации проекта.

В данной статье предлагается доработка стандартного генетического алгоритма, которая позволит учесть дополнительные ограничения и повысить качество работы алгоритма. Отличительной особенностью предлагаемой доработки является новый способ кодирования хромосом.

Генетический алгоритм

Для разработки генетического алгоритма необходимо определить структуру хромосом и функцию приспособленности. Поскольку решением рассматриваемой задачи является набор покрывающих прямоугольников, тривиальным подходом является структура хромосом в виде последовательности дескрипторов прямоугольников. Однако, такой подход порождает множество проблем с генетическими операторами, поскольку хромосомы будут иметь различную длину в зависимости от количества прямоугольников в кодируемом ими решении. По этой причине мы будем использовать другой подход.

Для каждого элемента таблицы определим покрывающий прямоугольник максимальной площади, для которого данный элемент является левым верхним углом. Назовем такие покрывающие прямоугольники

локально-максимальными. Справедливо утверждение, что существует такая последовательность локально-максимальных прямоугольников, что их последовательный выбор приведет к точному решению задачи. Следовательно, решение может быть найдено в виде последовательности локально-максимальных прямоугольников, представленных своими дескрипторами. Таким образом, хромосома будет представлять собой перестановку на множестве дескрипторов локально-максимальных прямоугольников. При таком подходе длина каждой хромосомы L будет фиксирована и равна количеству локально-максимальных прямоугольников, что соответствует числу «1» в таблице состояний.

Однако, равная длина всех хромосом не решает всех проблем с генетическими операторами. Если считать отдельными генами хромосомы непосредственно элементы перестановки P , то есть i -ым геном особи считать число p_i , то гены получаются взаимозависимыми: если какой-то ген равен j , то никакой другой ген этой особи не может принимать значение j . Указанная интерпретация интуитивно понятна и не требует дополнительных вычислительных затрат, однако зависимость генов приводит к тому, что стандартные операторы оказываются неприменимы, так как все они работают с представлением набора генов в виде строки независимых бит.

Альтернативным подходом, предлагаемым в данной статье, является использование промежуточного представления особей в виде некоторого объекта, легко трансформируемого в перестановку. Такой объект должен задаваться при помощи битовой или числовой строки или матрицы, состоящей из независимых элементов.

Перестановке соответствует полное паросочетание в полном двудольном графе [Aho, 1987]. Пусть C – матрица размера $L \times L$. Элемент c_{ij} соответствует весу ребра между i -той вершиной первой доли и j -той вершиной второй доли в графе G . Зная такую матрицу, можно найти полное паросочетание минимального веса в графе G . Такая задача решается достаточно быстро за полиномиальное время [Асанов, 2001]. Упорядочим ребра паросочетания по номеру вершины первой доли. Тогда номера вершин второй доли образуют перестановку на множестве чисел $(1..L)$.

Таким образом, хромосомой в генетическом алгоритме является матрица C , а процесс кодирования является двухуровневым.

1. Строится двудольный граф с весами ребер, равными элементам матрицы C , в графе ищется полное паросочетание минимального веса.
2. Получившееся паросочетание определяет перестановку, которая задает порядок выбора локально-максимальных прямоугольников.

Приспособленность особи, как и окончательное решение задачи, определяются на основе алгоритма, аналогичного рассмотренному выше «жадному»:

1. Установить общее количество искомым элементов: $Need \leftarrow N$, количество используемых прямоугольников $N_{rect} \leftarrow 0$.
2. Взять очередной локально-максимальный прямоугольник R из перестановки, количество «1» в нем N_R .
3. Установить все элементы таблицы в прямоугольнике R в значение «0».
4. Уменьшить количество искомым элементов: $Need \leftarrow Need - N_R$, увеличить количество используемых прямоугольников $N_{rect} \leftarrow N_{rect} + 1$.
5. Если найденных элементов не достаточно ($Need > 0$), перейти к шагу 2.

Приспособленность особи будет равна количеству использованных прямоугольников N_{rect} .

Независимость элементов матрицы позволяет использовать стандартные генетические операции мутации и скрещивания. При селекции особей используются две основные классические идеи: «принцип рулетки» и «принцип элитизма» [Mitchell, 1999]. Размер популяции остается постоянным за счет удаления наименее приспособленных особей на каждом шаге. Условием окончания работы предлагаемого генетического алгоритма является достижения определенного поколения с номером M . Это позволяет заранее оценить время работы алгоритма или, наоборот, задать параметр M так, чтобы алгоритм завершил свою работу за указанное фиксированное время.

Предложенный алгоритм может быть легко адаптирован для общего случая, когда таблица состояний содержит и другие значения, соответствующие ЛЭ в частично работоспособном состоянии. В этом случае изменяется только этап определения локально-максимальных прямоугольников: элементы с несовместимым остаточным базисом [Турун, 2011] не могут попасть в один блок.

Реализация и пример работы

Для экспериментального сравнения различных алгоритмов была создана программная реализация приведенных выше «жадного» и генетического алгоритмов (рис. 2). Для генетического алгоритма предусмотрена возможность задания параметров, которые существенно влияют на качество находимого решения. К таким параметрам относятся

- численность популяции;
- вероятность мутации;
- процент лучших особей, переходящих в новое поколение без изменений;
- процент особей, принимающих участие в скрещивании в каждом поколении;
- количество поколений.

Значения параметров определяются экспериментально, поскольку общих теоретических рекомендаций по их подбору не существует. Очевидно, значение параметров численности популяции и количества поколений должно зависеть от объема исходных данных.

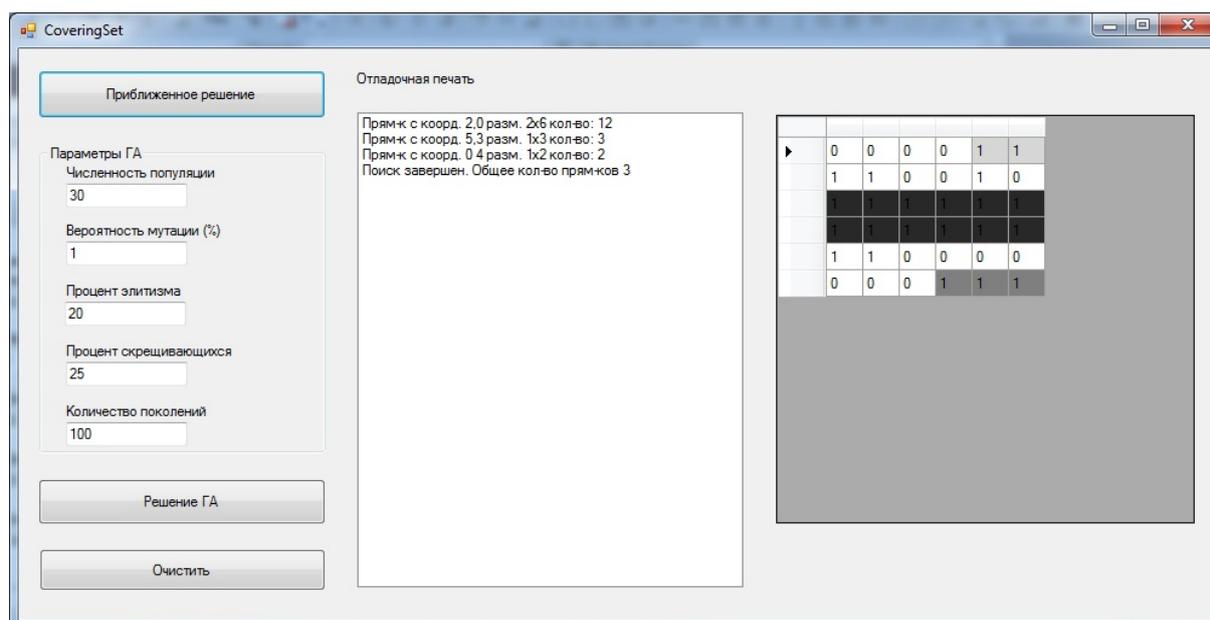


Рис. 2. Программная реализация алгоритмов. Приближенное решение

На рис. 2 в правом окне показано решение, которое находит «жадный» алгоритм (значения таблицы состояний можно видеть на рисунке, $N=16$). Данное решение, использующее 3 прямоугольника, не является оптимальным, поскольку 16 единиц можно покрыть двумя прямоугольниками, если не выделять целиком весь максимальный прямоугольник. Оптимальное решение можно увидеть на рис. 3.

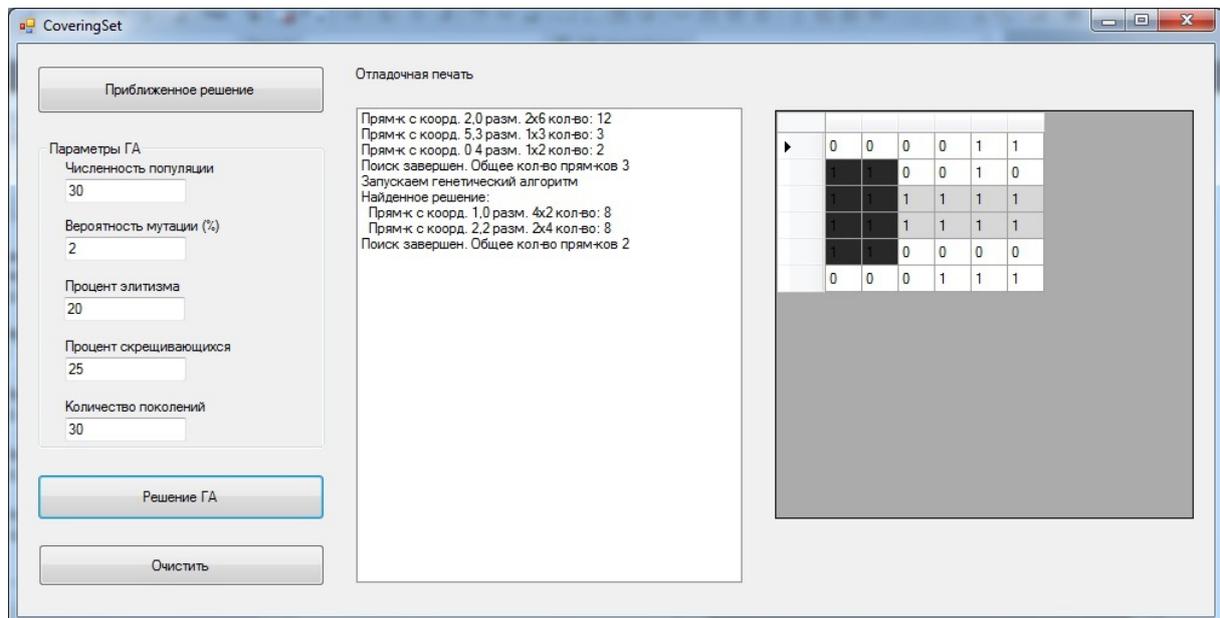


Рис. 3. Точное решение, найденное генетическим алгоритмом

Для генетического алгоритма был выполнен подбор оптимальных параметров, при которых алгоритм устойчиво находил оптимальное решение. Значения этих параметров, а также результат работы алгоритма можно увидеть на рис. 3.

Оценка эффективности

Время работы генетического алгоритма существенно зависит от значений параметров, поэтому сравнение его эффективности с другими алгоритмами затруднительно. Если подобрать параметры, при которых генетический алгоритм стабильно находит точное решение, то время его работы получается в несколько раз больше, чем время работы «жадного» алгоритма.

Выполним оценку количества операций, выполняемых генетическим алгоритмом, по сравнению с алгоритмом, осуществляющим полный перебор. Количество операций в обоих алгоритмах определяется количеством анализируемых решений. Рассмотрим случай таблицы размером 6x6, содержащей $q=22$ единиц (рис. 3). Для оценки количества операций в генетическом алгоритме будем использовать найденные значения параметров: численность популяции $N_{pop}=30$, количество поколений $N_{gen}=30$. Для полного перебора количество вариантов определяется формулой

$$op_{\text{перебора}} = 2^q = 2^{22} \sim 10^6$$

Для генетического алгоритма число анализируемых решений определяется количеством рассмотренных особей и может быть вычислено следующим образом

$$OP_{ГА} = N_{pop} * N_{gen} = 30 * 30 \sim 10^3$$

Таким образом, генетический алгоритм рассматривает существенно меньше решений, за счет чего работает принципиально быстрее, по-прежнему находя в большинстве случаев точное оптимальное решение.

Для оценки эффективности предложенного подхода кодирования особей, были рассмотрены несколько модельных задач на графах [Данилова, 2009]. Для всех задач двухуровневый генетический алгоритм чаще находил решение, более близкое к оптимальному. Например, для задачи коммивояжера ГА с модифицированным кодированием примерно в 1.5 раза чаще находит решение с наименьшей длиной цикла, что соответствует оптимальному решению. То есть, ГА с двухуровневым кодированием с большей вероятностью дает достаточно хороший результат, следовательно, позволяет найти некоторое приемлемое решение быстрее.

В задачах, где можно было найти точное решение, оценивался процент запусков ГА, в которых это точное решение было найдено. Результаты для задачи разбиения графа на максимально независимые подграфы можно увидеть в табл. 1. Из таблицы видно, что ГА с модифицированным кодированием находит точное решение в 1.3 – 2 раза чаще, чем ГА с классическим кодированием.

Табл. 1. Сравнение ГА со стандартным и модифицированным кодированием

№ теста		1	2	3	4	5	6	7	8	9	10
% запусков, в которых было найдено точное решение	классическое кодирование	23,2	41,2	46,6	18,8	35,8	10,8	14,2	9,2	10	22,6
	модифицир. кодирование	43,8	52,2	64,6	27,8	45,6	19,2	19,8	12,6	19,2	31

Приведенные результаты свидетельствуют об эффективности двухуровневого кодирования в смысле повышения качества решения, а также скорости нахождения приемлемого решения.

Заключение

Для рассмотренной задачи реконфигурации ПЛИС был разработан генетический алгоритм решения первого этапа задачи – поиска минимального количества покрывающих прямоугольников. Особенностью алгоритма является двухуровневый способ кодирования особей в ГА. Этот способ позволяет использовать стандартные генетические операторы при соблюдении ограничений задачи. Предложенное решение может быть распространено на случай частичных отказов, когда КЛБ сохраняет частичную работоспособность.

Для реализации алгоритма не требуется дополнительных ресурсов на ПЛИС. Предполагается, что алгоритм выполняется на внешнем универсальном вычислительном устройстве, обладающем достаточной вычислительной мощностью и подключаемом к плате ПЛИС для диагностики и реконфигурации. По этой причине при оценке приближенных алгоритмов их скорость работы менее значима, чем их точность. Анализируя эффективность предложенного генетического алгоритма можно сказать, что его скорость работы принципиально лучше скорости точных переборных алгоритмов, а в

сравнении с приближенными алгоритмами он работает в среднем медленнее, но позволяет найти более точное решение. Применение предложенного двухуровневого кодирования на модельных задачах оптимизации показывает, что генетический алгоритм, использующий стандартные операторы, позволяет найти решение, более близкое к оптимальному, в 70-80% случаев (для задачи поиска кратчайшего цикла).

Таким образом, предложенный алгоритм позволяет увеличить точность решения задачи размещения, что ведет к уменьшению времени реконфигурации и, тем самым, к увеличению коэффициента готовности ПЛИС.

Литература

- [Aho, 1987] A.V. Aho, J.E. Hopcroft, J.D. Ullman, Data Structures and Algorithms, Addison-Wesley, USA, 1987
- [del Solar, 2006] M.R. del Solar, J.M.S. Perez, J.A.G. Pulido, M.A.V. Rodriguez. Placement and routing of Boolean functions in constrained FPGAs using a distributed genetic algorithm and local search. In: Parallel and Distributed Processing Symposium, IEEE, 2006
- [Mitchell, 1999] Mitchell M. An Introduction to Genetic Algorithms. Fifth printing. Cambridge, MA: The MIT Press, 1999
- [Турин, 1999] S.F. Tyurin. Retention of functional completeness of Boolean functions under "failures" of the arguments. In: Automation and Remote Control, 1999
- [Турин, 2011] S. Tyurin, V. Kharchenko. Redundant Bases for Critical Systems and Infrastructures: General Approach and Variants of Implementation. In: Proceedings of the 1st International Workshop on Critical Infrastructures Safety and Security, Kirovograd, Ukraine, 2011
- [Турин, 2013] S.F. Tyurin, A.V. Grekov, O.A. Gromov, I.S. Ponurovskiy. Adaptable Logical FPGA-Elements. In: Радіоелектронні і комп'ютерні системи, Kharkov, Ukraine, 2013
- [Yakimets, 2011] N. Yakimets, V. Kharchenko. Reliable FPGA-Based Systems out of Unreliable Automata: Multi-Version Design Using Genetic Algorithms. In: Design of Digital Systems and Devices, Springer, Berlin, 2011
- [Zeidman, 2002] B. Zeidman. Designing with FPGAs and CPLDs, CMP Books, Lawrence, USA, 2002.
- [Асанов, 2001] М.О. Асанов, В.А. Баранский, В.В. Расин, Дискретная математика: графы, матроиды, алгоритмы, НИЦ «Регулярная и хаотическая динамика», Ижевск, 2001
- [Гладков, 2006] Л.А. Гладков, В.В. Курейчик, В.М. Курейчик. Генетические алгоритмы, М.:ФИЗМАТЛИТ, 2006
- [Данилова, 2009] Данилова Е.Ю., Городилов А.Ю. Сравнение генетических алгоритмов на примере задачи коммивояжера // Вестник Пермского ун-та, вып. 3(29), серия «Математика. Механика. Информатика», 2009
- [Уваров, 2007] С.С. Уваров. Проектирование реконфигурируемых отказоустойчивых систем на ПЛИС с резервированием на уровне ячеек. // Автоматика и телемеханика, №9, 2007
- [Цыбин, 2006] С.А. Цыбин, А.В. Быстрицкий, С.Н. Скуратович. Архитектура отказоустойчивой ПЛИС емкостью свыше 100 тыс. вентилях // Сборник трудов всероссийской научно-технической конференции "Проблемы разработки перспективных микро- и нанoeлектронных систем (МЭС)", 2006

Сведения об автора



Алексей Городилов – старший преподаватель каф. МО ВС, Пермский государственный национальный исследовательский университет, Россия, г. Пермь, 614990, ул. Букирева, д. 15; e-mail: gora830@yandex.ru

Основные научные интересы: Генетические алгоритмы, эвристические методы, комбинаторные алгоритмы

Two-level genetic algorithm for programmable logic devices reconfiguration

Aleksey Gorodilov

Abstract: *The article considers the problem of programmable logic device reconfiguration after a failure of logic elements. The task is relevant to areas such as space exploration and important industrial facilities management that use highly reliable fault-tolerant systems. The article considers the failures of logic elements in the configurable logic blocks. For optimal reconfiguration logic modules must be placed on a programmable logic device without using faulty logic elements, while minimizing occupied area, the bond lengths and delays. The first step in solving the problem is to choose compactly located logic elements that will be involved in the implementation of the circuit. Compactness is understood as the minimum amount of information necessary to describe the set of logic elements. This task can not be solved exactly and effectively, but can be approximately solved by using various heuristics, including genetic algorithms. This article describes several well-known approaches to the problem and offers an original two-level coding method for constructing a genetic algorithm that takes into account features of the problem and uses standard genetic operators. The proposed algorithm is implemented in practice in the form of a computer program to compare its performance with some other algorithms. Comparison of different heuristic algorithms in terms of accuracy of the solution and estimated speed of the proposed algorithm are given. One can say that the proposed genetic algorithm works on average longer than other heuristic algorithms, but finds a more accurate solution. This leads to a reduction of reconfiguration time and, thereby, increase availability ratio of the programmable logic device.*

Keywords: *programmable logic devices, reconfiguration, reliability, genetic algorithm.*

ОСОБЕННОСТИ ПОСТРОЕНИЯ СОВРЕМЕННОЙ БЕСПРОВОДНОЙ КОРПОРАТИВНОЙ СЕТИ

Юрий Лисецкий, Наталья Каревина

Аннотация: Представлено построение современной беспроводной корпоративной сети, реализующей концепцию использования собственных устройств. Сформулированы требования к современным беспроводным сетям корпоративного уровня, изложены принципы их построения и особенности архитектуры. Рассмотрены вопросы организации информационной безопасности в беспроводных корпоративных сетях.

Ключевые слова: беспроводная сеть, инфраструктура, пользовательские сервисы, профилирование устройств, точки доступа, коммутация, отказоустойчивость, триангуляция, аутентификация, контроль доступа, информационная безопасность.

ACM Classification Keywords: H. Information Systems. H.1 Models and Principles. H.1.1 Systems and Information Theory. F.1.2 Models of Computation. 1.6.1 Simulation Theory.

Введение

За последние несколько лет мобильные беспроводные устройства прочно проникли в жизнь современного человека [Палагин, 2006]. Границы рабочего места “размываются” и “мигрируют” в сторону мобильности офисной среды. Компьютерный парк на предприятиях и в организациях обновляется значительно реже по сравнению с личными (персональными) устройствами сотрудников, да и по удобству использования существенно им уступает. Массовое распространение смартфонов и планшетных компьютеров приводит к постепенному вытеснению стационарных компьютеров и ноутбуков с рабочих мест. В сложившихся условиях одна из наиболее актуальных задач в беспроводной корпоративной сети – реализация концепции использования собственных устройств (BYOD – Bring your own device). Ключевым элементом концепции является внедрение системы управления политиками и построение безопасной корпоративной беспроводной сети.

Требования к сети

В большинстве случаев вопрос развёртывания беспроводной сети решается установкой нескольких точек доступа и является как бы надстройкой над проводной инфраструктурой, управляемой отдельно [Богомолов, 2004]. Такой подход оправдывает себя в домашнем сегменте и является недостаточным для применения в корпоративном [Букин, 2007]. Требования к беспроводным сетям уже давно не ограничиваются только организацией интерфейса между беспроводным клиентским устройством и проводной инфраструктурой.

Современная корпоративная сеть должна иметь возможность передачи мультимедийных данных (данные, голос и видео), обеспечивать высокую производительность и безопасность, обладать хорошей масштабируемостью, простотой развёртывания, управления и улучшенными эксплуатационными характеристиками, а также предоставлять дополнительный функционал по расширению спектра

пользовательских сервисов и их адаптации. Кроме того, при построении корпоративной беспроводной сети внедряемое решение должно обладать возможностями профилирования подключаемых устройств, выполнять процедуру гостевого доступа и иметь возможность отслеживать местоположение излучающих устройств [Лисецкий, 2008].

Построение беспроводной сети

В основу архитектуры беспроводной сети (рис. 1) положен принцип централизованного управления и расширения беспроводных сервисов [Бобров, 2004]. Основной функционал возлагается на контроллер беспроводной сети, а точки доступа работают в «облегченном режиме». Точки доступа обеспечивают радиоинтерфейс и шифрование пользовательских данных, передавая все пользовательские данные на контроллер в зашифрованном туннеле, а также предоставляют широкие функции по диагностике и отчетности.

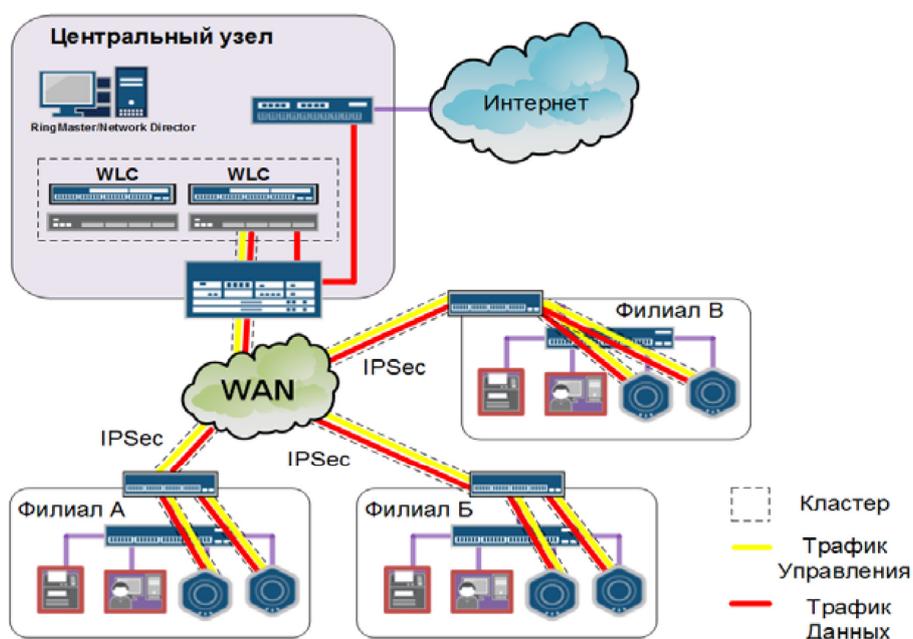


Рисунок 1. Архитектура беспроводной сети

Для организации радиопокрытия в решении присутствует широкая линейка точек доступа. Их можно условно разделить на три группы: системы начального уровня, корпоративного уровня и точки доступа во внешнем исполнении. Все предлагаемые производителем точки доступа поддерживают стандарт 802.11n и отличаются количеством радиомодулей, поддерживаемых потоков передачи данных и возможностью подключения внешней антенны. Предлагаемые устройства поддерживают три режима работы:

- точка доступа (обслуживание абонентов в режиме точка-многоточка);
- мост (реализация линии связи точка – точка);
- полносвязный режим (MESH-сеть, предусматривающая беспроводную линию связи между точками).

Применение универсальных устройств в соответствующих режимах работы делает возможным построение распределенной беспроводной сети под любые требования абонентов.

Традиционно применяются два типа архитектуры: размещение контроллера на центральном узле и в филиалах. При размещении контроллера на центральном узле применяется централизованная коммутация. При этом весь трафик проходит через контроллер, что ведет к значительной загрузке WAN-каналов. В представленных на рынке промышленных наборах программных и аппаратных средств для беспроводной сети точки доступа используют собственные приватные протоколы производителей оборудования, имеют возможность продолжать обрабатывать обращения пользователей и осуществлять взаимодействие с RADIUS-сервером в случае отказа WAN-канала. Несмотря на отсутствие перерыва в предоставлении сервиса, данные протоколы носят частный характер и предназначены для ограниченного числа удаленных офисов. Так, например, решение компании Juniper Networks свободно от данного недостатка и позволяет использовать локальную коммутацию. При этом через контроллер проходит только служебный трафик, а все данные коммутируются точкой доступа непосредственно на шлюз назначения. В системе реализована предпочтительная для предоставления мобильности модель оптимальной передачи пользовательского трафика при сохранении централизованного контроля и учета. Распределенная коммутация обеспечивает беспрецедентное качество беспроводного канала, реализуя минимальную в отрасли задержку передачи данных, что особенно важно для передачи «голосового» трафика.

Необходимость повышения отказоустойчивости сети вызвала отказ от идеи использования горячего резервирования и привела к кластеризации контроллеров. Объединение контроллеров в кластер позволяет существенно оптимизировать количество используемых лицензий на подключение точек доступа. Нет необходимости держать неиспользуемые лицензии на случай выхода из строя первичного контроллера. Основным отличием данного решения является возможность объединения в кластер любых типов контроллеров как разных аппаратных серий, так и виртуальных. Реализация общей программной платформы дает администраторам возможность проводить обновление программного обеспечения без перерыва в предоставлении сервисов.

Большинство современных беспроводных устройств поддерживает работу в двух диапазонах – 2,4 ГГц и 5 ГГц. Исторически диапазон 2,4 ГГц начал использоваться раньше и на данный момент значительно более загружен по сравнению с диапазоном 5 ГГц. Система беспроводной связи компании Juniper обеспечивает автоматическую балансировку клиентов между точками, а функция группирования клиентов осуществляет приоритетное подключение клиентских устройств в диапазон 5 ГГц, и только при невозможности присоединения происходит подключение в диапазоне 2,4 ГГц. Функционал балансирования и группирования абонентов обеспечивает оптимальное использование пропускной способности радиосети.

Для обеспечения максимальной пропускной способности данное решение позволяет обнаруживать, классифицировать и локализовать источники помех в режиме реального времени в обоих рабочих диапазонах. Вся информация по помехам консолидируется в системе управления Network director, предоставляя графический интерфейс для устранения неполадок.

С помощью системы управления Network director также осуществляется управление жизненным циклом беспроводной сети: планирование и развертывание, конфигурирование и отладка, мониторинг, отчетность и определение местоположения.

Отслеживание местоположения пользователей, реализация бесшовного перехода между точками доступа и обеспечение заданного качества обслуживания требуют более плотного размещения точек

доступа. Функции самовосстановления и качество работы сервисов обеспечиваются централизацией управления с помощью контроллера и зависят от правильного планирования беспроводной сети.

На начальном этапе разворачивания сети используется 3D-планировщик, который учитывает затухание радиосигнала в различных видах материалов, взаимное расположение точек доступа и предназначен для проведения расчетов как внутри помещения, так и на открытой местности. Данный этап очень важен для дальнейшей работы системы. В результате его выполнения осуществляется частотно-территориальное планирование беспроводной сети.

Основываясь на плане размещения точек доступа и используя метод триангуляции, система позволяет определять точное местоположение пользователя и осуществлять поиск специализированных RFID-меток.

Безопасность беспроводной сети

Применение радиолокации предоставляет возможность организовать периметр безопасности и предотвратить доступ в сеть за территорией предприятия (рис. 2). В этом случае, даже обладая действующими аутентификационными данными (логин/пароль), злоумышленник не сможет получить доступ к сети, находясь на прилегающей к офису территории.

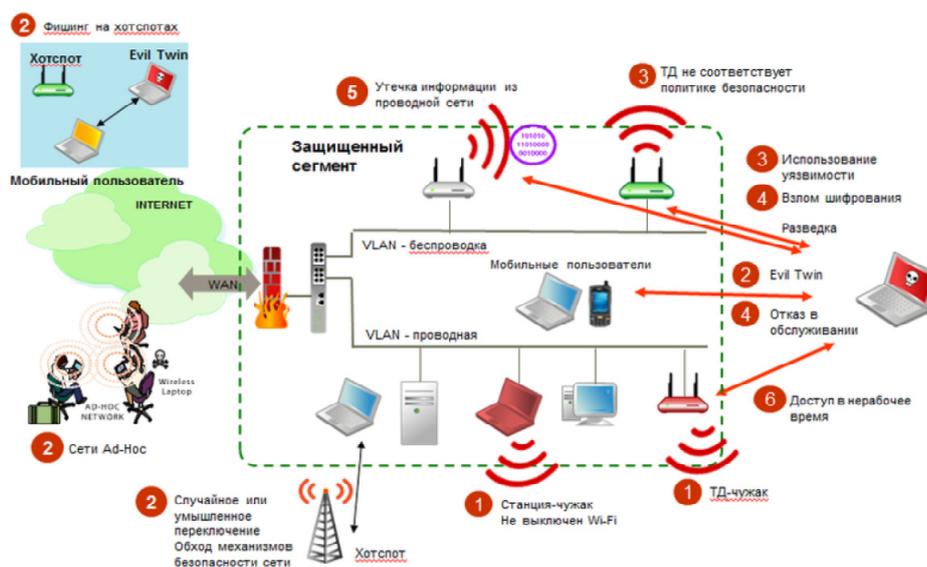


Рисунок 2. Организация безопасности беспроводной сети

Также этот функционал является интересным при выделении зон с запретом на использование беспроводного доступа, например, на территории процессингового центра или серверной, или же при отслеживании перемещений излучающего устройства по территории предприятия, предоставляя возможность поиска дорогостоящего мобильного оборудования, снабженного RFID-меткой, и осуществление контроля перемещений пользователей [Пастоев, 2006].

Повышение уровня безопасности сети осуществляется применением беспроводной системы предотвращения (обнаружения) вторжений (IPS/IDS). Система проводит внутренний анализ на наличие атак на беспроводную сеть и имеет возможность отправлять интересующий трафик на внешнюю систему

предотвращения вторжений. Возможность исторического анализа делает систему незаменимой при расследовании инцидентов.

В современной корпоративной беспроводной сети важно осуществлять динамический контроль доступа пользователей. Для этих целей служит приложение управления безопасностью SmartPass. Программное обеспечение интегрируется с системой управления Network director и системой отслеживания местоположения, обеспечивая контроль доступа на основе физического положения и возможность расширенных отчетов.

Платформа SmartPass Connect предназначена для реализации концепции использования собственных устройств. При необходимости обеспечения доступа к корпоративной сети затраты труда администраторов значительно возрастают, а при их значительных объемах могут требовать наличие в организационной структуре выделенных менеджеров. Платформа позволяет автоматизировать процедуру регистрации устройства и организацию доступа к сети с помощью корпоративных учетных данных. SmartPass интегрируется с контроллером домена и имплементирует на подключаемое устройство необходимые сертификаты. При этом система осуществляет профилирование устройства – определяется его тип, операционная система, ее версия, информация о приложениях клиента (совместно с SRX AppTrack и UAC) и т.д. Получаемая информация используется в корпоративных политиках и дает возможность предоставлять гранулированный доступ к сети.

Для доступа незарегистрированных пользователей на предприятиях необходимо внедрить процедуру гостевого доступа. SmartPass реализует полный жизненный цикл управления гостевым доступом (регистрация пользователя, уведомление о выделенных учетных данных, проведение аутентификации и авторизации, осуществление ограничения доступа, логирование событий безопасности и предоставление подробной отчетности).

Для каждого идентификатора беспроводной сети (SSID) может быть реализован адаптированный Web-портал с возможностью самообслуживания. После регистрации пользователь получает уведомление одним из трех возможных вариантов: учетные данные распечатываются на принтере, отправляются на электронную почту или отправляется СМС на указанный номер мобильного телефона. Гостевые сессии могут быть ограничены по длительности и времени использования, иметь разные политики доступа, но в то же время могут поддерживать мониторинг активных сессий, аккаунтинг, логирование и детальные отчеты. Решение поддерживает RFC 3576 предусматривающие изменение авторизации и динамическое изменение параметров во время сессии, включая качество обслуживания QoS, списки доступа ACLs, выделяемую пропускную способность и т.д.

Стандартизованный интерфейс программирования приложений API обеспечивает значительное расширение функционала беспроводной системы за счет интеграции с системами биллинга, различными платформами аналитики и отчетности.

Таким образом, решение компании Juniper Networks для построения беспроводной сети является одним из наиболее оптимальных по соотношению стоимости и доступности функционала. Предлагаемый набор промышленных программных и аппаратных средств позволяет создавать беспроводные системы корпоративного класса, обладающие высокой отказоустойчивостью и соответствующие всем требованиям по безопасности, производительности и масштабируемости.

Заключение

В настоящее время корпоративные беспроводные сети переживают период расцвета. Развитие бизнес-процессов переносит критически важные приложения в беспроводную среду, делая необходимым

реализацию интерактивных бизнес-приложений. Требования мобильности, повышения конкурентоспособности и производительности труда увеличивают требования к сервисам, предлагаемым в беспроводной сети. Описанный в данной статье подход позволяет построить универсальную корпоративную беспроводную сеть, основанную на стандартных протоколах, адаптированную под потребности практически любой организации и позволяющую оптимизировать стоимость владения беспроводным сегментом.

Литература

- [Палагин, 2006] Палагин А.В., Алишов Н.И., Громовский А.В. Средства удаленного доступа на базе мобильной персональной системы // УСИМ. – 2006. – № 6. – С. 26 – 32, 51.
- [Богомолов, 2004] Богомолов Ю. О беспроводных решениях // Экспресс-Электроника. – 2004. – № 12. – С. 33 – 40.
- [Букин, 2007] Букин М. WiMAX на корпоративном рынке // PC WEEK/RE. – 2007. – № 20. – С. 24 – 27.
- [Лисецкий, 2008] Лисецкий Ю.М., Бобров С.И. WiMax сети. Реализации и перспективы // Управляющие системы и машины. – 2008. – № 4. – С. 88 – 93.
- [Бобров, 2004] Бобров С.И. Принципы проектирования корпоративных сетей // Материалы третьей научно-практической конференции «Информационные технологии в энергетике»: сб. науч. тр. – К.: НАН Украины, Институт проблем моделирования в энергетике им. Г.Е. Пухова НАН Украины, 2004. – С. 34 – 37.
- [Пастоев, 2006] Пастоев А., Петров Н. Безопасность беспроводных сетей // Директор информационной службы. – 2006. – № 4. – С. 17 – 19.

Информация об авторах



Лисецкий Юрий - Киев, Институт проблем математических машин и систем НАН Украины, к.т.н., докторант. E-mail: Iurii.Lysetskyi@snt.ua.

Основные области научных исследований: проблемы принятия решений и управления в технических и экономических системах, математические методы, модели и технологии исследования сложных систем, информационные технологии поддержки принятия решений, проблемно и функционально-ориентированные компьютерные системы и сети.



Каревина Наталья - Киев, Институт проблем математических машин и систем НАН Украины, к.и.н., н.с. E-mail: Natka_Kn@ukr.net.

Основные области научных исследований: системы поддержки принятия решений в сложных системах, прогрессивные информационные технологии в социальных науках.

Building Peculiarities of the Modern Wireless Corporate Network

Iurii Lysetskyi, Natalia Karevina

Abstract: Under consideration is the building of the modern wireless corporate network that implements the concept of using their own devices. Represented the requirements for the modern wireless networks of the enterprise-level, set out the principles of their building and architectural peculiarities. The questions of the organization of the information security in the wireless corporate networks are considered in the article.

Keywords: wireless network, infrastructure, user services, profiling devices, access point, commutation, fail-safe feature, triangulation survey, authentication, access audit, information security.

DEPENDENCE OF INFORMATION AND TELECOMMUNICATION TECHNOLOGY DEVELOPMENT ON ECONOMIC INDICATORS

Galina Gayvoronska, Illia Ganytskyi, Petr Yatsuk

Abstract: *The article describes an analysis of the relationships between different ICT services indicators and economic indicators. Article also gives view for determining the impact of the economic state of the country on developing information and communication technologies, as well as the influence of the development one information technology on other.*

Keywords: *telecommunication technologies, ICT indicators, technology development.*

Keywords classification of ACM: *D.2.9 Management, K.6.3 Life cycle.*

Introduction

Information and communication technology (ICT) - is a sphere susceptible to rapid, continuous and cardinal changes in the entire world. Various international researches show that there is a close relationship between ICT development and the economic condition of the country. Level of telecommunication technologies development (CT) has a big impact on the economic indicators of the national economy as a whole. The main goal of this research was to determine the existence and degree of correlation between the technological and economic indicators to determine the impact of the economic state of ICT development.

To determine the dependence of the technology development level on the economic status, country-specific research performed following indicators:

1. Indicators that reflect the diversity of users simultaneously used terminals and communications equipment:
 - The proportion of households using radio, television, computer with Internet access, using the services of fixed-line telephone, mobile communications;
 - The proportion of individuals using computer as the terminal equipment and a mobile cellular telephone.
2. Economic indicators information and communications services (ICS):
 - The average salary index and the industrial production index.
3. Utilization ICS:
 - Percentage of the population that uses the ICS.

The impact of economic factors on the development of telecommunications

Statistical sample contains the values of above indicators for 53 countries over the period from 2008 to 2012. Source of statistics is Internet resources such as International Telecommunication Union (ITU) [ITU, 2014] and the United Nations Economic Commission for Europe (UNECE) [UNECE, 2014]. Due to the facts that in the given statistics are missing values, they are taken as "0.00". That is why the results of calculations have an error. However, the number of such values is small, and the impact of missing data on the results of the research is insignificant.

To investigate the relationship between ICS economic indicators availability [ITU, 2009] and ICS utilization rates correlation analysis is applied [Bendat & Piersol, 1993]. The main goal of the analysis is determine degree and nature of the relationship between random observations, which are distributed by multivariate normal law. The correlation coefficient characterizes the presence of a linear relationship between the indicators X, Y , and is described by the expression [Astafurova, 2014]:

$$r_{x,y} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 (y_i - \bar{y})^2}} \quad (1)$$

where x_i – values, from sample X ; y_i – values, from sample Y ; \bar{x} , \bar{y} – average values of samples X and Y .

The degree of association between the indicators is estimated using the correlation coefficient calculated using the formula (1) and has a value from -1 to +1. Positive or negative sign near one points to a direct or inverse relationship between values x and y [Eliseeva, 2001].

Average values of the samples X and Y are defined as:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i = \frac{1}{n} (x_1 + x_2 + \dots + x_n) \quad (2)$$

$$\bar{y} = \frac{1}{n} \sum_{i=1}^n y_i = \frac{1}{n} (y_1 + y_2 + \dots + y_n) \quad (3)$$

where n – size of the sample.

The lack of normal distribution of the parameters is defined while preprocessing source statistics. In this regard, statistics are calculated by use of nonparametric methods [Hartigan, 1975; Jain & Dubes, 1988; Johnsonbaugh & Schaefer, 2004]. The results of statistics calculations with methods of nonparametric statistics are presented in the form of a matrix in Table 1. The rows describe the main statistical random variables characteristics for the given parameters.

Table 1. The results of statistical indicators calculations

	EE1	EE2	EE3	HH1	HH2	HH3f	HH3m	HH4	HH6	HH5	HH10
SS1	2227,97	109,5	60,17	59,07	74,93	37,73	77,00	46,46	43,47	59,63	79,15
SS2	3750301	449,5	578,7	464,2	872,9	822,5	446,6	887,1	917,9	512,8	311,8
SS3	1936,57	21,20	24,06	21,55	29,55	28,68	21,13	29,79	30,30	22,65	17,66
SS4	6742,50	191,2	94,60	99,60	100,0	94,80	99,30	96,00	97,40	99,70	99,64
SS5	82,10	76,40	8,00	4,58	5,30	0,30	26,30	0,13	0,08	0,90	29,56
SS6	6660,40	114,8	86,60	95,02	94,70	94,50	73,00	95,87	97,32	98,80	70,07
SS7	-	90,80	67,00	82,00	100,0	37,80	85,10	92,00	63,00	-	-
SS8	1287,00	104,9	65,95	60,20	89,61	37,80	85,10	48,96	40,98	61,19	85,66

Characteristics and parameters used in the table:

SS1 - the average value of the sample, calculated by formula (2) or (3).

SS2 - dispersion calculated by the formula:

$$D_B = \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n} \quad (4)$$

where x_i - i -th element of the sample; \bar{x} - the average value of the sample; n - sample size.

SS3 - standard deviation - sample values scattering factor calculated by the formula:

$$\sigma = \sqrt{D_B} = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n}} \quad (5)$$

where x_i - i -th element of the sample; \bar{x} - the average value of the sample; n - sample size.

SS4 - the maximum value of the sample.

SS5 - minimum value of the sample.

SS6 - the range defined by the difference of maximum and minimum values.

SS7 - mode, the most frequent value in the sample.

SS8 - median - the value is the midpoint of a set.

EE1 - average wage value in the country in U.S. dollars.

EE2 - the industrial production index.

EE3 - percentage of the population that uses ICS.

HH1 - the percentage of households with a radio.

HH2 - the percentage of households with a television.

HH3f - the percentage of households with fix-line phone only.

HH3m - the percentage of households with a mobile cellular telephone.

HH4 - the percentage of households with a computer.

HH5 - the percentage of individuals who used a computer in the last 12 months.

HH6 - the percentage of households with Internet access at home.

HH10 - percentage of individuals using wireless technology.

The resulting correlation values calculated by formula (1) are shown in Table 2.

Table 2. Correlation coefficients

	EE1	EE2	EE3	HH1	HH2	HH3f	HH3m	HH4	HH6	HH5	HH10
EE1	1	0,398	0,771	0,257	0,059	0,281	0,103	0,652	0,753	0,499	0,411
EE2	0,398	1	0,374	0,247	0,008	0,174	0,053	0,352	0,365	0,344	0,262
EE3	0,771	0,374	1	0,264	0,016	0,209	0,052	0,769	0,839	0,650	0,488
HH1	0,257	0,247	0,264	1	0,188	0,420	0,385	0,260	0,278	0,139	0,096
HH2	0,059	0,008	0,016	0,188	1	0,491	0,477	0,107	0,055	0,093	0,092

HH3f	0,281	0,174	0,209	0,420	0,491	1	0,780	0,102	0,180	0,311	0,323
HH3m	0,103	0,053	0,052	0,385	0,477	0,780	1	0,016	0,076	0,257	0,299
HH4	0,652	0,352	0,769	0,260	0,107	0,102	0,016	1	0,931	0,826	0,746
HH6	0,753	0,365	0,839	0,278	0,055	0,180	0,076	0,931	1	0,779	0,706
HH5	0,499	0,344	0,650	0,139	0,093	0,311	0,257	0,826	0,779	1	0,907
HH10	0,411	0,262	0,488	0,096	0,092	0,323	0,299	0,746	0,706	0,907	1

The following relationships based on the values of the correlation coefficients are highlighted:

- A fairly high rate of correlation (0.771) between the growth of wages (EE1) and increasing number of ICS users (EE3);
- Average correlation (0.652) between the level of wages of the population (EE1) and percentage the use of computers (HH4);
- High value of the correlation coefficient (0.753) for wages (EE1) and access to the Internet for home users (HH6);
- Strong dependence (0.769) exists between the percentage of the population that uses the ICS (EE3) and users, which have a computer at home (HH4);
- Strong dependence (0.839) between the percentage of the population that uses the ICS (EE3) and users that have access to the Internet at home (HH6);
- The average dependence (0.650) between the percentage of the population that uses the ICs (EE3) and the use of computer technology in industry (HH5);
- High value of the correlation coefficient (0.780) between the access and use of stationary (HH3f) and mobile (HH3m) telephony that is caused due the needs of connection between users regardless of the location;
- Very high degree of correlation (0.931) between the indicators (HH4) and (HH6). Consequently, almost any user of computer technology in the household has access to the Internet;
- Highest correlation value (0.826) between the number of home computers (HH4) and computers at work (HH5) indicates uniform implementation of computer technology;
- A high correlation (0.746) between the indicators (HH4) and (HH10) shows that with computer technology home users prefer to use mobile technology;
- Highest correlation value (0.779) for samples (HH6) and (HH5) may be due to the fact that more information resources become available, improving the service sector, increasing information infrastructure in general;
- There is a high degree of dependence (0.706) between the access to the Internet (HH6) and the use of mobile technology (HH10);
- Very high correlation (0.907) between the indicators (HH5) and (HH10), showing that with computer technology mobile technology is introduced into the industry.

Conclusion

The obtained results concretize proverbial dependence of ICT development on the economic condition of the country, and allow us to estimate how the increases of resources allocated to information technology are increasing their qualitative and quantitative characteristics. Analysis of the results shows that most of the ICT

takes just computer technology, both in the household and in industry. The close relationship between the fixed and mobile telephone connection points on their mutual influence, so a person who has a fixed-line phone probably has a mobile phone for access to the telephone network, regardless of location. Together with the need for the entire spectrum of the ICS regardless of location through the development of wireless technologies we have a big impact of computer technology.

Bibliography

- [Astafurova, 2014] Astafurova I.S., Linear pair correlation. Access mode: http://abc.vvsu.ru/Books/um_statistikazo/page0004.asp
- [Bendat & Piersol, 1993] Bendat J.S., Piersol A.G., Engineering application of correlation and spectral analysis. Wiley-Interscience, New York, NY, 1993.
- [Eliseeva, 2001] Eliseeva I.I., Econometrics: A Textbook. Ed. II - Moscow: Finance and Statistics, 2001.
- [Hartigan, 1975] J.A. Hartigan, Clustering Algorithms, Wiley and Sons, 1975.
- [ITU, 2009] Manual for measuring and use households and individuals to ICS. Access mode: http://www.itu.int/dms_pub/itu-d/opb/ind/D-IND-ITCMEAS-2009-PDF-R.pdf
- [ITU, 2014] ITU Statistics. Access mode: <http://www.itu.int/en/ITU-D/Statistics/Pages/stat/default.aspx>
- [Jain & Dubes, 1988] A.K. Jain and R.C. Dubes, Algorithms for Clustering Data, Prentice Hall, 1988.
- [Johnsonbaugh & Schaefer, 2004] R. Johnsonbaugh, M. Schaefer, Algorithms, Pearson Prentice Hall, 2004, ISBN 0-13-122853-6.
- [UNECE, 2014] UNECE Statistics. Access mode: <http://w3.unece.org/pxweb/>

Authors' Information



Galina Gayvoronska – School of information technologies and cybersecurity of V.S. Martinovsky Institute of Refrigeration Cryotechnologies and Ecoenergetics ONAFT, technical science's doctor, professor, chief of the information-communication technologies' department; Dvoryanskaya str., 1/3, Odessa-26, 65026, Ukraine; tel. (048)-720-91-81, e-mail: GSGayvoronska@gmail.com

Major fields of scientific research: optimization of transient periods at telecommunication networks' evolution. Calls' streams, load and intermodal inclination in nets. Problems of perspective access networks' and fully optical switching systems' development.



Illia Gannytskyi – School of information technologies and cybersecurity of V.S. Martinovsky Institute of Refrigeration Cryotechnologies and Ecoenergetics ONAFT, PhD Telecommunication technologies, lecturer in the information-communication technologies department, Dvoryanskaya str., 1/3, Odessa-26, 65026, Ukraine; tel. (048)-720-91-48;

e-mail: igannytskyi@ikt-osar.od.ua

Major fields of scientific research: streams of calls on telecommunication network.



Petro Yatsuk – National Commission for the State Regulation of Communications and Informatization, Chairman, Khreshchatyk Str., 22, Kyiv – 01001, Ukraine.

e-mail: yatsuk@ikt-osar.od.ua

Major fields of scientific research: telecommunication networks.

МЕТОД ОБЕСПЕЧЕНИЯ СТРУКТУРНОЙ ЖИВУЧЕСТИ ТЕЛЕКОММУНИКАЦИОННОЙ СЕТИ

Нина Князева

Аннотация: Представлен метод обеспечения структурной живучести телекоммуникационной сети, реализующий потоковый подход на основе предложенного показателя структурной живучести – нижней границы структурной живучести, обеспечивающей гарантированное устойчивое функционирование телекоммуникационной сети, достижение требуемого значения которой осуществляется введением структурной избыточности, осуществляемой с использованием оптимизационной процедуры резервирования тех ветвей сечений, разделяющих множество путей, организуемых для обслуживания поступающих в сеть потоков требований, выбор которых определяется на основе максимизации эффективности каждого этапа резервирования.

Ключевые слова: структурная живучесть, сеть, резервирование, путь, сечение, показатель живучести.

Ключевые слова классификации АСМ: С.2. Computer-communication networks, Н. Information Systems - Н.1 Models and Principles, К. Computing Milieux - К.6 Management of computing and information system.

Введение

Одним из важнейших свойств телекоммуникационных сетей, обеспечивающих их эффективное функционирование, является живучесть. Вопросы обеспечения живучести телекоммуникационных сетей становятся все более актуальными в связи с их интенсивным развитием в последнее десятилетие. Современные тенденции развития телекоммуникаций, переход к сетям нового поколения – NGN (Next Generation Network – сети следующего поколения), FGN (Future Generation Network – сети будущего поколения (БС)) – обуславливают особую важность вопросов оценки, обеспечения и повышения живучести ТКС с учетом тенденций их развития. Применительно к Future Networks в документах ИТУ [1] рекомендуется, чтобы проектирование, эксплуатация и развитие БС осуществлялись таким образом, чтобы обеспечить надежность, безопасность, а также живучесть, определяющую возможность системы своевременно выполнять свою миссию – обеспечивать выполнение критического подмножества функций для достижения цели функционирования, а также приемлемый уровень обслуживания, даже если нормальная эксплуатация сети осложняется различными проблемами. Для обеспечения живучести телекоммуникационных сетей требуется решение ряда научно-технических задач на всех этапах их жизненного цикла – проектирования, модернизации или совершенствования, эксплуатации. Решение задач, в ходе которых удовлетворяются требования живучести, как правило, организуется путем поиска оптимальных или квазиоптимальных вариантов структурных преобразований и способов защиты элементов систем.

Под живучестью телекоммуникационной сети (ТКС) понимают свойство, характеризующее способность сети эффективно функционировать при получении повреждений (разрушений) вследствие некоторых

неблагоприятных внешних воздействий или восстанавливать данную способность в течение заданного времени [2, 3]. Существенный вклад в разработку вопросов теории живучести систем различного назначения внесли работы В.М. Вишневого, Ю.Ю. Громова, А.Г. Додонова, В.Ф. Крапивина, И.А. Рябинина, В.К. Попкова, Ю.И. Стекольников и др. Следует отметить, что на данном этапе развития теории живучести пока не сложились условия, позволяющие сформулировать и реализовать единый подход к разработке задач оптимизации характеристик живучести различных типов систем. Существенная специфика в построении и функционировании ассоциативных, ассоциативно-структурных и структурных систем вынуждает искать специфические подходы при постановке и решении соответствующих оптимизационных задач. Тем не менее, сформулированы принципы, руководство которыми приводит к положительному эффекту в деле обеспечения живучести систем различного назначения [2].

Принцип 1. Элементы системы должны обладать малой структурной значимостью и высокой стойкостью.

Принцип 2. Структура системы должна обеспечивать возможно большее или достаточное (в задачах оптимизации) число состояний способности.

Принцип 3. Состояния способности системы должны обеспечиваться возможно меньшим числом элементов.

Принцип 4. Различные состояния способности системы должны обеспечиваться различными элементами.

Конкретные способы обеспечения живучести систем, как правило, являются реализацией одного или одновременно нескольких принципов. При этом важным вопросом является выбор показателей живучести, которые должны удовлетворять следующим основным требованиям [2]:

- а) по смысловому содержанию выбор должен соответствовать определению свойства живучести систем;
- б) показатель должен обеспечивать достаточно высокий уровень системности исследований;
- в) показатель должен обеспечивать возможность разработки моделей живучести систем, доступных для проведения исследований и выполнения расчетов;
- г) показатель должен быть чувствительным к манипуляциям на уровне характеристик свойства живучести.

Особую актуальность сегодня приобрели задачи синтеза живучих систем, оценки, обеспечения и повышения живучести систем, что определяется потребностями практики. Разработка подходов к решению задач синтеза живучих систем, оценки, обеспечения и повышения живучести систем предполагает поиск и формулирование системы устойчивых взглядов (правил, принципов) на возможность эффективного управления свойством живучести систем. Известно, что оценка показателей живучести систем осуществляется аналитическими методами. Существуют различные аналитические модели и методы синтеза живучих систем, оценки, обеспечения и повышения живучести систем различного назначения, основанные на разнообразных подходах – на использовании теоретико-игрового, энтропийного подходов, на использовании логико-вероятностных, детерминированных и стохастических моделей невозстанавливаемых и восстанавливаемых систем, а также методы, которые основаны на использовании искусственного интеллекта [2 – 4]. Отметим, что рассмотренные общие проблемы синтеза

живучих систем, оценки, обеспечения и повышения живучести систем различного назначения относятся и к ТКС.

Постановка задачи исследования

Телекоммуникационные сети относятся к структурным системам, в соответствии с чем особое значение для ТКС приобретают вопросы обеспечения структурной живучести, что определяет необходимость разработки методов обеспечения структурной живучести ТКС, основанных на использовании потокового подхода [3, 5]. Целью данной работы является разработка метода обеспечения структурной живучести ТКС, реализующего потоковый подход с использованием предложенного показателя структурной живучести ТКС, определяемого нижней границей структурной живучести, обеспечивающей гарантированное устойчивое функционирование ТКС, достижение требуемого значения которой обеспечивается введением структурной избыточности (поскольку свойством живучести обладают только избыточные системы), осуществляемой с использованием оптимизационной процедуры резервирования тех ветвей сечений, разделяющих множество путей, организуемых для обслуживания поступающих в сеть потоков требований, выбор которых определяется на основе максимизации эффективности каждого этапа резервирования.

Выбор показателя структурной живучести телекоммуникационной сети

Выбор показателя живучести системы определяется исключительно типом задачи, решаемой в ходе исследования. Для ТКС показатели структурной живучести интерпретируются как количественные меры связности для структуры графа, при этом задачи оценки структурной живучести сети можно свести к задачам анализа связности графов, оценки величины кратчайших путей и максимальных потоков, оценки вероятности формирования работоспособной структуры в случае неблагоприятных внешних воздействий и т.п. [3, 5]. В работе [2] отмечается, что все многообразие задач исследования и разработки живучих систем может быть обеспечено тремя критериями – пригодности, сравнительной оценки и оптимальности. Основной задачей ТКС является обслуживание потоков требований различных классов в соответствии с требуемым качеством обслуживания QoS (Quality of Service) для соответствующих классов потоков, что обуславливает необходимость в разработке показателя, базирующегося на так называемом «потоковом» подходе. Именно потоковый подход позволяет [2, 3, 5]:

- обосновать требования к структуре ТКС с учетом заданного (директивного) уровня показателя структурной живучести;
- выбрать лучший по показателю структурной живучести проект системы из рассматриваемых альтернатив;
- обеспечить показателю структурной живучести максимальное значение в некоторой области допустимых значений.

В данной работе на основе использования потокового подхода в качестве показателя структурной живучести предложена нижняя граница структурной живучести (НГСЖ) ТКС, обеспечивающая гарантированное устойчивое функционирование ТКС, определяемая как средневзвешенное значение вероятностей непоражения множества допустимых сечений сети, разделяющих множество допустимых путей, организуемых для обслуживания поступающих в сеть потоков требований, оптимизация значения которого состоит в достижении его максимального значения при существующих ограничениях.

Выбор показателя – НГСЖ – определяется тем, что данный показатель удовлетворяет основным требованиям (а) – (г) [2], а именно:

- а) по смысловому содержанию НГСЖ определяется на основе потокового подхода, который используется для решения задач синтеза живучих систем, оценки, обеспечения и повышения живучести систем;
- б) достаточно высокий уровень системности исследований обеспечивается учетом множеств всех допустимых путей, организуемых для обслуживания поступающих в сеть потоков требований, и множеств всех допустимых сечений, разделяющих эти пути;
- в) показатель НГСЖ имеет высокую степень формализации, что позволяет выполнить его расчет и провести исследование его изменений при изменении ситуации в ТКС;
- г) расчет НГСЖ осуществляется на основе учета требований на передачу потоков информации, поступивших в сеть, и возможностей по их обслуживанию, что позволяет обеспечить чувствительность показателя к любым манипуляциям на уровне характеристик ТКС.

Кроме того, данный показатель полностью соответствует критериям разработки живучих систем – пригодности, сравнительной оценки и оптимальности. НГСЖ определяется на основе потокового подхода, что соответствует критерию пригодности; определение достижения показателем требуемого значения осуществляется на основе учета существующих ограничений, что определяется критерием сравнительной оценки; требуемое значение НГСЖ достигается в процессе оптимизации (максимизации) его значения, что соответствует критерию оптимальности.

Метод обеспечения структурной живучести телекоммуникационной сети

Процедурная модель метода обеспечения структурной живучести ТКС представлена на рис. 1.

Расчет и максимизация НГСЖ осуществляется следующим образом.

1. Формирование исходных данных

Имеется ТКС, модель которой описывается взвешенным графом с заданными матрицами вероятностей непоражения ветвей сети (ребер графа) $P = [p_{st}]$ и требований на передачу потоков информации $\Phi = [\varphi_{st}]$ ($s, t = \overline{1, n}$, $s \neq t$, n – количество пунктов сети). Каждый поток φ_{st} принадлежит соответствующему классу k ($k = \overline{1, K}$, K – число классов потоков). Каждому классу k присваивается приоритет исходя из заданного показателя качества обслуживания QoS. Для потоков φ_{st} соответствующего класса задается их значимость – «вес» w_{st} . Задаются также:

- а) требуемое значение нижней границы структурной живучести НГСЖ_{тр} ТКС, при котором обеспечивается гарантированное устойчивое функционирование ТКС с заданным качеством обслуживания поступающих в сеть потоков требований φ_{st} ($s, t = \overline{1, n}$, $s \neq t$, n – число пунктов сети), достижение которого обеспечивается введением минимально необходимого дополнительного резервного оборудования стоимостью СДР – при решении прямой задачи оптимального резервирования;
- б) предельная стоимость дополнительного резервного оборудования СДР_{пр}, при введении которого обеспечивается возможность достижения максимального (при данных затратах на резервирование) значения нижней границы структурной живучести НГСЖ ТКС – при решении обратной задачи оптимального резервирования.

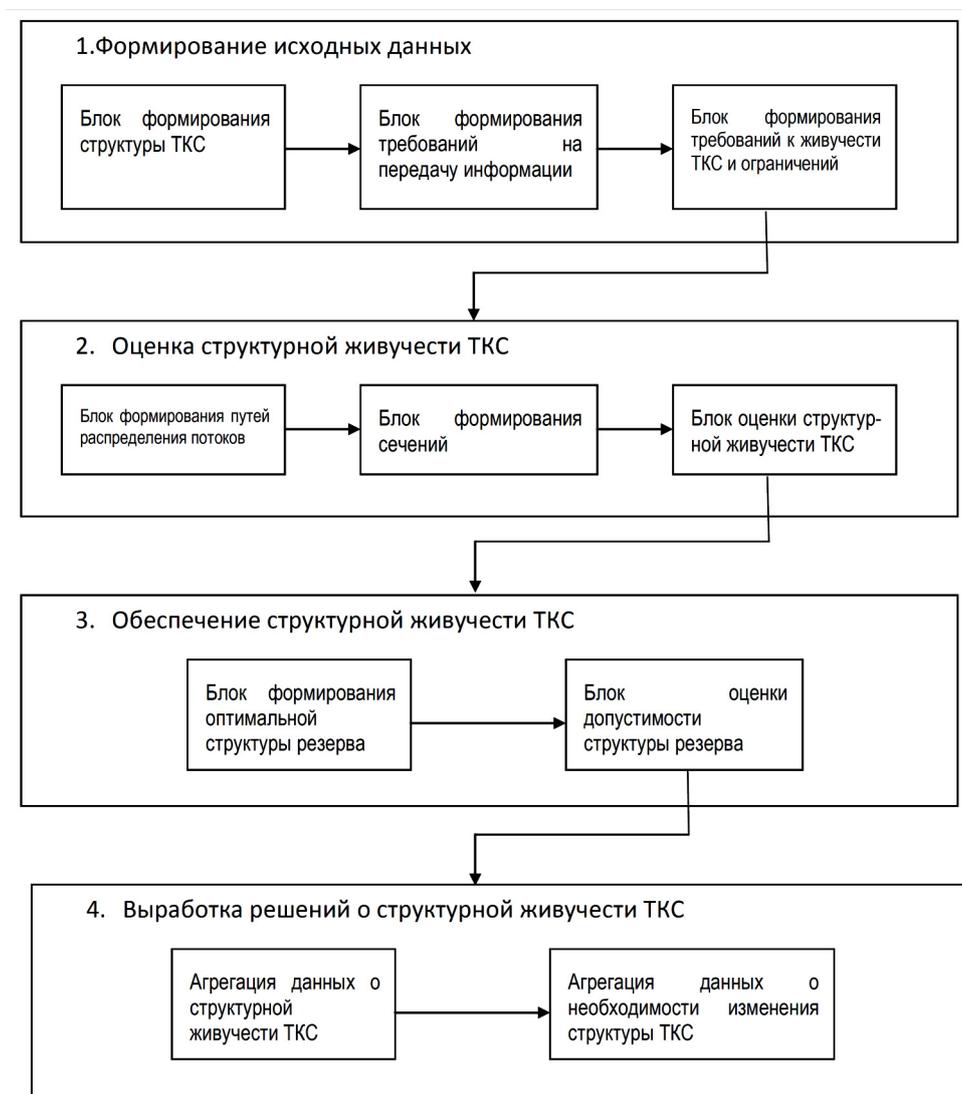


Рисунок 1. Процедурная модель метода обеспечения структурной живучести ТКС

2. Оценка структурной живучести ТКС

Для каждого требования на передачу потоков информации φ_{st} ($s, t = \overline{1, n}$, $s \neq t$, n – число пунктов сети) формируется множество путей m_{st} для его обслуживания. С использованием представленных принципов 1-4 обеспечения живучести систем среди множества путей m_{st} выбираются то множество допустимых путей $m_{st}^{\text{доп}}$, которые соответствуют указанным принципам, и для них формируется множество сечений, которые ранжируются также с учетом указанных принципов. В результате формируется множество допустимых сечений $\sigma_{st}^{\text{доп}}$. Для каждого требования φ_{st} осуществляется расчет значения вероятности непоражения π_{st} множества допустимых сечений $\sigma_{st}^{\text{доп}}$ [6].

Вероятность непоражения π_{st}^l l -го сечения $\sigma_{st}^l \in \sigma_{st}^{\text{доп}}$ ($l = \overline{1, L}$, L – количество сечений, составляющих множество $\sigma_{st}^{\text{доп}}$) рассчитывается как вероятность параллельного соединения ветвей β_{xy} , которые составляют сечение σ_{st}^l :

$$\pi_{st}^l = 1 - \prod_{\beta_{xy} \in \sigma_{st}^l} (1 - \rho_{xy}) \quad (1)$$

Вероятность непоражения π_{st} всех допустимых сечений $\sigma_{st}^{доп}$, которые разделяют пункты s и t , рассчитывается как вероятность последовательного соединения сечений:

$$\pi_{st} = \prod_{\sigma_{st}^l \in \sigma_{st}^{доп}} (1 - \prod_{\beta_{xy} \in \sigma_{st}^l} (1 - \rho_{xy})) \quad (2)$$

Нижняя граница структурной живучести НГСЖ ТКС определяется как средневзвешенная величина по всем рассчитанным значениям π_{st} для множеств сечений $\sigma_{st}^{доп}$, при этом веса w_{st} , с которыми учитываются значения π_{st} в выражении (3), зависят от класса k ($k = \overline{1, K}$, K – число классов потоков) потоков требований φ_{st} ($s, t = \overline{1, n}$, $s \neq t$, n – число пунктов сети):

$$\text{НГСЖ} = \sum_{s=1}^n \sum_{t=1}^n \pi_{st} \cdot w_{st} \quad (3)$$

Выражение (3) для расчета НГСЖ используется при условии (4):

$$\sum_{s=1}^n \sum_{t=1}^n w_{st} = 1. \quad (4)$$

Если значения w_{st} не связаны нормировочным условием (4), то НГСЖ рассчитывается в соответствии с выражением (5):

$$\text{НГСЖ} = \left(\sum_{s=1}^n \sum_{t=1}^n \pi_{st} \cdot w_{st} \right) / \left(\sum_{s=1}^n \sum_{t=1}^n w_{st} \right). \quad (5)$$

Полученное значение НГСЖ сравнивается с требуемым значением нижней границы структурной живучести НГСЖ_{тр} ТКС:

$$\text{НГСЖ} \geq \text{НГСЖ}_{тр}. \quad (6)$$

При выполнении условия (6) – значение показателя структурной живучести ТКС соответствует требуемому, при невыполнении – переход к осуществлению последовательных шагов оптимального резервирования ветвей сечений, разделяющих потоки требований φ_{st} в сети для обеспечения требуемого значения показателя структурной живучести ТКС – к этапу 3 метода.

3. Обеспечение структурной живучести ТКС

Возможна постановка двух оптимальных задач математического программирования, связанных с обеспечением ТКС достаточным количеством резервных элементов для достижения требуемого значения показателя структурной живучести ТКС.

1. Необходимо найти такое количество резервных элементов для сети (системы), чтобы требуемый показатель структурной живучести сети в целом обеспечивался при минимальных суммарных затратах на все резервные элементы.
2. Необходимо найти такое количество резервных элементов для сети (системы), чтобы при заданных допустимых затратах на систему резервных элементов в целом обеспечить максимально возможный показатель структурной живучести сети.

В формальной записи эти задачи имеют следующий вид.

I. Прямая задача

Найти вектор оптимальной структуры резерва $M(m_1, m_2, \dots, m_b)$, обеспечивающий минимум функции

$$\text{СДР} = \sum_{i=1}^b c_i m_i \rightarrow \min \quad (7)$$

при условии

$$\text{НГСЖ} = \sum_{s=1}^n \sum_{t=1}^n \pi_{st} \cdot w_{st} \geq \text{НГСЖ}_{\text{тр}} \quad (8)$$

(или при невыполнении условия (4) $\text{НГСЖ} = ((\sum_{s=1}^n \sum_{t=1}^n \pi_{st} \cdot w_{st}) / (\sum_{s=1}^n \sum_{t=1}^n w_{st})) \geq \text{НГСЖ}_{\text{тр}}$).

II. Обратная задача

Найти вектор оптимальной структуры резерва $M(m_1, m_2, \dots, m_b)$, обеспечивающий максимум функции

$$\text{НГСЖ} = \sum_{s=1}^n \sum_{t=1}^n \pi_{st} \cdot w_{st} \rightarrow \max \quad (9)$$

(или при невыполнении условия (4) $\text{НГСЖ} = ((\sum_{s=1}^n \sum_{t=1}^n \pi_{st} \cdot w_{st}) / (\sum_{s=1}^n \sum_{t=1}^n w_{st})) \rightarrow \max$)

при условии

$$\text{СДР} = \sum_{i=1}^b c_i m_i \leq \text{СДР}_{\text{пр}} \quad (10)$$

Здесь использованы следующие обозначения:

НГСЖ – нижняя граница структурной живучести сети (НГСЖ_{тр} – требуемое необходимое значение);

СДР – стоимость дополнительного резервного оборудования сети (СДР_{пр} – заданное предельное значение);

m_i – число резервных элементов на i -м участке системы;

b – количество участков системы;

c_i – стоимость одного элемента i -го участка системы;

π_{st} – вероятность непоражения множества всех допустимых сечений $\sigma_{st}^{\text{доп}}$, которые разделяют пункты s и t , рассчитывается в соответствии с выражением (2);

w_{st} – заданная значимость («вес») потоков φ_{st} соответствующего класса;

$s, t = \overline{1, n}$, $s \neq t$, n – число пунктов сети;

Участки системы – это ветви сети, образующие сечения для заданных потоков требований на передачу информации. Последовательная нумерация ветвей сети принята для удобства формулировки задачи прямого (обратного) резервирования.

Для решения этих двойственных задач оптимального резервирования наиболее эффективно применение градиентного метода. Удобство этого метода заключается в том, что для решения задачи в этом случае не нужно знать аналитического выражения исследуемой функции, а нужно лишь иметь значение функции

и ее первых частных производных в точках, в какие мы попадаем в процессе движения к экстремуму функции [6].

Создание оптимальной резервируемой системы представим в виде следующего многошагового процесса. Рассматривается система, которая состоит из b участков, при том, что на начальном этапе процесса допускается, что ни у одного из участков нет резервных элементов. На первом шаге процесса отыскивается такой участок, добавление к которому одного резервного элемента дает наибольший "удельный" выигрыш в приросте показателя структурной живучести системы в целом, то есть наибольший прирост на единицу стоимости. На втором шаге отыскивается следующий участок (включая и тот, к которому только что был присоединен резервный элемент), добавление к которому одного резервного элемента дает опять наибольшее относительное увеличение показателя структурной живучести системы в целом. Аналогичным способом процесс построения оптимальной системы продолжается дальше.

Допустим, что на некотором шаге построенного таким образом процесса каждый i -й участок уже имеет по m_i резервных элементов, то есть всего сделано $k = \sum_{i=1}^b m_i$ шагов описанного процесса.

Для указанного k -го шага вероятность непоражения $\pi_{st}(k)$ всех допустимых сечений $\sigma_{st}^{доп}$, разделяющих пункты s и t , которая обеспечивается в результате выполняемого резервирования, рассчитывается в соответствии с выражением (11):

$$\pi_{st}(k) = \pi_{st}(m_1, m_2, \dots, m_b) = \prod_{\sigma_{st}^i \in \sigma_{st}^{доп}} (1 - \prod_{i \in \sigma_{st}^i} (1 - \pi_i(m_i))). \quad (11)$$

Здесь $\pi_i(m_i)$ – вероятность непоражения i -го участка (ветви ТКС), обеспечиваемая при кратности резервирования m_i .

$\pi_{st}(k)$ на основе выражения (11) рассчитывается для всех заданных потоков требований φ_{st} ($s, t = \overline{1, n}$). В результате определяется НГСЖ в соответствии с выражением (3) (или (5), если не выполняется условие(4)).

Суммарная стоимость резервных элементов после выполнения k -го шага будет равняться (12):

$$C(k) = C(m_1, m_2, \dots, m_b) = \sum_{i=1}^b c_i m_i. \quad (12)$$

Отметим, что $k = \sum_{i=1}^b m_i$.

Для выбора направления движения на $(k + 1)$ -ом шаге процесса следует прибавить еще один резервный элемент к такому i -му участку (i -й ветви) ($i = \overline{1, b}$), для которого максимальной является величина:

$$\gamma(m_i + 1) = \frac{1}{c_i} [(\pi_{st}(m_1, m_2, \dots, m_i + 1, \dots, m_b) - \pi_{st}(m_1, m_2, \dots, m_i, \dots, m_b))].$$

После преобразования получим:

$$\gamma(m_i + 1) = \frac{1}{c_i} \prod_{i=1}^b \pi_i(m_i) \left[\frac{\pi_i(m_i + 1)}{\pi_i(m_i)} - 1 \right] = \text{НГСЖ}(k) \frac{\pi_i(m_i + 1) - \pi_i(m_i)}{c_i \cdot \pi_i(m_i)} \quad (13)$$

Здесь $\text{НГСЖ}(k)$ – значение показателя структурной живучести ТКС, полученное в результате выполнения k шагов процесса обеспечения структурной живучести. Поскольку величина $\text{НГСЖ}(k)$ входит множителем при расчете величины $\gamma(m_i+1)$ (выражение (13)) для всех $i = \overline{1, b}$, можно упростить вычислительную процедуру, связанную с проведением процесса построения оптимальной системы. Очевидно, что содержание процесса не изменится, если на каждом шаге выбирать тот участок, в котором наибольшей является величина

$$\gamma(m_i+1) = \frac{\pi_i(m_i+1) - \pi_i(m_i)}{c_i \cdot \pi_i(m_i)} \quad (14)$$

Поэтому, рассчитав для каждого участка значение $\gamma(m_i+1)$, можно определить, в какой последовательности нужно добавлять резервные элементы, чтобы, остановившись на любом шаге процесса, мы получили наилучшее решение для данного шага. Процесс останавливается на таком k -ом шаге, при котором для задачи первого вида (прямой) выполняется условие (15):

$$\text{НГСЖ}(k-1) < \text{НГСЖ}_{\text{тр}} \leq \text{НГСЖ}(k), \quad (15)$$

для задачи второго вида (обратной) выполняется условие (16):

$$\text{СДР}(k) \leq \text{СДР}_{\text{пр}} < \text{СДР}(k-1). \quad (16)$$

4. Выработка решений о структурной живучести ТКС

При выполнении условия (15) (для прямой задачи вида (7) с ограничением (8)) или условия (16) (для обратной задачи вида (9) с ограничением (10)) – значения требуемых показателей структурной живучести сети достигнуты. В противном случае – выработка следующих возможных рекомендаций по:

- Изменению значений элементов матрицы вероятностей непоражения ветвей сети на вычисленную в результате решения задачи величину;
- Изменению классов (приоритетов) и «весов» определенных требований;
- Ограничению величины поступающих в сеть требований;
- Изменению задаваемых предельно допустимых значений нижней границы структурной живучести $\text{НГСЖ}_{\text{тр}}$ ТКС – при решении прямой задачи оптимального резервирования или предельной стоимости дополнительного резервного оборудования $\text{СДР}_{\text{пр}}$ – при решении обратной задачи оптимального резервирования.

Предложенный метод может быть использован в случае, когда множества допустимых путей и допустимых сечений являются множества всех возможных путей и всех возможных сечений в ТКС.

Пример расчета и максимизации НГСЖ

Реализацию предложенного метода обеспечения структурной живучести ТКС выполним для сети, представленной на рис.2.

1. Формирование исходных данных.

Сеть обслуживает поступающие потоки требований: $\varphi_{14}, \varphi_{36}, \varphi_{35}, \varphi_{42}$. Приоритеты всех потоков заданы одинаковыми. Заданы стоимости резервного оборудования ветвей (в условных единицах) и значения вероятностей непоражения ветвей сети, которые приняты одинаковыми для основного и резервного оборудования ветвей (табл.1).

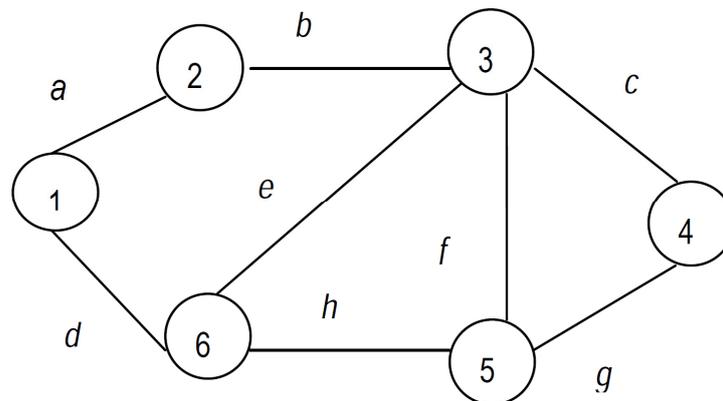


Рисунок 2. Граф сети

Таблица 1. Исходная информация по ветвям сети

Ветвь	a	b	c	d	e	f	g	h
Стоимость (у.е.)	1	2	3	4	4	3	2	1
Вероятность непоражения	0,9	0,9	0,9	0,9	0,9	0,9	0,9	0,9

Необходимо минимизировать затраты на резервное оборудование СДР, обеспечив требуемое значение нижней границы структурной живучести ТКС НГСЖ_{тп} = 0,9950. Следовательно, требуется решить прямую задачу оптимального резервирования.

2. Оценка живучести ТКС

Для каждого заданного требования φ_{st} формируется множество путей m_{st} для его обслуживания. Полученный результат представляется в дизъюнктивной нормальной форме ДНФ. Так, для требования φ_{14} получим множество путей (17) [6]:

$$m_{14} = abc + dhg + dec + dhfc + defg + abfg + abchg. \quad (17)$$

Здесь символ «+» используется для обозначения операции дизъюнкции.

Из множества полученных путей для каждого требования выбираются допустимые пути (соответствующие принципам 1 – 4). Учитывая заданные значения вероятностей непоражения ветвей (табл. 1), считая допустимыми пути, ранг которых $r \leq 3$, получим в результате из множества (17) следующее множество допустимых путей (18):

$$m_{14}^{r \leq 3} = abc + dhg + dec. \quad (18)$$

(Ранг r пути – это число ветвей, образующих путь.)

Аналогично для всех остальных заданных требований сформируем множество допустимых путей в ДНФ (19):

$$\begin{cases} m_{36}^{r \leq 3} = e+fh+cgh+bad \\ m_{35}^{r \leq 3} = f + cg + eh \\ m_{42}^{r \leq 3} = cb + gfb \end{cases} \quad (19)$$

Для расчета показателя структурной живучести – НГСЖ – для полученных допустимых путей следует определить разделяющие их множества сечений. Множество сечений формируется как двойственная булева функция относительно множества путей, а именно: все операции логического сложения заменяются на операции логического умножения и наоборот – все операции логического умножения заменяются на операции логического сложения. В результате этих преобразований формируется конъюнктивная нормальная форма КНФ. Затем все скобки в КНФ раскрываются с использованием тождеств и законов булевой алгебры и выражение функции из КНФ преобразуется в ДНФ (но уже для множества сечений) [6]. Среди полученного множества сечений выбираются те, которые соответствуют принципам 1 – 4, т.е. сечения минимального ранга (с минимальным числом образующих их ветвей).

Отметим, что в данном примере заданы одинаковыми значения p_{xy} – вероятности непоражения ветвей β_{xy} , следовательно, выбор как путей, так и сечений осуществляется по рангу, однако, в случае различных значений p_{xy} допустимыми следует считать пути и сечения с максимальным значением вероятности

непоражения. При этом вероятность непоражения P_{st}^k k -го пути μ_{st}^k из пункта s в пункт t рассчитывается на основе выражения (20):

$$P_{st}^k = \prod_{\beta_{xy} \in \mu_{st}^k} p_{xy} . \quad (20)$$

Вероятность непоражения Π_{st}^l l -го сечения σ_{st}^l рассчитывается в соответствии с выражением (1).

В данном примере для множеств допустимых путей $m_{14}^{r \leq 3}$, $m_{36}^{r \leq 3}$, $m_{35}^{r \leq 3}$, $m_{42}^{r \leq 3}$ получим множества (21) допустимых сечений ранга $r \leq 3$:

$$\begin{cases} \sigma_{14}^{r \leq 3} = cd + ch + cg + ad \\ \sigma_{36}^{r \leq 3} = aeh + deh \\ \sigma_{35}^{r \leq 3} = fce + fch + fge + fgh \\ \sigma_{42}^{r \leq 3} = b + cg + cf \end{cases} \quad (21)$$

В соответствии с данными табл. 1 выполним расчет вероятности непоражения полученных множеств допустимых сечений на основе выражения (2). Для множества сечений $\sigma_{14}^{r \leq 3}$ получим $\pi_{14} = 0.9606$, для множества сечений $\sigma_{36}^{r \leq 3}$ – $\pi_{36} = 0.9980$. Далее получим: $\pi_{35} = 0.9960$; $\pi_{42} = 0.8821$. Рассчитаем показатель структурной живучести сети в соответствии с выражением (3): НГСЖ = 0.9512 (при расчете НГСЖ значения всех весов приняты: $w_{st} = 0,25$).

Поскольку условие (8) получения оптимального решения не выполняется, – полученное значение НГСЖ менее требуемого: $0.9512 < 0,9950$, – осуществляется переход с следующему этапу расчетов.

3. Обеспечение структурной живучести ТКС

Выполнение данного этапа метода осуществляется в постановке задачи (7) при ограничении (условии) (8). Обеспечение структурной живучести ТКС осуществляется с использованием градиентного метода на основе расчетов в соответствии с выражением (14).

Результаты первого шага расчетов, на основе которых выбирается участок сети (ветвь), подключение к которому резерва обеспечивает максимальную эффективность шага на основе (14), представлены в табл. 2.

В Табл. 2 в столбцах 2 – 5 шага «0» указаны значения вероятностей непоражения множеств сечений (21), в столбцах 6, 7 и 8 – рассчитанное значение НГСЖ, стоимость вводимого резервного оборудования

(равная нулю для шага «0») и значение приращения показателя структурной живучести, соответственно. Для шага «1» представлены расчеты изменения каждого из представленных показателей при введении резервного оборудования на каждом из участков (a, b, \dots, h). В результате формируется столбец 8 значений приращения показателя структурной живучести, максимальное значение элемента которого и определит выбор оптимального направления процесса поиска. Как видим, на первом шаге выбирается участок « b ». Резервирование обеспечивает НГСЖ = 0.9812. Поскольку $0.9812 < 0.9950$, процесс продолжается до получения оптимального решения, т.е. до выполнения условия (8). Результаты последующих расчетов представлены в табл. 3.

В результате получен вектор оптимальной структуры резерва:

$$M(m_a, m_b, m_c, m_d, m_e, m_f, m_g, m_h) = (1, 1, 1, 0, 0, 0, 0, 0).$$

Это означает, что следует добавить по одному резервному элементу на участках a, b и c , что обеспечивает значение показателя живучести НГСЖ = 0,9952. При этом минимально возможная стоимость резервного оборудования составляет СДР = 2 + 3 + 1 = 6 у.е.

В случае решения задачи с ограниченной стоимостью резервного оборудования, т.е. когда задается СДР_{пр}, может оказаться, что полученное значение НГСЖ не удовлетворяет существующим требованиям к структурной живучести ТКС. В этом случае, на основе продолжения процесса резервирования, создается возможность выработки необходимых рекомендаций относительно изменения исходных данных.

Таблица 2. Результаты расчетов первого шага обеспечения требуемого значения НГСЖ

№ шага	П ₁₄	П ₃₆	П ₃₅	П ₄₂	НГСЖ	С	$\gamma(m_i+1)$
1	2	3	4	5	6	7	8
0	0.9606	0.9980	0.9960	0.8821	0.9512	0	0
1							
a	0.9693	0.9989	0.9960	0.8821	0.9616	1	0.0025
b	0.9606	0.9980	0.9960	0.9703	0.9812	2	0.0115
c	0.9870	0.9980	0.9978	0.8982	0.9703	3	0.0039
d	0.9781	0.9989	0.9960	0.8821	0.9638	4	0.0012
e	0.9606	0.9998	0.9978	0.8821	0.9601	4	0.0002
f	0.9606	0.9980	0.9996	0.8901	0.9621	3	0.001
g	0.9693	0.9980	0.9978	0.8901	0.9638	2	0.0024
h	0.9693	0.9998	0.9978	0.8821	0.9623	1	0.0032

Таблица 3. Последовательность шагов обеспечения структурной живучести системы

№ шага	П ₁₄	П ₃₆	П ₃₅	П ₄₂	НГСЖ	С	$\gamma(m_i+1)$
0	0.9606	0.9980	0.9960	0.8821	0.9592	0	0
1(b)	0.9606	0.9980	0.9960	0.9703	0.9812	2	0.0115
2(c)	0.9870	0.9980	0.9978	0.9880	0.9927	2+3	0.0039
3(a)	0.9960	0.9989	0.9978	0.9880	0.9952	2+3+1	0.0025

Заключение

Представлен метод обеспечения структурной живучести ТКС на основе предложенного показателя живучести – нижней границы структурной живучести, обеспечивающей гарантированное устойчивое функционирование ТКС. Предложен способ определения нижней границы структурной живучести. Задача обеспечения структурной живучести ТКС сформулирована как задача математического программирования. Представлен метод достижения требуемого значения нижней границы структурной живучести на основе введения структурной избыточности путем использования оптимизационной процедуры резервирования ветвей сечений, выбор которых определяется на основе максимизации эффективности каждого этапа резервирования. Решение оптимизационной задачи введения структурной избыточности при заданных ограничениях либо позволяет обеспечить требуемый показатель структурной живучести ТКС, либо обеспечивает выработку рекомендаций по: изменению значений элементов матрицы вероятностей непоражения ветвей сети на вычисленную в результате решения задачи величину; изменению классов (приоритетов) и «весов» определенных требований; ограничению величины поступающих в сеть требований; изменению задаваемых предельно допустимых значений нижней границы структурной живучести НГСЖтр ТКС – при решении прямой задачи оптимального резервирования или предельной стоимости дополнительного резервного оборудования СДРпр – при решении обратной задачи оптимального резервирования. Предложенный метод может быть использован в случае, когда множества допустимых путей и допустимых сечений являются множества всех возможных путей и всех возможных сечений в ТКС.

Выполнен пример реализации предложенного метода обеспечения структурной живучести телекоммуникационной сети.

Благодарности

Настоящая работа была выполнена при поддержке интернационального проекта ITNEA XXI Института информационных теорий и их приложений FOI ITNEA и Ассоциации ADUIS Украина (Ассоциация разработчиков и пользователей интеллектуальных систем), в связи с чем автор выражает свою искреннюю благодарность.

Литература

1. Рекомендация ITU-T Y.3001: Глобальная информационная инфраструктура, аспекты протокола интернет и сети последующих поколений [Электронный ресурс] / ITU: Committed to connecting the world.— Режим доступа: [www/ http://www.itu.int/](http://www.itu.int/) — 10.03.2012 г. — Загл. с экрана.
2. Стекольников Ю.И. Живучесть систем – СПб.: Политехника, 2002. – 155с.
3. Додонов А.Г., Ландэ Д.В. Живучесть информационных систем. – К.: Наук. думка, 2011. – 256 с.
4. Синтез и анализ живучести сетевых систем : монография / Ю.Ю. Громов, В.О. Драчев, К.А. Набатов, О.Г. Иванова. – М. : «Изд-во Машиностроение-1», 2007. – 152 с.
5. Князева Н.О. Повышение структурной живучести телекоммуникационной сети., International Journal Information models and analyses, 2013.– V2, №3. – P. 275-284.
6. Князева Н.О. Теорія проектування комп'ютерних систем і мереж. Ч.2. Методи аналізу і синтезу комп'ютерних систем і мереж, Одеса: СПД, 2012. – 240 с.

Authors' Information



Нина Князева – Факультет информационных технологий и кибербезопасности Одесской национальной Академии пищевых технологий, профессор, заведующая кафедрой информационных систем и сетей; ул. Дворянская, 1/3, Одесса-26, 65026, Украина; тел. (048)-7209176; моб. (38067)-430-17-28; e-mail: knyazeva@ukr.net.

Главные области научного исследования: Управление в телекоммуникационных сетях, проблемы повышения надежности и живучести телекоммуникационных сетей

The Method of the Ensurance of Structural Survivability of Telecommunication Network

Nina Kniazieva

Summary: One of the most important properties of telecommunication networks to ensure their effective functioning is survivability. To ensure the survivability of telecommunication networks it's required to solve a number of scientific and engineering problems at all stages of their life cycle – design, modernization or improvement, operation. Questions of ensurance of survivability of telecommunication networks are becoming more actual due to the intensive development of telecommunications in the past decade. The synthesis problems of survivability systems, evaluation, ensurance and improvement of structural survivability of the systems, as determined by the needs of practice acquired particular actuality today. The aim of this work is to develop a method of ensurance of structural survivability of telecommunication network (TCN), which implements the streaming approach using the proposed index of structural survivability TCN defined lower boundary of structural survivability, providing a guaranteed stable functioning TCN, the achievement of the desired value which is performed by the introduction of structural redundancy (as the property of survivability possess only redundant systems) using the optimization procedure of the sections branches reservation, the choice of which is determined on the basis of maximizing the effectiveness of each stage of the reservation. The optimization of value of the structural survivability of TCN is to achieve its maximum value at the existing restrictions. The method to ensure structural survivability TCN, based on the formation of sets of admissible routes for each service requirements, for which are formed the sets of admissible sections is developed. Are determined the probable values of unaffacting each section, and for each set of sections, which are formed on the basis of the lower boundary of the structural survivability TCN obtained as an average value of probable unaffacting sections. The paper proposes a developed the method of the ensurance of structural survivability of TCN, which is designed for procedural model, which includes the execution of the sequence of steps: 1) formation of the source data, 2) the evaluation of the structural survivability of TCN, 3) the ensurance structural survivability of TCN, 4) the development of solutions for the structural survivability of TCN. For each stage, are formulated the objectives and are determined ways to decide them. For stage 1) the list and the method to present the initial data are determined; for stage 2) are determined ways of forming sets of admissible routes and obtaining for them sets of

admissible sections, for which are given methods for calculating the probabilities of unaffecting of each sections and the probabilities of unaffecting of all admissible sections which separate points s and t ($s, t = \overline{1, n}, s \neq t, n$ – number of network points), is formulated a condition for the lower bound of the required structural survivability of TCN. In stage 3) optimization problem is formulated to ensure the required value of the lower limit of the structural survivability of TCN as a mathematical programming problem, is shown the feasibility of using the gradient method to solve it, a method of calculating the efficiency of each step of the optimization process. Stage 4) is final. At this stage is drawn the conclusion that the obtained solution of the optimization problem to introduce structural redundancy with in sets of limits or provides the required value of the index of structural survivability TCN or provides the procedure of recommendations on: changing the value of the matrix elements of the probability of unaffecting branches of the network which is computed value by solving the problem; the change of classes (priorities) and the "weights" of certain requirements; limiting values of network requirements, the change of defined limit values, the lower boundary of the structural survivability TCN. The proposed method can be used in the case where the sets of admissible routes and admissible sections are the sets of all possible routes and all possible sections in the TCN.

An example of implementation of the proposed method to ensure structural survivability of the telecommunications network is made.

РАСПОЗНАВАНИЕ ОБЪЕКТОВ С ИСПОЛЬЗОВАНИЕМ КРИТЕРИЯ НА ОСНОВЕ ВЕКТОРНОЙ МЕРЫ БЛИЗОСТИ ОБРАЗОВ В ПРОСТРАНСТВЕ ОШИБОК

Петр Четырбок

Аннотация: В данной работе решается проблема построения отображения множества распознаваемых образов (векторов параметров образов) на множество векторов ошибок распознавания образов нейронную сеть, которое позволяет связать классификацию образов с анализом векторов в пространстве ошибок. Цель статьи – построение функционала и решающего правила (векторного критерия для распознавания образов) для решения задачи классификации образов. Критерий позволяет группировать образов, ихное распознавание и сравнение. Векторный критерий формулируется на базе близости образов в пространстве ошибок. Предложен алгоритм перехода от пространстве параметров изображений сигналов в пространстве ошибок распознавания образов. Корректность полученных научных результатов, выводов и рекомендаций данной работы была подтверждена результатами экспериментальных исследований на базе разработанной универсальной системы интеллектуального анализа данных, которая решает задачу распознавания электрооптических изображений NEFFClass BGC GG. Эксперименты сделаны на базе «Института анализа прикладных систем» НТУУ «КПИ». Результаты, полученные в работе наглядно демонстрируют эффективность использования разработанных моделей, методов и алгоритмов для решения задач распознавания сигналов

Ключевые слова: neural networks, basic method, method of accreditation, vector criterion

Введение

Для нейронных сетей актуальным является модификация методов обучения с использованием в качестве критерия близости образов скалярной величины, равной сумме норм Евклида, обобщенной нормы Евклида и нормы Чебышева. Актуальным является также исследование эффективности модифицированных методов в решении задач с трудно распознаваемыми образами сигналов в условиях помех.

Анализ литературы. Из известных методов обучения нейронных сетей наиболее широкое применение имеют градиентные методы со случайным изменением начальных условий [Наукин, 2006]. Недостатком этих методов является трудности распознавания образов в случае близости по норме Евклида сравниваемых образов. Существует много эвристических алгоритмов классификации без учителя [Наукин, 2006], основанных на использовании мер близости между объектами. Каждый из них имеет свою область применения, а наиболее распространенным недостатком является отсутствие четкой формализации задачи: совершается переход от идеи кластеризации прямо к алгоритму, в результате неизвестно, что ищется.

1) *Базовый способ:* для вектора данных x^i и каждого ядра a_i вычисляется $y_i = d(x, a_i)$. По правилу „победитель забирает все” строка ответов y_i преобразуется в строку, где только один элемент, соответствующий максимальному y_i , равен 1, остальные - нули. Эта строка и является результатом

функционирования сети. По ней может быть определен номер класса (номер места, на котором стоит 1) и другие показатели.

2) *Метод аккредитации*: за слоем элементов базового метода, выдающих сигналы 0 или 1 по правилу "победитель забирает все" (далее называем его слоем базового интерпретатора), надстраивается еще один слой выходных сумматоров. С каждым (i -м) классом ассоциируется q -мерный выходной вектор z^i с координатами z_j^i . Он может формироваться по-разному: от двоичного представления номера класса до вектора ядра класса. Вес связи, ведущей от i -го элемента слоя базового интерпретатора к j -му выходному сумматору определяется в точности как z_j^i . Если на этом i -м элементе базового интерпретатора получен сигнал 1, а на остальных – 0, то на выходных сумматорах будут получены числа z_j^i .

3) *Нечеткая классификация*. Пусть вектор данных x обработан слоем элементов, вычисляющих $y_i = d(x, a_i)$. Идея дальнейшей обработки состоит в том, чтобы выбрать из этого набора $\{y_i\}$ несколько самых больших чисел и после нормировки объявить их значениями функций принадлежности к соответствующим классам. Предполагается, что к остальным классам объект наверняка не принадлежит. Для выбора семейства G наибольших y_i определим следующие числа:

$$y_{\max} = \max \{y_i\}, M_y = \frac{1}{k} \sum_i y_i, s = (1 - \alpha)M_y + \alpha y_{\max},$$

где число α характеризует отклонение "уровня среза" s от среднего значения M_y , $\alpha \in [-1, 1]$, по умолчанию обычно принимается $\alpha = 0$.

Множество $J = \{i | y_i \in G\}$ трактуется как совокупность номеров тех классов, к которым может принадлежать объект, а нормированные на единичную сумму неотрицательные величины

$$f_i = \frac{y_i - s}{\sum_{j \in J} (y_j - s)} \quad (\text{при } i \in J \text{ и } f = 0 \text{ в противном случае})$$

интерпретируются как значения функций принадлежности этим классам.

4) Метод интерполяции надстраивается над нечеткой классификацией аналогично тому, как метод аккредитации связан с базовым способом. С каждым классом связывается q -мерный выходной вектор z^i . Строится слой из q выходных сумматоров, каждый из которых должен выдавать свою компоненту выходного вектора. Весовые коэффициенты связей, ведущих от того элемента нечеткого классификатора, который вычисляет f_i , к j -му выходному сумматору определяются как z_j^i . В итоге вектор выходных сигналов сети есть

$$z = \sum_i f_i z^i.$$

В отдельных случаях по смыслу задачи требуется нормировка f_i на единичную сумму квадратов или модулей.

5) Способ построения векторного критерия для распознавания образов. Строится отображение множества распознаваемых образов на множество векторов ошибок распознавания образов нейронной сетью, которое позволяет связать классификацию образов с анализом векторов в пространстве ошибок.

Выбор одного из описанных пяти вариантов использования сети определяется нуждами пользователя. Предлагаемые пять способов покрывают большую часть потребностей. В работе разработан пятый метод.

Постановка проблемы

Построить отображение множества распознаваемых образов (векторов параметров образов) на множество векторов ошибок распознавания образов нейронной сетью, которое позволяет связать классификацию образов с анализом векторов в пространстве ошибок.

Цель статьи – построение функционала и решающего правила (векторного критерия для распознавания образов) для решения задачи классификации образов. Функционал равен скалярному произведению векторов ошибок при распознавании нейронной сетью образов и соответствующих им эталонов. Впервые построено решающее правило для классификации образов в виде утверждения: каждому образу, распознаваемому многослойным персептроном в многофакторном пространстве ошибок будет соответствовать свое значение функционала и образ ближе к эталону, чем больше $\cos(\lambda)$

$$\cos(\lambda) = \frac{(\overline{E}, \overline{X})}{\|\overline{E}\|_c \|\overline{X}\|_c},$$

где E - вектор ошибок в пространстве ошибок, полученный при распознавании нейронной сетью входного образа, X - вектор ошибок, полученный при распознавании нейронной сетью эталона. Что позволяет связать классификацию распознаваемых образов с поведением функционала в многофакторном пространстве ошибок.

Решение проблемы

Формулировка векторного критерия близости образов в пространстве ошибок. Переход из пространства параметров в пространство ошибок. Для распознавания образов построим функционал

$$F = E_2 x_1 + E_3 x_2 + E_4 x_3, \quad (1)$$

E_2 – среднеквадратическая ошибка, полученная при распознавании образа:

$$E_2 = \frac{1}{n} \sqrt{\sum_j^n (y_j - d_j)^2}$$

где y_j – реальное выходное состояние нейрона j выходного слоя нейронной сети при подаче на ее входы образа; d_j – идеальное (желаемое) выходное состояние этого нейрона. Суммирование ведется по всем нейронам выходного слоя n .

E_3 – линейная ошибка сети, полученная как сумма модулей поразрядного отклонения образа от эталона:

$$E_3 = \frac{1}{n} \sum_j^n |y_j - d_j|$$

E_4 – максимальная ошибка поразрядного отклонения образа от эталона:

$$E_4 = \max_{j=1..n} |y_j - d_j|$$

(x_1, x_2, x_3) – вектор ошибок при распознавании эталонного сигнала.

Для каждого из образов получили свое значение функционала, т.е. функционал однозначно определяет образ, запомненный сетью. Если для двух образов величины E_2 , E_3 , E_4 совпадут, то функционалы для этих образов будут различны. Если в качестве образов использовать волны, то первый член функционала отвечает за отклонение частоты образа от эталонной, второй член отвечает за отклонений

по амплитуде волны образа от эталонной, третий член анализирует пиковые всплески. Функционал имеет нижнюю границу при $F = 0$. Для образов волновой природы функционал позволяет выделить вредные гармоники, подавить шумы и выделить полезный сигнал.

При распознавании сигнала нейронной сетью вычисляем функционал. При n эталонных образах сигналов мы можем получить n функционалов F_1, F_2, \dots, F_n , где n – количество сигналов, которые распознает сеть. Как же определить какому образу соответствует входной сигнал или он не распознается сетью?

В соответствии с введенным функционалом (1) минимизируемой целевой функцией ошибки нейронной сети является величина:

$$E(w) = \frac{1}{2} \sum_{j,p} (y_{j,p}^{(N)} - d_{j,p})^2$$

где $y_{j,p}^{(N)}$ – реальное выходное состояние нейрона j выходного слоя N нейронной сети при подаче на ее входы p -го образа; d_{jp} – идеальное (желаемое) выходное состояние этого нейрона.

Суммирование ведется по всем нейронам выходного слоя и по всем обрабатываемым сетью образам. Минимизация ведется методом градиентного спуска [Накин, 2006]. При распознавании образа нейронной сетью вычисляем функционал, используя весовые коэффициенты, полученные при обучении сети. Функционал F равен скалярному произведению векторов ошибок (E_2, E_3, E_4) и (x_1, x_2, x_3) .

Найдем косинус угла между векторами ошибок:

$$\cos(\lambda) = \frac{(\bar{E}, \bar{X})}{\|\bar{E}\|_c \|\bar{X}\|_c}$$

Аналогично находим косинусы для всех эталонных образов F_1, F_2, \dots, F_n , где n – число эталонных образов предъявленных сети при обучении. Тот функционал F_i , для которого $\cos(\lambda)$ максимален определяет соответствующий входному сигналу эталонный образ. Для расчета взвешенного критерия используется взвешенная норма в пространстве ошибок.

Проведем эксперименты для градиентного алгоритма с использованием взвешенного критерия на основе векторов ошибок при распознавании областей на электрооптических изображениях. Закодируем базу правил с помощью кода Грея (0 - 000, 1 - 001, 2 - 011, 3 - 010, 4 - 110) (Таблица 1).

Используя 54 образов (по 6 с каждого класса) научим двухслойный персептрон распознавать образцы согласно базе правил. Остальные 45 образцов будут использованы для тестирования.

$$F = (\bar{E}, \bar{X}) = \cos(\lambda) \|\bar{E}\|_c \|\bar{X}\|_c$$

Проведены эксперименты на реальных данных и установлено, что использование нейро-нечеткого классификатора NEFClass целесообразно применять для распознавания областей на электрооптических изображениях. При тестировании были ошибочно классифицированы 4 образца с проверочного набора данных [Zaichenko, 2008]. Но для градиентного метода с использованием взвешенного критерия на основе векторов ошибок уровень правильной классификации на проверочной части выборки выше, а именно ошибочно классифицирован 1 образец с проверочного набора данных (Таблица 2).

Таблица 1. База правил

№ правила	Код признаков	№ Класс
1	1 110 110 110 110	0
2	1 110 000 001 110	1
3	1 110 000 000 110	1
4	1 110 001 000 110	1
5	1 011 010 001 001	2
6	1 001 000 001 000	3
7	1 110 110 001 110	4
8	1 010 110 010 010	5
9	1 010 010 011 010	5
10	1 110 110 010 010	5
11	1 000 000 000 000	6
12	1 010 011 001 011	7
13	1 001 000 000 001	8
14	1 001 001 000 001	8
15	1 001 000 000 000	8

Таблица 2. Сравнительная таблица для алгоритмов обучения весовых коэффициентов

Алгоритм обуч. весовых коэфф.	Обучение		Тестирование	
	Ср.кв.ошибка	Ошибка класс-ии	Среднекв.ошибка	Ошибка класс-ии
Классический	6,650668	0	7,285827	4
Градиентный	5,9893	0	6,829068	4
Сопр. градиента	1,132871	0	3,314763	4
Генетический с треугольной ФП	11,110936	0	13,677424	4
Генетический с гаусовской ФП	3,204446	0	4,568338	4
Градиентный с исп. взвешен. критерия	3,456378	0	4,435278	1

Выводы

Каждому образу, распознаваемому многослойным персептроном в многофакторном пространстве ошибок соответствует свой вектор ошибок. Впервые построено решающее правило для классификации образов в виде утверждения: каждому образу, распознаваемому многослойным персептроном в многофакторном

пространстве ошибок будет соответствовать свой вектор ошибок и образ ближе к эталону, чем больше $\cos(\lambda)$.

$$\cos(\lambda) = \frac{(\overline{E}, \overline{X})}{\|\overline{E}\|_c \|\overline{X}\|_c},$$

где E - вектор ошибок в пространстве ошибок, полученный при распознавании нейронной сетью входного образа, X - вектор ошибок, полученный при распознавании нейронной сетью эталона. Предложенный в статье функционал и решающее правило (взвешенный критерий на основе векторов ошибок для распознавания образов) позволяют улучшить распознавания областей на электрооптических изображениях.

Литература

[Haykin, 2006] Haykin, "Simon Neural Networks: A Comprehensive Foundation", Second Edition, Prentice Hall 2006, 1104 p (in Russian)

[Zaychenko, 2008] Zaychenko, Y.P., "Fuzzy models and methods in intelligent systems", K.: Publishing house „Word”, 2008, 344 p (in Russian)

Информация об авторах

Петр Четырбок - аспирант НТУУ "КПИ", 03056, Киев-56, Украина, e-mail: petr58@ukr.net

Pattern recognition using a criterion based on vector criterion of patterns proximity in the space of errors

Petro Chetyrbok

Abstract: In the given work displaying is built of great number of appearances on the great number of vectors of errors of pattern recognition by the neuron network, which allows to link classification of appearances with the analysis of vectors in space of errors. A criterion allows grouping appearances, recognizing and comparing them. A vector criterion is formulated of closeness of signal images in space of errors. An algorithm is offered for transition from space of parameters of signal images in space of errors of pattern recognition. An optimum decided rule is built for classification of signal images of signals with the use of weighed criterion of closeness of recognizable signal images in space of errors of recognition. Authenticity of the received scientific results, conclusions and recommendations of this thesis work has been confirmed by the results of experimental researches of the developed universal system of intellectual data analysis, which solve the task of recognition of objects of the electro-optical images NEFFClass BGCGG (Basic Gradient Conjugate Gradient Genetic), conducted on the base of „Institute of the Applied Systems Analysis” NTUU „KPI”. The results received in work evidently demonstrate efficiency of the use of the developed models, methods and algorithms for the solution of tasks of recognition of signals.

Keywords: neural networks, basic method, method of accreditation, vector criterion

ОБЕСПЕЧЕНИЕ БЕЗОПАСНОСТИ ИСПОЛЬЗОВАНИЯ БАЗ ДАННЫХ В УСЛОВИЯХ ЧРЕЗВЫЧАЙНЫХ СИТУАЦИЙ

Александр Кузёмин, Алексей Василенко

Аннотация: Рассматриваются проблемы уменьшения риска, увеличения надежности и безопасности хранения и использования баз данных (БД) в условиях чрезвычайных ситуаций (ЧС). Предлагается технологическая последовательность решения поставленной задачи. Для анализа данных учитывается особенность представления количественных и качественных параметров в условиях ЧС. Показана возможность логического и функционального ER моделирования CASE технологии с учетом минимизации риска жизнедеятельности в зоне ЧС. Приведено техническое решение поставленной задачи.

Ключевые слова: Чрезвычайные ситуации, База Данных (БД), архив, резервирование, система управления базой данных (СУБД), Сервер.

Постановка задачи исследования

На сегодняшний день информационные системы (общегосударственные, региональные, ведомственные и др. ситуационные центры) в условиях опасности возникновения чрезвычайных ситуаций (ЧС) играют важную роль при обеспечении безопасности жизнедеятельности населения [Kuzomin & Torojev, 2006]. При возникновении риска возникновения ЧС и при ликвидации последствий ЧС необходимо сохранить, обработать и использовать жизненно важную информацию для принятия управленческих решений. На сегодняшний день информационные системы (общегосударственные, региональные, ведомственные и др. ситуационные центры) в условиях опасности возникновения ЧС играют важную роль при обеспечении безопасности жизнедеятельности населения. При возникновении риска возникновения ЧС и при ликвидации последствий ЧС необходимо сохранить, обработать и использовать жизненно важную информацию для принятия управленческих решений. Однако ЧС порождают и риски потери информации – данных, которые определяют надежность и живучесть баз данных в условиях ЧС.

Повышение риска уменьшения надежности и безопасности хранения и использования БД связано с возникновением в условиях ЧС аппаратных и программных сбоев. Риски уменьшения надежности баз данных (БД) определяются соответствующими источниками отказа (Рис. 1). В условиях ЧС необходимо обеспечить живучесть ситуационного центра и в частности БД. Под живучестью БД следует понимать свойство БД выполнять своё функциональное назначение с минимальным риском и максимальной надежностью в условиях ЧС.

Целью данной статьи является представление результатов разработки информационной технологии, которая направлена на уменьшения риска отказа БД, повышение надежности и живучести баз данных ситуационных центров в условиях ЧС.

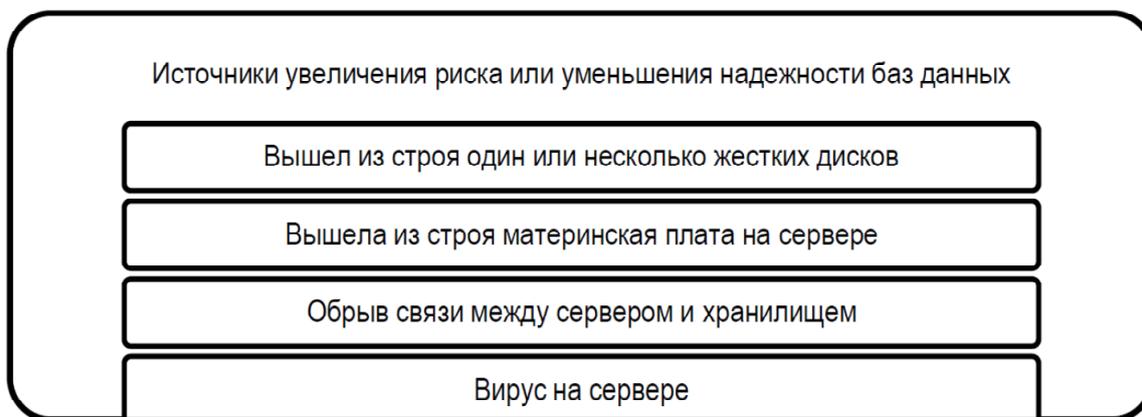


Рисунок 1. Источники увеличения риска или уменьшения надежности баз данных

При этом можно определить следующую технологическую последовательность решения поставленной задачи:

1. Техническое задание на разработку БД.
2. Анализ качества данных для ситуационного центра.
3. Выбор архитектуры программно-аппаратного комплекса ситуационного центра [1].
4. Разработка модели управления БД (СУБД).
5. Оценка рисков и надежности БД в условиях ЧС.
6. Возврат к пункту 3, если допустимые значения оцениваемых критериев превышаются.
7. Минимизировать коэффициент цена/риск или цена/надежность.
8. Возврат к пункту 3, если допустимые значения оцениваемых критериев превышаются.
9. Окончательный выбор архитектуры в соответствии с требованиями ТЗ.

Рассмотрим ключевые этапы предлагаемой технологии.

Анализ качества данных для ситуационного центра.

Для проектирования ситуационного центра необходимо выполнить системный анализ данных, который в первую очередь определяет проектные решения для БД. При проектировании баз данных доминирующее значение приобретают сами данные, их хранение и обработка. Однако их содержание для применения к качеству баз данных требуется уточнить. Выделяемые показатели качества должны иметь практический интерес для пользователей и быть упорядочены в соответствии с приоритетами практического применения. Для оценивания качества информации, которая должна использоваться в ситуационном центре может применяться общий подход к выделению адекватной номенклатуры стандартизированных в ISO 9126 базовых характеристик и субхарактеристик [Липаев, 2001]. Каждый выделяемый показатель качества должен быть пригоден для достоверного экспертного оценивания или измерения, а также для сравнения с требуемым значением.

Мерой качества функциональной пригодности БД (Рис. 2) может быть степень покрытия целей ситуационного центра, назначения и функций в доступной пользователям форме.

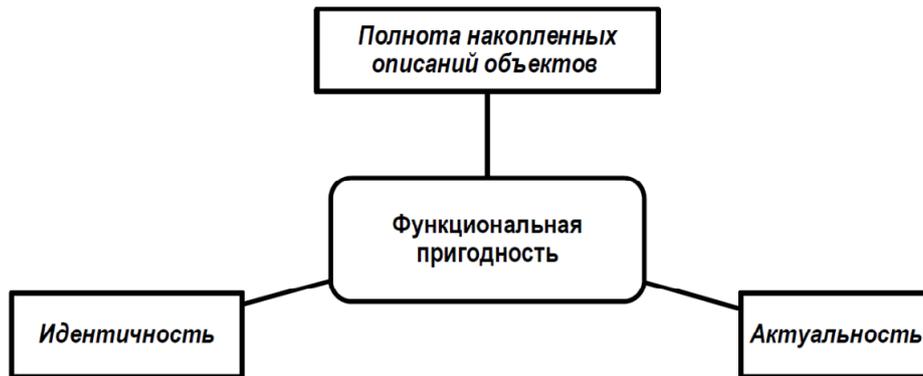


Рисунок 2. Проблема функциональной пригодности

Корректность или достоверность данных — это степень соответствия данных об объектах в базах данных реальным объектам в зоне возникновения ЧС, в данный момент времени, определяющаяся изменениями самих объектов, некорректностями записей об их состоянии или некорректностями расчетов их характеристик. Сюда же можно отнести и некоторые объемно-временные характеристики сохраняемых и обрабатываемых данных (Рис. 3).



Рисунок 3. Объемно-временные характеристики БД

Защищенность информации реализуется средствами СУБД в сочетании с поддерживающими их средствами защиты данных. Цели, назначение и функции защиты тесно связаны с особенностями функциональной пригодности каждой базы данных. В распределенных БД показатели защищенности тесно связаны с характеристиками целостности и отражают степень тождественности одинаковых данных в памяти удаленных компонентов ситуационного центра.

Кроме окружающей среды, в ситуационном центре учитывается ресурс. Этот ресурс определяется средой технических и организационных средств, который может использовать лицо принимающее решение (ЛПР) для предупреждения или ликвидации последствий ЧС. Среди отмеченных выше данных имеются количественные и качественные параметры. Количественные параметры в БД традиционно представляются в виде таблиц. Для использования БД в ситуационном центре выполняется статистический анализ данных, который позволяет выделить микроситуации, которые соответствуют наиболее влияющим параметрам окружающей среды в зоне действия ЧС [Kuzemin & Lyashenko, 2007; Kuzemin et al, 2007]. Микроситуации для прошедших ЧС могут быть использованы для наполнения прецедентной базы количественных данных, для которых выделены наилучшие решения в виде качественных параметров. Качественные параметры представляют любое решение или действие ЛПР, которое имеет три основные характеристики: цель действия; описание действия; средство его выполнения.

Все эти характеристики можно получить из ответов специалистов и экспертов, которые могут квалифицированно ответить на следующие три вопроса: "Какой должен быть результат?" – ответ будет характеризовать цель (назначение действия); "Что делать?" - ответ будет давать описание действия; "Как делать?" - ответ будет характеризовать средство выполнения (умение и возможность). Ставятся три вопроса, на которые необходимо ответить эксперту или ЛПР – «КТО», «ЧТО» и «КАК».

Для предлагаемой технологии разработки ситуационного центра для подсистемы ликвидации лавинной опасности рассмотрим пример «Ликвидация последствий схода лавины». В результате анализа данных (регрессионного, дисперсионного и кластерного анализа) были выявлены наиболее влияющие параметры (3 – 5), которые определялись как микроситуации с наибольшей вероятностью (0.9 – 0.95) приводили к удачным или рациональным решениям при возникновении или ликвидации последствий ЧС [Куземин & Сорочан, 2004]. При возникновении ЧС принимались соответствующие управляющие решения, которые можно было бы оценивать по своей эффективности для обеспечения наименьших рисков жизнедеятельности, социальных и материальных потерь. Такие удачные решения составляли основу прецедентной БЗ и используются при поиске управляющих решений с минимальными рисками и минимальным временем на выполнение. Для прецедентной БЗ были установлены соответствующие понятия, категории, отношения и сценарии действий для предупреждения и ликвидации последствий ЧС [Диги, 2005].

В процессе фиксации ситуации и ликвидации последствий ЧС для рассматриваемого примера на вопрос «КТО» можно выделить таких «Исполнителей»: гражданин, заметивший возникновение ЧС, сотрудники снеголавинной службы министерства по чрезвычайным ситуациям (МЧС), диспетчер, ответственное лицо службы МЧС, эксперты МЧС. На вопрос «ЧТО» могут использоваться «Документы» для описания действий: метеорологическая сводка, заявка о происшествии, лавиноопасная экспертиза, рекомендации по проведению протилавиных мероприятий, акт о последствиях.

«Действия участников при ликвидации последствий ЧС» могут носить предметный характер: сообщение, фиксация, ликвидация, регистрация последствий.

Проблемная, контролируемая «Микроситуация на языке представления ситуаций» выглядит следующим образом:

- микроситуация с порядковым номером;
- гражданин замечает физические изменения в снежном покрове;
- сотрудники снеголавинной станции объявляют штормовое предупреждение;
- сотрудники снеголавинной станции сообщают диспетчеру службы МЧС;
- диспетчер выясняет место расположение возможного схода лавины;
- диспетчер передает первоначально обработанную информацию совету экспертов;
- эксперты производят анализ и принимают решения;
- сотрудники служба МЧС реализуют выработанные рекомендации относительно лавинной опасности;
- эксперт службы МЧС оценивает последствия происшествия.

Понятия, выявленные для данной ситуации распределяются в соответствующих иерархиях категорий таким образом, что бы можно было отобразить дерево аналогичных функциональных структур для ликвидации последствий ЧС.

Подсистема службы МЧС для ликвидации лавинной опасности на первом этапе принятия решений может быть определена как регистрационная структура системы управления, которая по функциональному назначению аналогична соответствующей структуре МЧС в пожарной части.

Категория «Документы» отражает в документах действия, которые требуются при описании управленческих действий для ликвидации лавинной опасности, в иерархии категории. «Метеорологическая сводка» соответствует перечню документов, который соответствует подключению всех необходимых служб обеспечения безопасности жизнедеятельности. «Лавинная экспертиза» необходима для отображения причин схода лавины и установления возможных противолавинных сооружений. Акт о последствиях помогает определить величину материального ущерба, вызванного сходом лавины.

Сотрудники МЧС и эксперты МЧС выполняют определенные функции – анализ поступивших данных, принятие решений и выработку рекомендаций по проведению противолавинных мероприятий, ликвидации лавинной опасности и проведению экспертного анализа ситуации (выявление причин и последствий).

Штормовое предупреждение и проведение лавиноопасной экспертизы являются контролирующими действиями из категории «Управленческие действия».

Обозначим понятия следующим образом: c_1 – гражданин, c_2 – диспетчер, c_3 – команда МЧС, r_1 – сообщить, r_2 – проверить, r_3 – ликвидировать, e_1 – снеголавинная служба, e_2 – сообщения очевидцев, e_3 – штормовое предупреждение, e_4 – экспертиза, e_5 – справка о последствиях, e_6 – база данных.

Тогда контролируемая или проблемная микроситуация $MSit_1$ будет выглядеть так:

$$MSit_1 = \{c_1r_2e_5, c_1r_2e_4, c_3r_1e_1, c_2r_2e_2, c_2r_2e_3, c_2r_2e_6, c_2r_3e_7\}. \quad (1)$$

После формирования описания ситуации выполняется запрос к прецедентной БЗ на поиск близких микроситуаций [Кузмин & Сорочан, 2004]. Была найдена микроситуация, описывающая вызов пожарной команды. Для этой микроситуации была дана самая высокая оценка по показателю качества принятого

решения – минимуму рисков социального и экономического для обеспечения безопасности жизнедеятельности людей и инфраструктуры в районе ЧС. Была получена следующая микроситуация с описанием действий или решений:

- микроситуация с порядковым номером;
- клиент вызывает пожарную команду;
- диспетчер фиксирует заказ;
- диспетчер передает заказ свободной машине;
- команда производит выезд и ликвидацию пожара.

В символьном виде данная прецедентная микроситуация $MSit'_1$ представляется так:

$$MSit'_1 = \{c'_1 r'_1 e'_1, c'_2 r'_2 e'_2, c'_2 r'_2 e'_3, c'_2 r'_3 e'_4\}, \quad (2)$$

где c'_1 – клиент; c'_2 – диспетчер; e'_1 – пожарная часть; e'_2 – вызов; e'_3 – приезд пожарной команды на место происшествия; e'_4 – ликвидация пожара; r'_1 – вызвать; r'_2 – фиксировать; r'_3 – выполнить.

Для выделенных понятий определялась их «близость» в соответствии метрикой расстояния между понятиями.

Полученные результаты могут быть использованы для создания базы знаний повторного использования в ситуационном центре при поиске рекомендаций ЛПР как по предупреждению так и при ликвидации последствий ЧС.

Рекомендации для принятия управленческих решений по ликвидации лавинной опасности, получаются после модификации близкого удачного управленческого решения для микроситуации вызова пожарной команды. Близость микроситуаций определяется в такой последовательности:

1. Для количественных данных прецедентов множество микроситуаций, полученных в результате статистического анализа [Kuzemin & Lyashenko, 2007; Kuzemin et al, 2007] на выбранной метрике евклидова расстояния

$$d_{ik} = \sqrt{\sum_{j=1}^N (x_{ij} - x_{kj})^2}, \quad d_{ik} \leq \Delta,$$

где $d_{ik} \leq \Delta$ - условие для оценивания близости микроситуаций, x_{ij} , x_{kj} - количественные данные сравниваемых микроситуаций.

2. Для качественных данных в соответствии с метрикой сходства Хэмминга определяется схожесть процессов, т.е. ответов на вопрос «ЧТО». Похожесть процессов и действий в первую очередь заключается в фиксации вызова, связанного с возникшей ЧС. Действия еще могут быть такими: проверка и обработка полученной информации о ситуации, принятие решения и реализация рекомендаций по ликвидации. Функциональный блок «Зафиксировать вызов о пожаре» заменен на «Зафиксировать штормовое предупреждение».
3. Для ЛПР выдается рекомендации по предупреждению или ликвидации последствий ЧС выбранных управленческих решений – ответы на вопросы «КТО» и «КАК».

Выбор архитектуры программно-аппаратного комплекса ситуационного центра

В рамках данной статьи рассматриваются проблемы проектирования БД и СУБД. БД для ситуационного центра проходят разные этапы своего жизненного цикла, начиная от замысла системы, предпроектного обследования, включая этапы проектирования, эксплуатации, а далее - модернизации системы.

Создание БД для ситуационного центра практически невозможно без использования средств автоматизации проектирования (CASE-систем) [Диги, 2005]. Их использование позволяет не только ускорить работы и повысить качество их выполнения, но и дает инструменты для организации коллективного труда группы проектировщиков. Использование инструментальных средств при проектировании БД затрагивает разные этапы жизненного цикла. Это в определенной мере предопределяет процесс обследования и дает инструмент для отображения его результатов.

Различают прямое проектирование (*forward-engineering*) - процесс получения структуры базы данных для выбранной целевой СУБД на основе построенной ER- модели, и обратное проектирование (*reverse - engineering* – реверс - инжиниринг) - когда ER-модель получается на основе существующей базы данных. CASE - средства обычно поддерживают оба эти процесса. Для исследования использовались инструменты AllFusion ERwin Data Modeler 4.1.4.

В данной статье не ставилась цель представить логические и физические модели СУБД. Здесь отмечаются наиболее важные аспекты, которые при моделировании СУБД должны учесть чрезвычайную важность и сложность БД для ситуационного центра в условиях возможного внешнего воздействия на БД при ЧС. Особого внимания, заслуживают вопросы обеспечения высокой надежности и минимума рисков при предупреждении и ликвидации последствий ЧС.

Прежде всего, в ER моделях для **повышения надежности** (в соответствии со стандартом ISO 9126 – одним из важнейших показателей качества БД) следует учесть различные виды сбоев БД. В общем случае для БД известны два вида аппаратных сбоев: «мягкие» сбои, которые приводят к внезапной остановке работы компьютера (например, аварийное выключение питания), и «жесткие» сбои, характеризующиеся потерей информации на носителях внешней памяти. Программные сбои — это аварийное завершение работы системы управления базой данных (СУБД) или аварийное завершение пользовательской программы, в результате чего некоторая транзакция остается незавершенной.

Аппаратные сбои фиксируются в каждом конкретном случае в журналах (*archive logs*) (иногда поддерживаются две копии журнала, располагаемые на разных физических дисках), в которую поступают записи обо всех изменениях основной части БД. Ведется статистический и системный анализ показателей надежности БД и соответствующих рисков нарушения живучести БД в условиях ЧС.

При этом СУБД должна обладать способностью восстановления последнего согласованного состояния БД после любого аппаратного или программного сбоя.

Самая простая процедура обеспечения надежности восстановления БД — откат транзакции, выполненной пользователем, для чего все записи от одной транзакции связывают обратным списком от конца к началу (аналог Undo).

При «мягком» сбое во внешней памяти основной части БД могут находиться объекты, модифицированные транзакциями, не закончившимися к моменту сбоя, и могут отсутствовать объекты, модифицированные транзакциями, которые к моменту сбоя успешно завершились (по причине использования буферов оперативной памяти, содержимое которых при «мягком» сбое пропадает). В таком случае во внешней памяти журнала должны обязательно находиться записи, относящиеся к операциям модификации обоих видов объектов. Для восстановления БД после жесткого сбоя используют журнал и архивную копию БД.

Физическое разнесение экземпляров БД дает возможность избежать аппаратного сбоя и повышает надежность работы ситуационного центра в целом.

Универсальным принципом обеспечения надёжности является **резервирование**. В общем случае применяются четыре основных вида **резервирования**:

- Аппаратное резервирование, например, дублирование;
- Информационное резервирование, например — методы обнаружения и коррекции ошибок;
- Временное резервирование, например, методы альтернативной логики;
- Программное резервирование, применение независимых функционально равноценных программ.

За основной подход в повышении надежности БД можно использовать резервное копирование, при котором следует размещать на отдельных устройствах резервного копирования. В противном случае при сбое устройства, содержащего БД, резервные копии окажутся недоступными. Кроме того, размещение данных и их резервных копий на отдельных устройствах повышает производительность ввода-вывода, как при записи резервных копий, так и в *процессе производственного использования базы данных* (Рис. 4).

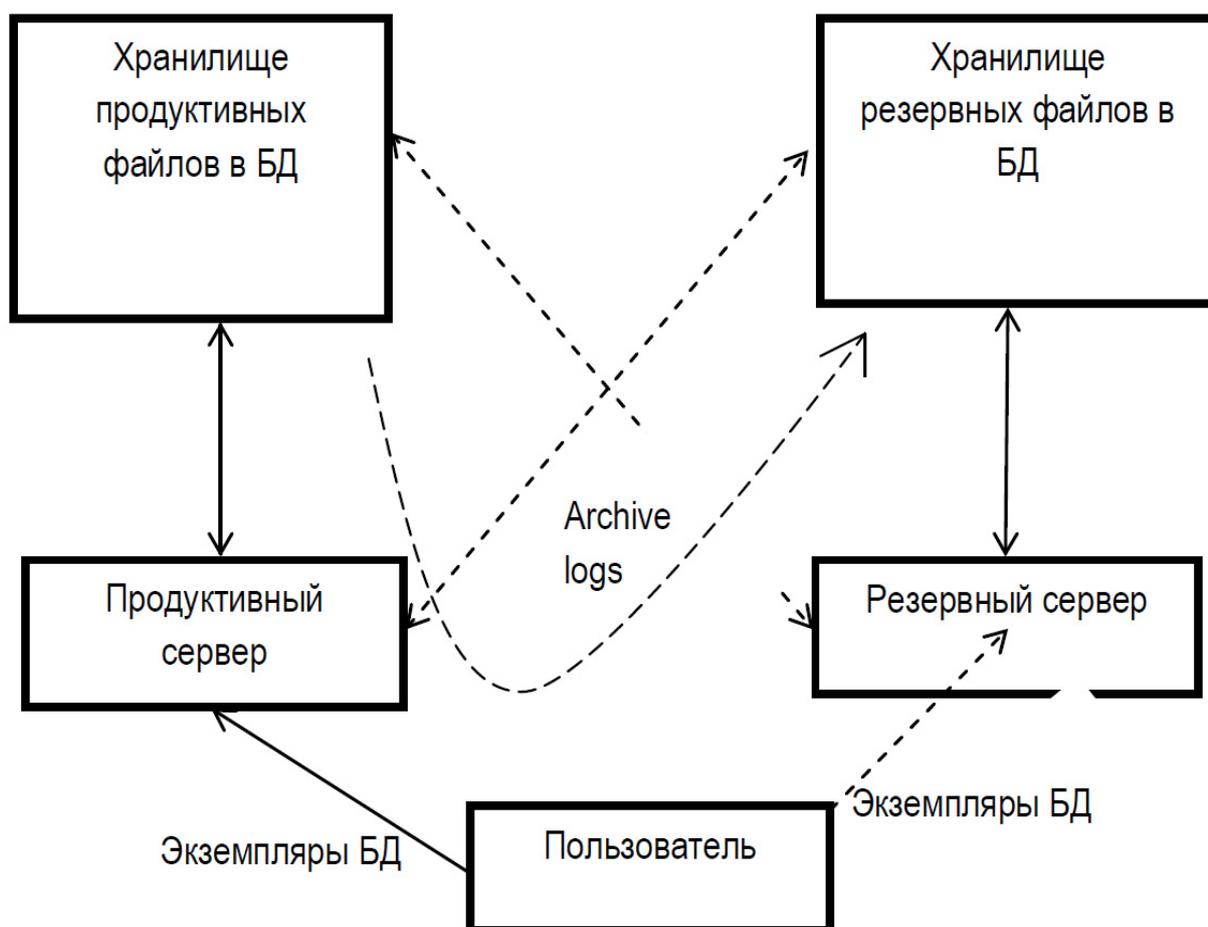


Рисунок 4. Схема резервирования сервера и БД

При логическом и физическом ER моделировании необходимо учесть технические решения по защите данных и сервисов, для ситуационного центра, как правило, важными являются две категории:

- Время восстановления (*recovery time objective (RTO)*) — допустимое время простоя сервиса в случае сбоя;
- Точка возврата (*recovery point objective (RPO)*) — допустимый объем возможных потерь данных в случае сбоя.

Эти категории обусловлены задачами ситуационного центра, и от их значений напрямую зависит архитектура предлагаемого решения и, соответственно, его стоимость.

При разработке сервера БД существуют **две проблемы**:

- A. Ограничение пропускной способности чтения данных.
- B. Ограничение пропускной способности записи данных.

Нет возможности решать две проблемы одновременно. Прежде всего, решается проблема, связанная с чтением данных, когда СУБД не в состоянии обеспечить то количество выборки, которое требуется.

Проблема А — чтения данных решается в такой последовательности:

1. Оптимизируются запросы и конфигурационные параметры (**оптимизация mysql, оптимизация postgres**);
2. В целях повышения управляемости, производительности и доступности для больших БД выполняется **секционирование** (англ. *partitioning*) — разделение хранимых объектов БД (таких как таблиц, индексов, материализованных представлений) на отдельные части с отдельными параметрами физического хранения.

Возможные критерии разделения данных, используемые при секционировании:

- по предопределённым диапазонам значений;
- по спискам значений;
- при помощи значений хэш-функций.

Скорее всего, у клиента есть несколько огромных таблиц (обычно всю нагрузку обеспечивают всего несколько таблиц СУБД из всех имеющихся). Причем чтение в большинстве случаев приходится только на самую последнюю их часть (т.е. активно читаются те данные, которые недавно появились). Разработчик распределяет нагрузку на таблицу по ее партициям. Следовательно, выборка типа «**SELECT * FROM articles ORDER BY id DESC LIMIT 10**» будет выполняться только над последней партицией, которая значительно меньше всей таблицы.

3. Следующий пункт разработки возможен **при использовании нескольких серверов**. Это **репликация**. Репликация - это синхронное/асинхронное копирование данных с ведущих серверов на ведомые (или возможно тоже ведущие) сервера. Ведущие сервера называют **мастерами** (*master*), ведомые — **слейвами** (*slave*). Вариантов репликации бывает несколько, но для решения проблемы чтения, используется *master-slave* репликация. Это решение потребует некоторых изменений в приложении (обычно не больших, хотя не все так просто), т.к. нужно будет читать данные с разных серверов БД, а не одного. Также, необходимо будет учесть репликационный лаг (задержка копирования данных на слейв с мастера — т.е. время, через которое данные полностью скопируются) в работе приложения. Можно выбрать один из двух

вариантов — синхронную или асинхронную репликацию. В случае первой не придется заботиться о лаге, но это отразится на скорости отработки запросов на изменение/вставку данных. Репликация — это наращиваемое решение. Если одного слейва не хватает — ставится второй, третий и т.д. Принцип выбора слейвов на уровне приложения значения не имеет. Главное — это балансирование нагрузки на все слейвы. Естественно существует предельный максимум слейв-серверов (и обычно он связан с тем, что на каком-то этапе уже мастер становится слабым звеном, и начинаются проблемы с записью).

Если были опробованы выше перечисленные способы, то используют **шардинг** — разделение данных на уровне ресурсов. Концепция шардинга заключается в логическом разделении данных по различным ресурсам исходя из требований к нагрузке. Рассмотрим пример. Пусть у нас есть приложение с регистрацией пользователей, которое позволяет писать друг другу личные сообщения. Допустим, оно очень популярно и много людей им пользуются ежедневно. Естественно, что таблица с личными сообщениями будет намного больше всех остальных таблиц в базе (скажем, будет занимать 90% всех ресурсов). Зная это, мы можем подготовить для этой (только одной!) таблицы выделенный высокопроизводительный сервер, а остальные таблицы оставить на других менее производительных. Теперь мы можем идеально подстроить сервер для работы с одной специфической таблицей, постараться уместить ее в память, возможно, дополнительно партиционировать ее и т.д. Такое распределение называется **вертикальным шардингом**. Если таблица с сообщениями стала настолько большой, что даже выделенный сервер под нее одну уже не спасает. Необходимо делать **горизонтальный шардинг** — т.е. разделение одной таблицы по разным ресурсам. На разных серверах будет таблица с одинаковой структурой, но разными данными. Для рассматриваемого примера с сообщениями, мы можем хранить первые 10 миллионов сообщений на одном сервере, вторые 10 — на втором и т.д. необходимо иметь критерий шардинга — какой то параметр, который позволит определять, на каком именно сервере лежат те или иные данные.

Обычно, в качестве параметра шардинга выбирают ID пользователя (`user_id`) — это позволяет делить данные по серверам равномерно и просто. В предлагаемом примере при получении личных сообщений пользователей алгоритм работы будет такой:

1. Определить, на каком сервере БД лежат сообщения пользователя исходя из `user_id`;
2. Инициализировать соединение с этим сервером;
3. Выбрать сообщения.

Первый этап — определения конкретного сервера можно решать двумя путями:

1. Хранить в одном месте хеш-таблицу с соответствиями «пользователь=сервер». Тогда, при определении сервера, нужно будет выбрать сервер из этой таблицы. В этом случае узкое место — это большая таблица соответствия, которую **нужно** хранить в одном месте. Для таких целей очень хорошо подходят **базы данных «ключ=значение»**;
2. Определять имя сервера с помощью числового (буквенного) преобразования. Например, можно вычислять номер сервера, как остаток от деления на определенное число (количество серверов, между которыми Вы делите таблицу). В этом случае узкое место — это проблема добавления новых серверов. Необходимо делать перераспределение данных между новым количеством серверов.

Для шардинга не существует решения на уровне известных платформ, т.к. это весьма специфическая для отдельно взятого приложения задача. Естественно, делая горизонтальный шардинг необходимо ограничить возможности выборки, которые требуют пересмотра всей таблицы (например, последние посты в блогах людей будет достать невозможно, если таблица постов шардится). Такие задачи придется решать другими подходами. Например, для описанного примера, можно при появлении нового поста, заносить его ID в общий стек, размером в 100 элементов.

Горизонтальный шардинг имеет одно явное преимущество — он бесконечно масштабируем.

Проблему В — **записи данных** следует решать в такой же последовательности, как и проблему чтений. Для начала необходимо рассмотреть возможность партиционирования, далее — репликации (в этом случае мастер-мастер репликация), ну и конечно шардинг.

В большинстве случаев, хватает решений с партиционированием либо репликацией для того, чтобы справиться с требуемыми нагрузками. Современные сервера среднего уровня — это весьма мощные компьютеры, способные выдерживать большие нагрузки при грамотном распределении ресурсов. В уникальных ситуациях Вам придется использовать шардинг. В любом случае, необходимо закладывать возможность гибкого перехода на репликационное решение либо шардинг. Можно применять **гибридные решения** и вынести таблицу на отдельный сервер и настроить на нем репликацию.

Необходимо подчеркнуть, что разработка модели управления БД должна соответствовать ИСО/МЭК ТО 10032 –2007.

Оценка рисков и надежности БД

При выполнении этого этапа проектирования для ситуационного центра необходима априорная статистическая информация. Так, например, согласно результатам автоматизированного статистического исследования различного программного обеспечения на предмет ошибок, в исходном коде PostgreSQL было найдено 20 проблемных мест на 775 000 строк исходного кода (в среднем, одна ошибка на 39 000 строк кода). Для сравнения: MySQL — 97 проблем, одна ошибка на 4000 строк кода; FreeBSD (целиком) - 306 проблем, одна ошибка на 4000 строк кода; Linux (только ядро) — 950 проблем, одна ошибка на 10 000 строк кода.

Статистические данные используются для оценки риска при проектировании БД:

$$\mathfrak{R} = \frac{n}{N},$$

где n — количество ошибок с нежелательными последствиями; N — максимально возможное их количеству за конкретный период времени.

Эта формула позволяет рассчитать размеры общего и группового риска. При оценке общего риска величина N определяет максимальное количество всех отказов, а при оценке группового риска - максимальное количество отказов в конкретной группе, которая выбрана из общего количества по определенному признаку.

В (Табл. 1) приведены возможные источники увеличения риска и надежности БД при отказах и возможные пути их нейтрализации.

Таблица 1. Возможные источники увеличения риска и надежности БД при отказах и возможные пути их нейтрализации

N п/п	Возможные источники увеличения риска и надежности БД при отказах	Пути нейтрализации отказов
	Ошибочное восстановление БД, для которой выполняется транзакция.	Присвоения каждой базе статуса ошибки для запрашиваемого действия.
	Возникновение временных рисков: потеря данных во время восстановления базы данных (БД может находиться в состоянии транзакций, несмотря на нанесенный ущерб); некоторые таблицы могут быть заполняемы даже во время, когда большая часть информации повреждена.	Ввести экстренный график для снятия TRN LOG резервных копий во время между запросом на восстановление и полным восстановлением.
	Более чем одна ошибка базы данных в один и тот же момент времени.	Необходимо четкое ранжирование ответов на статус базы данных и четкая логика принятия решений о действии по отношению к той или иной БД в зависимости от наиболее верного определенного состояния последней.

Оценивание допустимых значений критериев выбора технического решения

Оценивание качества БД необходимо проводить в соответствии с требованиями стандарта ISO 9126 путем применения шкал количественных и/или балльных оценок свойств или уровней качества БД с несколькими градациями. В (Табл. 2) приведены наиболее важные для решаемой проблемы шкалы оценивания характеристик БД.

Таблица 2. Характеристики качества БД

Характеристики качества	Мера	Шкала
Надежность		
– <i>Завершенность:</i> • наработка на отказ при отсутствии рестарта.	Часы	10 - 1000
– <i>Устойчивость:</i> • наработка на отказ при наличии автоматического рестарта; • относительные ресурсы на обеспечение надежности и рестарта.	Часы %	10 - 1000 10 - 90
– <i>Восстанавливаемость:</i> • длительность восстановления.	Минуты	10 ⁻² - 10.
– <i>Доступность-готовность:</i> • относительное время работоспособного функционирования	Вероятность	0,7 - 0,99

Возможный результат проектирования при заданных ограничениях на допустимые значения риска, надежности и показателей качества БД

Полученное техническое решение в результате проведенного исследования представляется системой, которая позволяет содержать 14 резервных копий либо как минимум 2 за последние 2 недели резервных копий типа «FULL». Разработанный алгоритм архивирования копий позволяет добиться результатов с коэффициентом 1:11 от базового размера резервной копии. Центральная БД сообщается с локальными (либо удаленными, но входящими в общую доменную группу) серверами через Push, Pull либо комбинированный метод в зависимости от степени важности БД и возможных временных рамок потерь данных (Рис. 5). Центральная БД служит хранилищем и выполняет управление архивами и резервными копиями БД. Pull метод используется для получения информации от не критических БД. Push метод используется для запросов от критических БД. Комбинированный метод Pull/Push (1 + 1) для критических (сверхкритических) БД. Применяется архивирование Резервных Копий БД.

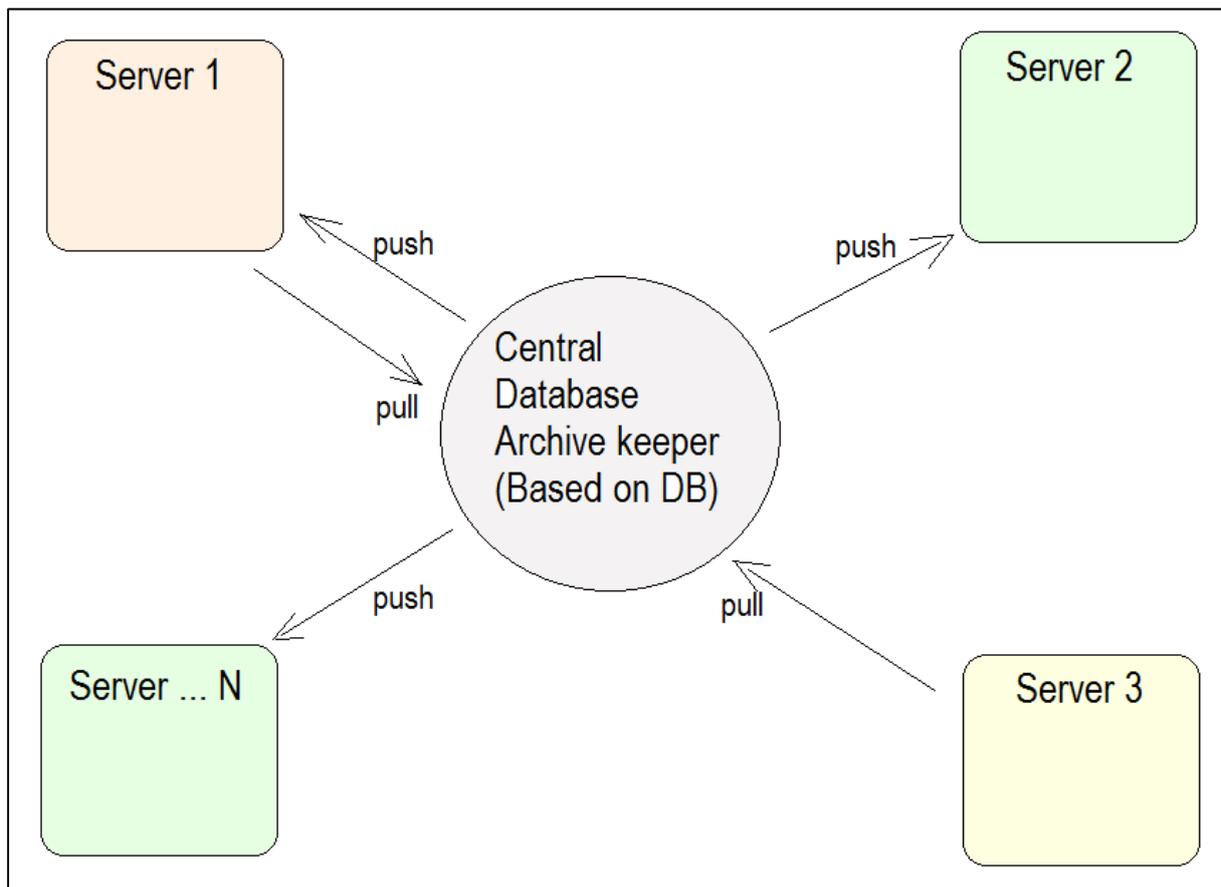


Рисунок 5. Базовые принципы связи основного «хранилища» с серверами

В результате выполнения предлагаемой технологии БД находятся в FULL и DIFFERENTIAL моделях восстановления. Используется от 1 до N серверов. Для каждой FULL модели восстановления имеется одна последняя архивированная копия на локальном сервере, кроме этого перемещается последняя архивированная копия в центральное хранилище. Максимальное число копий в центральном хранилище

не более 14 (либо за 2 последние недели, в случае если отрезок времени между FULL копиями превышает один день). Обеспечено резервное копирование (Full, Differential, Transaction Log, объекты БД). Используются архивирование копии данных. Применяется трансфер архива (или резервной копии в случае TRN LOG) на центральное хранилище. Выполняется управление архивом резервной копии (помещение в библиотеку, управление мгновенного доступа и подачи на восстановление). В режимах умеренного и полного риска применяется мгновенное восстановление.

Выводы

В статье предложена технологическая последовательность разработки БД для ситуационных центров в условиях возникновения ЧС. Реализация отказоустойчивого решения БД средствами дисковых массивов с применением вышеперечисленных технологий позволяет получить Заказчику необходимую надёжность и достичь поставленной цели.

Обеспечивается быстрое перемещение для SQL БД инстанций на сервере (либо самого сервера, либо группы серверов), быстрое автоматическое восстановление БД на сервере или группе серверов, восстановление данных может быть произведено следующими способами:

- Physical-to-Physical (P2P);
- Physical-to-Virtual (P2V).

Литература

[Kuzemin & Lyashenko, 2007] Alexander Kuzemin, Vyacheslav Lyashenko. Probabilistic and Multivariate Aspects of Construction of the Models and Procedures for Prediction of the Avalanche-Dangerous Situations Initiation. // Proc. of the Second International Conference i.TECH, Sofia, Bulgaria, ITHEA, 2007, pp. 284–289.

[Kuzemin et al, 2007] Alexander Kuzemin, Olesya Dyachenko, Darya Fastova Information Supply of Geo-information Systems for the Forecasting Problem of the Avalanche Danger. // Proc. of the Second International Conference i.TECH, Sofia, Bulgaria, ITHEA, 2007, pp. 289 – 294.

[Kuzomin & Torojev, 2006] Kuzomin A., Torojev A. Mobile means of control and prediction of avalanche climate using information conversion in acoustic. RANGE 291 // IDRC, DAVOS, 2006, Vol. 2, pp. 291 – 294.

[Диго, 2005] Диго С.М. Базы данных: проектирование и использование: Учебник. М.: Финансы и статистика, 2005.

[Куземин & Сорочан, 2004] Куземин А.Я., Сорочан М.В. Понятийное представление ситуации при поиске и классификации проектных решений. //Прикладная радиоэлектроника, Харьков: ХНУРЭ, 2004, Том 3 №3, С. 60 – 67

[Липаев, 2001] Липаев В.В. Выбор и оценивание характеристик качества программных средств. М.: «Синтег», 2001.

Благодарности

Статья публикуется при финансовой поддержке в рамках проекта ITHEA XXI Института информационной теории и приложений FOI ITHEA (www.ithea.org) и Ассоциации разработчиков и пользователей интеллектуальных систем ADUIS Украины (www.aduis.com.ua).

Информация об авторах



Oleksii Vasylenko – Aspirant of Kharkiv National University of Radio Electronics; Kharkiv, Ukraine; e-mail: ichbierste@gmail.com; tel.: +380 63 841 66 23

Major Fields of Scientific Research: General theoretical information research, Knowledge Discovery and Engineering, Business Informatics



Oleksandr Kuzomin – Prof. Dr., Professor of Informatics Department of Kharkiv National University of Radio Electronics; Kharkiv, Ukraine; e-mail: kuzy@daad-alumni.de; tel.: +38(057)7021515

Major Fields of Scientific Research: General theoretical information research, Decision Making, Emergency Prevention, Data Mining, Business Informatics

Data Loss Minimization in Situation's Centrums Data Bases

Oleksandr Kuzomin, Oleksii Vasylenko

Abstract: *Problems of reducing risks, increasing the safety and reliability and the use of databases (DB) in emergency situations (ES) are considered in the paper. To solve these problems a technological sequence is proposed. To analyze data, the peculiarity presentation of quantitative and qualitative parameters in emergency situations is calculated. The possibility of logical and functional ER modeling CASE technology with a regard to minimize the risk of life in the emergency area is shown. Technical solution of the problem is given.*

Keywords: *Emergency situations, Database (DB), archive, backup, database management system (DBMS), server.*

ОПРЕДЕЛЕНИЕ ИНТЕРВАЛА ЭФФЕКТИВНОСТИ ПАРАМЕТРОВ ПРОЦЕДУРЫ ВЫЧИСЛЕНИЯ ГРАНИЦЫ ПРОПУСКНОЙ СПОСОБНОСТИ ПАКЕТНОГО КОММУТАТОРА

Ташо Ташев, Арсений Баканов, Радостина Ташева

Аннотация: Задача вычисления бесконфликтного расписания для коммутатора пакетов с матричным переключателем является NP-полной с математической точки зрения. Поэтому постоянно предлагаются новые алгоритмы вычисления расписания. Их надо проверять. Пропускная способность коммутатора при работе „хорошего” алгоритма монотонно стремится к некоторому пределу. Нами была предложена процедура для определения количественных значений сходимости к пределу, использующая данные компьютерного моделирования. В этой работе представлены результаты моделировании одного алгоритма для конкретной модели входящего потока пакетов, выполненных на грид-структуре CERN-a (Ixplus.cern.ch). Проверялась гипотеза, что вычисление сходимости зависит от выбранного параметра m процедуры. Проверка проводилась для значений $m = 4$ и $m = 3$. Полученные данные подтверждают валидность гипотезы для выбранных значений m . В результате мы высказываем новое утверждение: полученные значения ошибки для внутренней переменной δ дают (независимую) оценку ошибок моделирования. На этой базе мы можем планировать такие эксперименты, которые дадут нам требуемую точность для определения границы при заданном ограничении доступной компьютерной мощности.

Ключевые слова: Моделирование, Коммуникационный узел, Матричный переключатель, Алгоритмы.

Классификация ключевых слов ACM: B.4.4 Performance Analysis and Design Aids, C.2.1 Network Architecture and Design, C.4 Performance of Systems

Введение

Коммуникационный узел для коммутации пакетов с матричным переключателем (crossbar switch node - коммутатор) имеет основную задачу - максимальное ускорение передачи пакетов данных, за счет параллельно распределенных потоков между узлами коммуникационной сети [Deb et al, 2006]. В идеальном случае узел передает пакеты между другими узлами, подключенными к его портам, с той скоростью с которой эти узлы генерируют пакеты, не внося дополнительных задержек и не теряя ни одного пакета [Elhanany and Hamdi, 2007]. Для достижения этой цели необходимо обеспечить бесконфликтное расписание для коммутации, вычисляемое управляющим блоком (Scheduler – Рис.1).

Формально, математическим путем доказано, что задача вычисления бесконфликтного расписания является NP-полной [Chen et al, 1990]. Существующие решения – алгоритмы PIM [Anderson et al, 1993], iSLIP [Gupta and McKeown, 1999], DISQUO [Shunyan et al., 2010], CTC(N) [Chang et al, 2012] и др. дают частичное достижение результата. При этом используются формализмы такие как теория массового обслуживания, генетические алгоритмы, матрицы-маски и др. [Elhanany, Hamdi, 2007; Kolchakov, 2011].

Достижения научно-технического прогресса ведут к постоянному повышению скоростей и объемов коммуникационного трафика [Атанасова и др., 2010], что требует новых алгоритмов. Их надо проверять на эффективность.

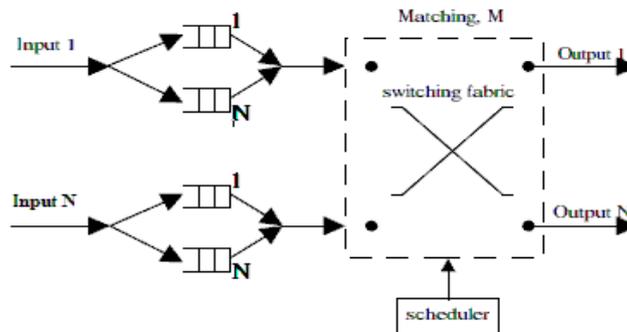


Рис 1. Логическая структура коммутатора с входящими буферами [Deb et al, 2006].

Первый шаг при проверке эффективности нового алгоритма для вычисления бесконфликтного расписания – это моделирование пропускной способности (throughput) коммутатора при равномерно распределенном (uniform i.i.d. Bernoulli) входящем трафике. Следующий шаг требует проверки при неравномерно распределенном трафике [Chang et al, 2012].

В настоящей работе использовано семейство шаблонов-матриц для моделирования неравномерно распределенного трафика на базе модели “горячей точки” (hotspot - Чао-модель [Chao-Lin et al., 2007]). Представлены результаты вычислительных экспериментов для PIM-алгоритма, модель которого построена нами с использованием аппарата Обобщенных сетей [Atanassov, 1997]. Пропускная способность коммутатора (Кпд) при работе PIM-алгоритма стремится к определенному пределу. Нами была предложена процедура для определения количественных значений сходимости к пределу, позволяющая вычислить этот предел однозначно [Ташев и др., 2013]. Там была высказана гипотеза, что вычисление сходимости зависит от выбираемого параметра m процедуры. Здесь представлены результаты проверки гипотезы. С помощью процедуры были получены численные значения для предела Кпд для двух значений параметра m процедуры. Они были сравнены и гипотеза получила подтверждение. На этой базе сделан вывод как планировать такие эксперименты, которые дадут требуемую точность для определения границы.

Задача построения бесконфликтного расписания

Проблема конфликтов определяется так: коммутатор имеет n входных каналов и n выходных каналов связи, то есть n источников пакетов через коммутатор могут посылать их к n получателям [Gupta and McKeown, 1999], Заявки на передачу пакетов формируют матрицу трафика T с размерностью $n \times n$, причем $T_{ij} = r$ ($r = 1, 2, \dots$) когда число заявок от i -того источника к j -тому приемнику равно r , и $T_{ij} = 0$ если нет заявки. Для бесконфликтной коммутации вычисляются последовательность бесконфликтных матриц $Q_1, Q_2, \dots, Q_i, \dots, Q_q$ (матрицы размерности $n \times n$, q – неизвестное целое), сумма которых давала бы T . Каждая Q_i ($1 \leq i \leq q$) такая, что в любой строке или столбце имеет не более одной единицы. “Длина” последовательности является целое положительное число $k_{\text{реш}}$ – это число матриц в решении (последовательности $Q_1 + Q_q$). Отметим, что матрица T представляет однонаправленный поток заявок – от входов к выходам коммутатора, представляющих пакетов с одной и той же длины.

Модель Чао (hotspot traffic) для равномерно распределенного по входам (i.i.d. Bernoulli) и неравномерно распределенного по выходам трафика дается как [Chao-Lin et al., 2007]: $\lambda_{ij} = \rho \cdot 0,5$ для $i = j$ и $\lambda_{ij} = \rho \cdot 0,5 / (n-1)$ в противном случае ($i, j \in 1, \dots, n$), где ρ - интенсивность нагрузки одной входящей линии. Для модели Чао строится базовая Трафичная матрица (для трафика типа "горячей точки"), которую будем обозначать $Chao_1$. Матрица для шаблона $Chao_1$ разворачивается в виде семейства шаблонов $Chao_i$, где $i=1, 2, \dots$, как показано на Рис. 2. Оптимальное решение для $Chao_i$ равно $k_{opt}=2 \cdot i \cdot (n-1)$ при размерности T равной $n \times n$ [Ташев, 2013], и вычисление Кпд производится как $Кпд = k_{opt} / k_{реш}$.

$$T = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \quad \begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{bmatrix} \quad \dots \quad \begin{bmatrix} k-1 & \dots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \dots & k-1 \end{bmatrix} \quad \dots$$

2 x 2 3 x 3 k x k

$$T = \begin{bmatrix} i & i \\ i & i \end{bmatrix} \quad \begin{bmatrix} i*2 & i & i \\ i & i*2 & i \\ i & i & i*2 \end{bmatrix} \quad \dots \quad \begin{bmatrix} i*(k-1) & \dots & i \\ \vdots & \ddots & \vdots \\ i & \dots & i*(k-1) \end{bmatrix} \quad \dots$$

2 x 2 3 x 3 k x k

Рис. 2 Матрицы трафика $Chao_1$ и семейства $Chao_i$ при размерности коммутатора до $k \times k$

Компьютерное моделирование PIM-алгоритма

Формально PIM-алгоритм нами специфицирован с помощью аппарата Обобщенных сетей (ОС) [Atanassov, 1997]. Основными элементами ОС являются переходы и позиции. В позициях находятся ядра, которые проходят через переходы соответственно с логических условий, имея необходимые характеристики. Любая ОС-модель имеет графическое представление (кроме формальной спецификации). Для модели PIM-алгоритма они представлены в [Tashev and Monov, 2012]. Переход от ОС-модели к вычислительным экспериментам выполнен сообразно [Tashev and Vorobiov, 2007]. Для программирования использован программный пакет Vfort, предоставленный для свободного пользования Московским Институтом математического моделирования РАН. Исходный код был компилирован средствами грид-структуры CERN-a (lplus.cern.ch) и результирующий код был выполнен там же. Основным ограничением при компьютерном моделировании являлось время выполнения кода программы (до 50 часов). Входными данными для алгоритма являются шаблоны $Chao_i$, матриц входящих заявок на коммутирование.

На Рис. 3 показаны результаты работы алгоритма. По горизонтальной оси координат показана размерность n входной матрицы T (от 3 до 60). Используются обозначения: C_1 – это Кпд входящего трафика вида $Chao_1$, C_5 - трафика вида $Chao_5$, и т.д.

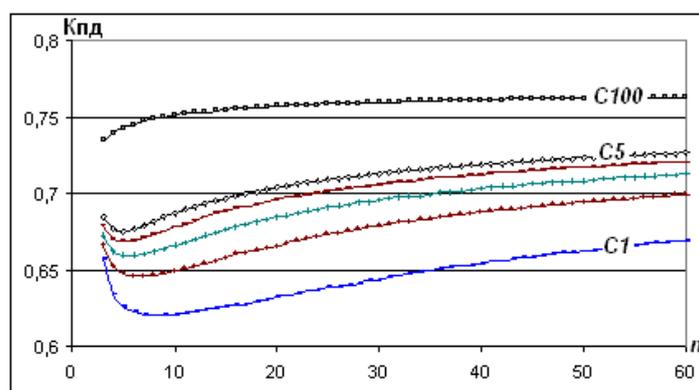


Рис. 3 Пропускная способность для шаблонов вида $Chao_{1-5}$ и $Chao_{100}$.

По вертикальной оси показано число $k_{\text{реш}}$ вычисленной последовательности выходных матриц Q_q , приведенное к пропускной способности коммутатора (т.е. $K_{\text{пд}} = k_{\text{опт}} / k_{\text{реш}}$). Кривые соответствуют результатам, полученными с шаблонами Chao_i трафика. Каждое значение для Chao₁ до Chao₅ – среднее от 10 000 „проходов” шаблона, а значения для Chao₁₀₀ – среднее от 1000 „проходов” шаблона. Результаты компьютерного моделирования указывают как на наличие предела $K_{\text{пд}}$ для каждой исследованной размерности коммутационного поля n при $i \rightarrow \infty$, так и при n клонящему к бесконечности. Для точного вычисления первого предела предложена соответствующая процедура.

Определение сходимости к верхней границе пропускной способности

Предложенная процедура вычисляет сходимость пропускной способности коммутатора к определенной верхней границе по результатам вычислительных экспериментов (при моделировании конкретного алгоритма с заданной семьей шаблонов для входящего трафика). Далее мы воспользуемся ею без формального описания. Для нас важно проверить гипотезу $\delta = m^{1/2}$ (при $m=10$, $\delta \approx 3,15$ [Ташев и др., 2013]).

Шаг 1. На основании данных Рис. 3 решаем что есть основания существованию верхней границы $K_{\text{пд}}$.

Шаг 2. Выбираем „шаг” для индекса i шаблона равен 4 ($m = 4$) в том смысле, что будем исследовать $K_{\text{пд}}$ для ряда шаблонов Chao₁, Chao₄, Chao₁₆, Chao₆₄, Chao₂₅₆, Результаты моделирования для первых четырех шаблонов даны на Рис. 4. Значения для C_1 и C_4 – среднее от 10 000 „проходов”, для остальных – 1000 „проходов”.

Шаг 3. Вычисляем разницу между $K_{\text{пд}}$ для соседних шаблонов. Строим график как показано на Рис. 5. Имеем $\Delta_1 = K_{\text{пд}}(\text{Chao}_4) - K_{\text{пд}}(\text{Chao}_1)$, $\Delta_2 = K_{\text{пд}}(\text{Chao}_{16}) - K_{\text{пд}}(\text{Chao}_4)$, $\Delta_3 = K_{\text{пд}}(\text{Chao}_{64}) - K_{\text{пд}}(\text{Chao}_{16})$

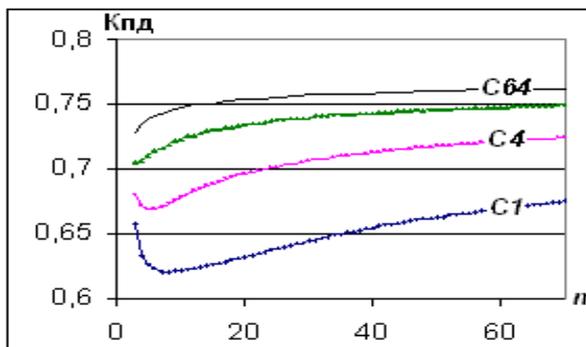


Рис.4 Пропускная способность для Chao₁ до Chao₆₄

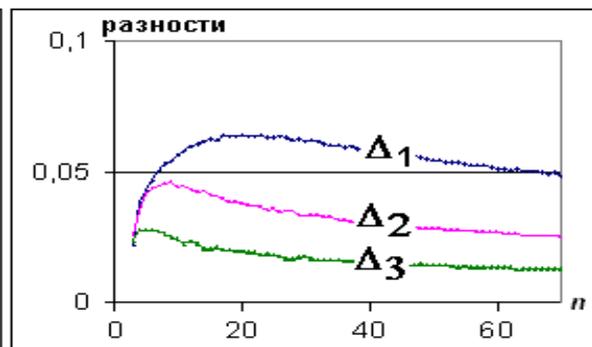


Рис.5 Разности пропускной способности ($m = 4$)

Шаг 4. Вычисляем отношение разностей $\delta_1 = \Delta_1 / \Delta_2$, $\delta_2 = \Delta_2 / \Delta_3$. Графики показаны на Рис. 6 (влево).

Оба отношения сходятся к значению 2, но у обоих видно влияние смещения минимума $K_{\text{пд}}$ C_1 и C_4 влево (при увеличении индекса шаблона). Второе отношение устремляется быстрее к 2, но ошибка большая (так как число „проходов” при C_{16} и C_{64} меньше). Поэтому вычислим $\delta_3 = \Delta_3 / \Delta_4$, $\Delta_4 = K_{\text{пд}}(\text{Chao}_{256}) - K_{\text{пд}}(\text{Chao}_{64})$ используя шаблон Chao₂₅₆. Результат показан на Рис. 6 (вправо): δ_3 уже колеблется около значения 2.

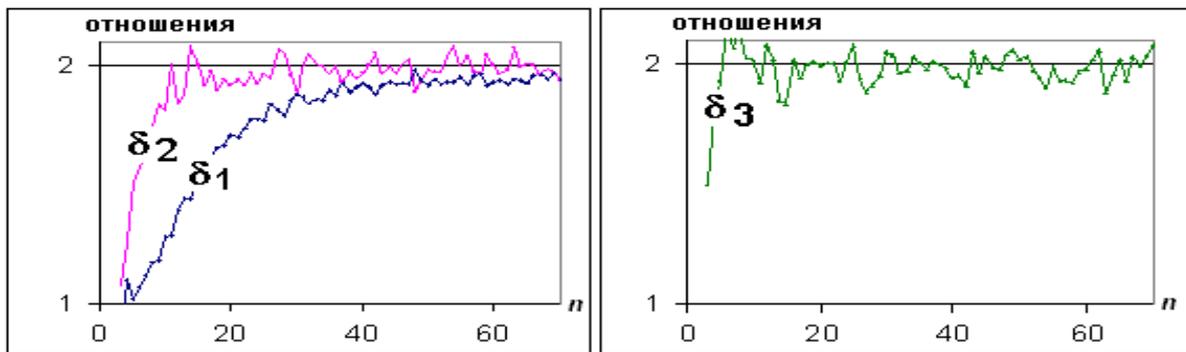


Рис.6 Отношения разностей пропускной способности ($m = 4$)

Шаг 5. Результаты моделирования показывают, что исключая зоны смещения минимума Кпд ($2 < n < 15$), разности Кпд между „шагами” шаблонов уменьшаются в геометрической прогрессии. Из теории известно, что бесконечный числовой ряд вида $1/a + 1/a^2 + 1/a^3 + \dots + 1/a^i + \dots$, где $a > 1$, сходится при $i \rightarrow \infty$ к значению $1/(a-1)$. В нашем случае разности $\Delta_2, \Delta_3, \dots$ образуют такой ряд с $a = 2$. Следовательно, коэффициент сходимости для наших результатов $\text{Sum}(2) = 1/(2-1) = 1$. А гипотеза $\delta = m^{1/2}$ подтверждается: $2 = 4^{1/2}$.

Поэтому вычисляем значения верхней границы для конкретных n , используя шаблон с самым большим номером - Chao_{256} , таким образом:

$$\text{Граница Кпд}(n) = \text{Кпд}(\text{Chao}_{256}(n)) + \text{Sum}(2). \Delta_4(n) = \text{Кпд}(\text{Chao}_{256}(n)) + 1. \Delta_4(n)$$

Результат показан на Рис. 7: с (δ граница) обозначено вычисление использующее $\delta = \text{const} = 2$. Самое большое значение $\text{Кпд}(\text{Chao}_{256}(70)) = 0,76767$, $\Delta_4(70) = 0,00618$, **Граница Кпд(70) = 0,77385**.

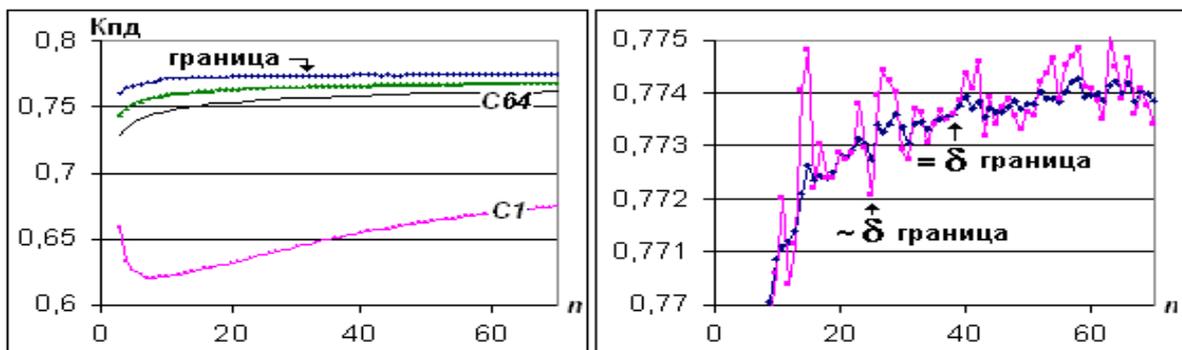


Рис.7 Граница пропускной способности ($m = 4$)

Шаг 6. Рассматриваем на Рис. 7 тенденции нарастания значения границы Кпд. Имея в виду самое большое значение 0,7738 для Кпд, полученное на Шаге 5, оцениваем значение при $n \rightarrow \infty$ на 0,775.

Шаг 7. Первичный анализ ошибок. На Рис. 7 справа показаны значения границы, вычисленные при $\text{Sum}(\delta = \text{const} = 2)$ и $\text{Sum}(\delta_3 = \text{variable})$ (обозначено $\sim \delta$ граница). Ошибка **Граница Кпд**(n) = $\pm 0,0003$ - уменьшается больше чем в два раза при вычислении использующее $\delta = \text{const} = 2$. Ошибка больше зависит - при таком выборе - от точности предыдущего шаблона ($\text{Chao}_{64}(n)$).

Для проверки гипотезы $\delta = m^{1/2}$ при $m = 3$ повторим Шаги 2 - 7.

Шаг 2. Будем исследовать Кпд для ряда шаблонов $Chao_1, Chao_3, Chao_9, Chao_{27}, Chao_{81}, \dots$. Результаты моделирования для первых четырех шаблонов даны на Рис.8. Повысим точность – все значения для C_1 до C_{81} – среднее от 10 000 „проходов”.

Шаг 3. Вычисляем разницу между Кпд для соседних шаблонов. Строим график как показано на Рис. 9.

Шаг 4. Вычисляем отношение разностей $\delta_1 = \Delta_1 / \Delta_2$, $\delta_2 = \Delta_2 / \Delta_3$. Графики показаны на Рис. 10 (влево). Сходимость – медленно, ошибки – меньше, по отношению ($m = 4$). Поэтому опять вычислим δ_3 : уже колеблется около 1,73 (Рис. 10 вправо).

Шаг 5. В случае $m = 10$ получили, что δ_3 колебалось около 3,15. Выше за случая $m = 4$ δ_3 было около 2. Сейчас за случая $m = 3$ получили значение для δ_3 около 1,73. Имеем еще одно подтверждение гипотезы: $1,732 = 3^{1/2}$. Тогда колебания значения δ_3 – это следствие ошибок компьютерного моделирования (в данном исследовании – из-за случайного выбора, используемого в PIM-алгоритме). В результате мы приходим к **Утверждению**: полученные отклонения значений внутренней переменной δ от точного значения $m^{1/2}$ дают (независимую) оценку численных ошибок компьютерного моделирования.

Закончим шаг 5. Принимаем, что разности $\Delta_3, \Delta_4, \dots$ образуют ряд с $a = 3^{1/2}$. Следовательно, коэффициент сходимости для $m = 3$ есть $Sum(3^{1/2}) = 1/(1,732-1) = 1,366$.

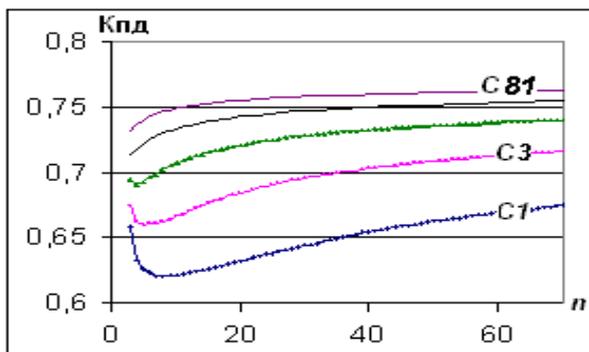


Рис. 8 Пропускная способность для $Chao_1$ до $Chao_{81}$

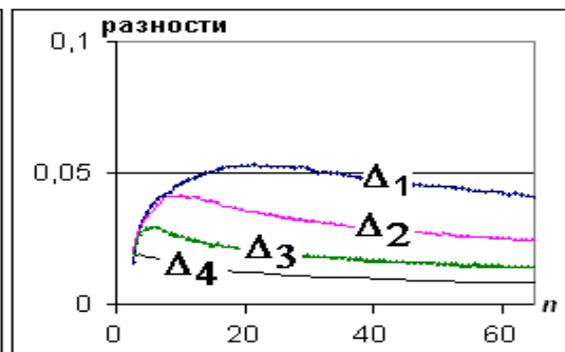


Рис. 9 Разности пропускной способности ($m = 3$)

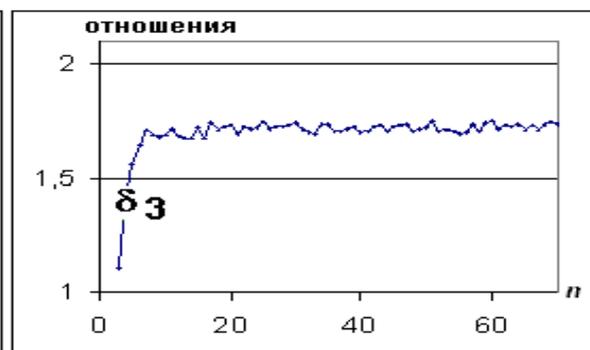
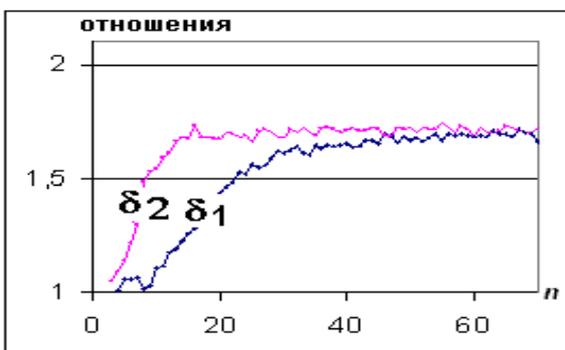


Рис. 10 Отношения разностей пропускной способности ($m = 3$)

Вычисляем значения верхней границы, используя шаблон $Chao_{81}$. Результат показан на Рис. 11: c ($= \delta$ граница) обозначено вычисление использующее $\delta = \text{const} = 1,732$. Самое большое значение $\text{Кпд}(Chao_{81}(70)) = 0,76281$, $\Delta_4(70) = 0,00819$, $1,366 \cdot 0,00819 = 0,1119 \rightarrow$ **Граница Кпд(70)=0,77400**.

Шаг 6. Рассматриваем на Рис. 11 тенденции нарастания значения границы Кпд. Имея в виду самое большое значение 0,7740 для Кпд, полученное на Шаге 5, подтверждается значение при $n \rightarrow \infty$ как 0,775.

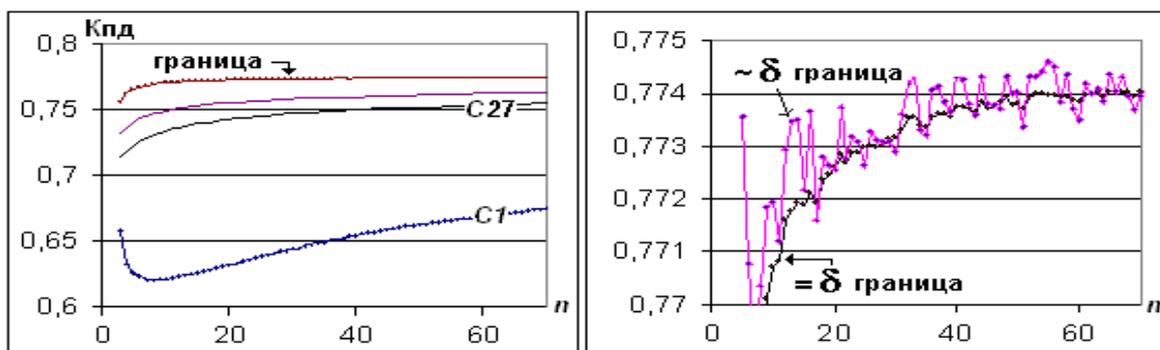


Рис. 11 Граница пропускной способности ($m = 3$)

Шаг 7. Первичный анализ ошибок. Более точны данные (10 000 симуляций) ведут к меньшей ошибке границы при $\delta_3 = \text{variable}$ (обозначено $\sim\delta$ граница). Ошибка $\text{ГраницаКпд}(n) = \pm 0,00015$ при $\delta = \text{const} = 3\%$ тоже меньше. Сравним оба графика $\text{Граница Кпд}(n)$ для $\delta = \text{const}$ как показано на Рис. 12.

Вычисленная $\text{Граница Кпд}(n)$ при $m = 3$ более „гладкая”. Затраченное компьютерное время для моделирования: $81+27=108$ условных единиц времени (10 000 „проходов”). Затраченное компьютерное время для моделирования при $m = 4$: $(265+64)/10 = 32$ условных единиц (1000 „проходов”), а потеря точности ~ 2 раза. Имея в виду, что нас ограничивает время моделирования, то пока не достигнут этот потолок, при стремлении достигнуть большей точности надо планировать эксперименты с $m=3$, при увеличенном числе „проходов” для каждого шаблона.



Рис. 12 Граница пропускной способности при $m = 4$ и $m = 3$

Заключение

Исследование подтверждает зависимость коэффициента сходимости от параметра m для исследованных значений $m = 4$ и $m = 3$. В результате полученные значения ошибки для внутренней переменной δ можно использовать как (независимую) оценку ошибок моделирования. На этой базе при планировании

дальнейших экспериментов для получения требуемой точности для определения границы Кпд, при заданном ограничении доступной компьютерной мощности, предлагается использовать значение $m = 3$. Описанную процедуру следует апробировать при исследовании других алгоритмов и моделей трафика.

Благодарности

Авторы благодарны доценту к.ф.н. Леандру Литову, Заведующий секцией Атомная Физика, Физический Факультет, Софийский Университет „Св. Климент Охридски“, Болгария, за предоставленный доступ к компьютерным ресурсам грид-структуры CERN.

Библиография

- [Anderson et al, 1993] T.Anderson, S.Owicki, J.Saxe and C.Thacker. High speed switch scheduling for local area networks. ACM Trans. Comput. Syst., vol. 11, no.4, 1993, pp.319-352.
- [Atanassov, 1997] Atanassov K. Generalized Nets and System Theory. Akad. Press "Prof.M. Drinov", Sofia, Bulgaria, 1997.
- [Chang et al, 2012] H.J.Chang, G.Qu, S.Zheng. Performance of CTC(N) Switch under Various Traffic Models. Springer, Lecture Notes in Electrical Engineering, Volume 126, 2012, pp. 785-793.
- [Chao-Lin et al., 2007] Chao-Lin Yu, Cheng-Shang Chang, Duan-Shin Lee.CR Switch: A Load-Balanced Switch With Contention and Reservation. IEEE/ACM Transactions on Networking, vol. 17 no.5, October 2007. pp.1659–1671.
- [Chen et al, 1990] T.Chen, J.Mavor, Ph.Denyer, D.Renshaw. Traffic routing algorithm for serial superchip system customisation. IEE Proc 137:[E]1, 1990.
- [Deb et al, 2006] S.Deb, D.Shah, S.Shakkottai. Fast Matching Algorithms for Repetitive Optimization: An Application to Switch Scheduling // Information Sciences and Systems, 2006 40th Annual Conference on, 22-24 March 2006, P. 1266-1271.
- [Elhanany and Hamdi, 2007] I.Elhanany, M.Hamdi. High-performance packet switching architectures. Springer-Verlag London Limited, 2007.
- [Gupta and McKeown, 1999] P.Gupta, N.McKeown. Designing and Implementing a Fast Crossbar Scheduler. IEEE Micro, Jan-Feb 1999, pp. 20-28.
- [Kolchakov, 2011] Kolchakov K. An Algorithm Synthesis of Non-Conflict Schedule by Diagonal Connectivity Matrix Activation. Proceedings of the International Conference AUTOMATICS AND INFORMATICS'11, John Atanasoff Society of Automatics and Informatics, Bulgaria, Sofia 03.10-07.10.2011., pp. B-247 – B251.
- [Shunyuan et al., 2010] Shunyuan Ye, Yanming Shen, Shivendra Panwar, DISQUO: A Distributed 100% Throughput Algorithm for a Buffered Crossbar Switch, Proceedings of IEEE Conference on High-Performance Switching and Routing (HPSR 2010), Richardson, TX, USA, 2010. pp. 63 – 68.
- [Tashev and Monov, 2012] Tashev T., Monov V. Modeling of the Hotspot Load Traffic for Crossbar Switch Node by Means of Generalized Nets. Proc. of Intelligent Systems (IS), 2012 6th IEEE International Conference, 6-8 Sept. 2012, Sofia, Bulgaria. pp.187-191.
- [Tashev and Vorobiov, 2007] Tashev T.D., Vorobiov V. M. Generalized Net Model for Non-Conflict Switch in Communication Node. Proc. Of Int. Workshop "DCCN'2007" September 10-12, 2007, Moscow, Russia. IPPI Publ., Moscow, 2007, pp.158-163.
- [Атанасова и др., 2010] Т. Атанасова, Т. Н. Савченко, Г.М. Головина, А С. Баканов. Интеллектуальная информационная среда обитания и субъективное восприятие качества жизни. Методы исследования психологических структур и их динамики: Вып. 5: / Под редакцией Т. Н. Савченко, Г.М. Головиной. - М.: Издательство «Институт психологии РАН», 2010. – стр. 164-170. (Труды Института психологии РАН)

[Ташев и др., 2013] Ташев Т., Баканова Н., Ташева Р. Исследование верхней границы пропускной способности коммутационного узла при входящем трафике типа "горячей точки". International Journal "Information Technologies & Knowledge", Vol.7, No. 2, 2013, pp. 182-189.

[Ташев, 2013] Ташев Т.Д. Моделирование пропускной способности MiMa-алгоритма для пакетного коммутатора при входящем трафике типа „горячей точки”. Proc. of International Conference "DCCN'2013", October 7-10 2013, Moscow, Russia. JSC "TECHNOSPHERA", Moscow, 2013. pp.257-264.

Информация об авторах



Ташев Ташо Димитров – инж.-математик, главный ассистент, Институт информационных и коммуникационных технологий Болгарской академии наук, ул. "Акад.Г.Бончев" бл.2, София 1113, Болгария; e-mail: ttashev@iit.bas.bg

Основные области научных исследований: Проектирование распределенных информационных систем, Методы и средства исследования сетевых моделей



Баканов Арсений Сергеевич – к.т.н., Институт психологии Российской академии наук, Москва, Россия; e-mail: arsb2000@pochta.ru

Основные области научных исследований: Интеллектуальные системы, Системы принятия решений, Софтверные технологии



Ташева Радостина Пенева – к.ф.н., Технический университет-София, бул. "Климент Охридски" 8, бл.10, София 1000, Болгария; e-mail: rpt@tu-sofia.bg

Основные области научных исследований: Процессы звездообразования, Галактики с активными ядрами, Компьютерное моделирование

Efficiency Interval Determination of a Calculation Procedure for Throughput Boundary of Crossbar Switch

Tasho Tashev, Arsenij Bakanov, Radostina Tasheva

Abstract: The problem connected to computation of non-conflict schedule for crossbar switch node is NP-complete from mathematical point of view. It is the reason for constantly suggested new algorithms for schedule calculations. The problem is that their efficiency has to be checked. The throughput of a "good" algorithm tends monotonically to some boundary. We suggest a procedure for quantitative determination of the convergence toward the boundary, utilizing data from computer simulation. In this paper are presented results of computer simulation for PIM algorithm with a certain model of incoming traffic (Chao model), made by means of CERN grid structure (lplus.cern.ch). We check the hypothesis that convergence computation depends on the chosen parameter m of the procedure. Checking is made for $m=4$ and $m=3$. The obtained data confirm validity of the hypothesis for the chosen values m . As a result we reach a new conclusion: the derived values of the error for the internal variable δ give (an independent) estimation of the simulation errors. Having this in mind we can plan experiments that will give us the required accuracy for the determination of the boundary under the limited computer power available.

Keywords: Modeling, Communication node, Crossbar switch, Algorithms.

IN MEMORIAM**Адиль Васильевич Тимофеев****1944 - 2014**

Профессор, доктор наук, Адиль Васильевич Тимофеев был выдающийся ученый, внёсший крупный вклад в науку в области нейроинформатики для оптимального и нейросетевого представления знаний и распознавания образов и теории робастного, адаптивного и нейросетевого управления движением, мульти-агентной навигации, функциональной диагностики и математического моделирования аэрокосмических, робототехнических и инфотелекоммуникационных систем.

В этой области им создана отечественная научная школа, признанная присвоением гранта "Государственная поддержка ведущих научных школ". За научные достижения ему дважды присуждались "Государственная стипендия для учёных", пятикратно звание и грант "Соросовского профессора", медаль "За научные достижения", звание "Заслуженный деятель науки Российской Федерации", грант "Государственная поддержка ведущих научных школ", звание "Заслуженный деятель науки Республики Дагестан" и почётное звание профессора трёх ведущих университетов Китая.

Под руководством А.В. Тимофеева выполнен ряд проектов по Федеральной целевой научно-технической программе по приоритетным направлениям развития науки и техники, Федеральной целевой программе "Интеграция", грантам РФФИ, РФФИ-ГФЕН, РГНФ, программам Президиума РАН, Минпромнауки РФ, Минобразования РФ, а также по Европейской программе "COPERNICUS", программам НАТО и Франко-Русского центра прикладной математики и информатики им. А.М.Ляпунова.

Он вел большую научно-организаторскую и педагогическую работу в Санкт-Петербурге и Северо-Кавказском регионе, являясь членом Объединенного Научного Совета по информатике, связи и управлению Санкт-Петербургского Научного Центра РАН, членом Санкт-Петербургской группы Национального Комитета по автоматическому управлению (IFAC), членом Научно-методического Совета по мехатронике и робототехнике, членом ряда учёных и диссертационных советов.

Им подготовлено более 15 кандидатов наук, опубликовано более 390 научных трудов, в том числе 10 статей в "Докладах АН" и 17 монографий и учебников.

Родился Адиль Тимофеев 17 августа 1944 г. в г. Махачкала. После окончания в г. Нальчике средней школы в 1961 г. поступил на кафедру П6 Приборостроительного факультета МВТУ им. Н.Э. Баумана, которую закончил с отличием в 1967 г. по специальности "математические и счётно-решающие приборы и устройства".

В 1967 г. поступил в аспирантуру на кафедру теоретической кибернетики математико-механического факультета Ленинградского государственного университета (ЛГУ), которую закончил досрочно под руководством чл.-корр. РАН В.А.Якубовича.

После защиты кандидатской диссертации на тему "Принципы построения нелинейных распознающих и адаптивных систем" на факультете прикладной математики и процессов управления ЛГУ ВАК присудил ему в 1970 г. учёную степень кандидата технических наук.

Работая старшим научным сотрудником НИИ математики и механики имени академика АН СССР В.И. Смирнова при Ленинградском государственном университете, он подготовил докторскую диссертацию "Адаптивное управление робототехническими системами", которую защитил в 1982 г. в МВТУ.

В 1984 г. А. Тимофеев был приглашён в Ленинградский институт авиационного приборостроения на должность профессора кафедры робототехнических и электромеханических систем, основанной чл.-корр. АН СССР Д.А. Завалишиным, где в 1997 г. был избран заведующим кафедрой.

С 1991 г. является заведующим лабораторией информационных технологий в управлении и робототехнике Санкт-Петербургского института информатики и автоматизации РАН (СПИИРАН) и одновременно был заведующим отделом интеллектуализации информационно-управляющих систем НИИ прикладной математики и автоматизации Кабардино-Балкарского Научного Центра РАН (КБНЦ РАН). С 2002 г. является также заведующим лабораторией мульти-агентных систем Института информатики и проблем регионального управления КБНЦ РАН.

В 2005 г. избран заведующим базовой кафедры нейроинформатики и робототехники Санкт-Петербургского государственного университета аэрокосмического приборостроения.

По его инициативе в 1996 г. был открыт в США международный журнал "Intelligent Control, Neurocomputing and Fuzzy Logic", он является членом редколлегии журналов "Мехатроника, Автоматизация, Управление", "Доклады АМАН" и "Дифференциальные уравнения и процессы управления", "Нейрокомпьютеры" и "Information Theories and Applications", "Information Technologies and Knowledge", "Information Modelling and Analyses", "Information Content and Processing".

В течение ряда лет являлся сопредседателем Научного Совета по управлению и автоматизации Объединённого Научного Совета по информатике, управлению и связи Санкт-Петербургского Научного Центра РАН, членом Международного общества по физике и управлению (ISPC) и Национального Комитета по автоматическому управлению (IFAC), Научно-методического Совета по мехатронике и робототехнике Министерства образования и науки РФ, заведующим базовой кафедры нейроинформатики и робототехники Санкт-Петербургского государственного университета аэрокосмического приборостроения при СПИИРАН, профессором кафедры информатики Санкт-Петербургского государственного университета и кафедры мехатроники и робототехники Балтийского государственного университета "Военмех".

Адиль Васильевич был полон сил, энергии и творческого вдохновения. Следуя семейным традициям, примеру знаменитых предков, он плодотворно работал, отдавая все свои силы, знания и опыт благородному служению науке и просвещению.

В наших сердцах останется глубокое сожаление о потере Учителя и Друга, и в то же время – светлое чувство причастности к его жизни.

Сохраним тепло дружеского общения **Адила Васильевича Тимофеева**.

Кр. Марков

ITHEA International Scientific Society

TABLE OF CONTENTS

Классификатор для статического обнаружения компьютерных вирусов, основанный на машинном обучении	103
Евгений Путин, Адиль Тимофеев	
Программная система для решения задачи синтеза Synthesis 1.0	113
Alexander Dokukin	
Информационная поддержка подготовки проектов управленческих решений по организационной деятельности	119
Нина Баканова	
Психологический подход к извлечению знаний в процессе человеко-компьютерного взаимодействия с системой электронного документооборота	124
Арсений Баканов	
Двухуровневый генетический алгоритм реконфигурации программируемых логических интегральных схем	131
Алексей Городилов	
Особенности построения современной беспроводной корпоративной сети	141
Юрий Лисецкий, Наталья Каревина	
Dependence of Information and Telecommunication Technology Development on Economic Indicators	147
Galina Gayvoronska, Illia Ganytskyi, Petr Yatsuk	
Метод обеспечения структурной живучести телекоммуникационной сети	152
Нина Князева	
Распознавание объектов с использованием критерия на основе векторной меры близости образов в пространстве ошибок	167
Петр Четырбок	
Обеспечение безопасности использования баз данных в условиях чрезвычайных ситуаций	173
Александр Кузёмин, Алексей Василенко	
Определение интервала эффективности параметров процедуры вычисления границы пропускной способности пакетного коммутатора	188
Ташо Ташев, Арсений Баканов, Радостина Ташева	
<i>In memoriam</i> : Адиль Васильевич Тимофеев	197
Table of contents	200