

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Том 21
2015
№ 12

ТЕОРЕТИЧЕСКИЙ И ПРИКЛАДНОЙ НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ

Издается с ноября 1995 г.

УЧРЕДИТЕЛЬ

Издательство "Новые технологии"

СОДЕРЖАНИЕ

ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

- Харламов А. А., Ермоленко Т. В. Нейросетевая среда (нейроморфная ассоциативная память) для преодоления информационной сложности. Поиск смысла в слабоструктурированных массивах информации. Часть II. Обработка информации в гиппокампе. Модель мира 883
- Лелеков С. Г., Лях А. М. Нечеткий вывод в адаптивных таксономических экспертных системах 890
- Дыкин В. С., Мусатов В. Ю., Варезников А. С., Большаков А. А., Сысоев В. В. Применение генетического алгоритма для определения архитектуры нейронной сети для прибора вида "электронный нос" 895

МОДЕЛИРОВАНИЕ И ОПТИМИЗАЦИЯ

- Кухаренко Б. Г., Солнцева-Чалей М. О. Спектральный метод с использованием полярной кривизны для анализа результатов кластеризации многомерных траекторий 901
- Моисеев А. А. Модель продаж стандартными партиями 906

СИСТЕМЫ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ

- Стемпковский А. Л., Тельпухов Д. В., Соловьев Р. А., Рухлов В. С. Тестовая система для сравнения алгоритмов, увеличивающих надежность комбинационных схем 910
- Петросянц К. О., Харитонов И. А., Кожухов М. В., Самбурский Л. М. Схемотехнические SPICE-модели биполярных и МОП-транзисторов для автоматизации проектирования радиационно стойких БИС 916

ЦИФРОВАЯ ОБРАБОТКА СИГНАЛОВ И ИЗОБРАЖЕНИЙ

- Червяков Н. И., Ляхов П. А., Калита Д. И. Выбор оптимального набора модулей системы остаточных классов для повышения производительности фильтров с конечной импульсной характеристикой 923

БЕЗОПАСНОСТЬ ИНФОРМАЦИИ

- Щеглов К. А., Щеглов А. Ю. Интерпретация и моделирование угрозы атаки на информационную систему. Часть I. Моделирование угрозы уязвимости и интерпретация угрозы атаки 930

Журнал в журнале

НЕЙРОСЕТЕВЫЕ ТЕХНОЛОГИИ

- Галушкин А. И. Нейрочипы и нейроморфные ЭВМ: проблемы моделирования 942
- Горбатков С. А., Касимова Л. И. Нейросетевой итерационный метод построения динамических моделей вероятности риска банкротства при неполных данных. 949
- Указатель статей, опубликованных в журнале "Информационные технологии" в 2015 г. 958

Главный редактор:

СТЕМПКОВСКИЙ А. Л.,
акад. РАН, д. т. н., проф.

Зам. главного редактора:

ИВАННИКОВ А. Д., д. т. н., проф.
ФИЛИМОНОВ Н. Б., д. т. н., с.н.с.

Редакционный совет:

БЫЧКОВ И. В., акад. РАН, д. т. н.

ЖУРАВЛЕВ Ю. И.,

акад. РАН, д. ф.-м. н., проф.

КУЛЕШОВ А. П.,

акад. РАН, д. т. н., проф.

ПОПКОВ Ю. С.,

чл.-корр. РАН, д. т. н., проф.

РУСАКОВ С. Г.,

чл.-корр. РАН, д. т. н., проф.

РЯБОВ Г. Г.,

чл.-корр. РАН, д. т. н., проф.

СОЙФЕР В. А.,

чл.-корр. РАН, д. т. н., проф.

СОКОЛОВ И. А., акад.

РАН, д. т. н., проф.

СУЕТИН Н. В., д. ф.-м. н., проф.

ЧАПЛЫГИН Ю. А.,

чл.-корр. РАН, д. т. н., проф.

ШАХНОВ В. А.,

чл.-корр. РАН, д. т. н., проф.

ШОКИН Ю. И.,

акад. РАН, д. т. н., проф.

ЮСУПОВ Р. М.,

чл.-корр. РАН, д. т. н., проф.

Редакционная коллегия:

АВДОШИН С. М., к. т. н., доц.

АНТОНОВ Б. И.

БАРСКИЙ А. Б., д. т. н., проф.

ВАСЕНИН В. А., д. ф.-м. н., проф.

ВИШНЕКОВ А. В., д. т. н., проф.

ГАЛУШКИН А. И., д. т. н., проф.

ДИМИТРИЕНКО Ю. И., д. ф.-м. н., проф.

ДОМРАЧЕВ В. Г., д. т. н., проф.

ЗАБОРОВСКИЙ В. С., д. т. н., проф.

ЗАГИДУЛЛИН Р. Ш., к. т. н., доц.

ЗАРУБИН В. С., д. т. н., проф.

КАРПЕНКО А. П., д. ф.-м. н., проф.

КОЛИН К. К., д. т. н., проф.

КУЛАГИН В. П., д. т. н., проф.

КУРЕЙЧИК В. М., д. т. н., проф.

КУХАРЕНКО Б. Г., к. ф.-м. н., доц.

ЛЬВОВИЧ Я. Е., д. т. н., проф.

МИХАЙЛОВ Б. М., д. т. н., проф.

НЕЧАЕВ В. В., к. т. н., проф.

ПОЛЕЩУК О. М., д. т. н., проф.

СОКОЛОВ Б. В., д. т. н., проф.

ТИМОНИНА Е. Е., д. т. н., проф.

УСКОВ В. Л., к. т. н. (США)

ФОМИЧЕВ В. А., д. т. н., проф.

ЧЕРМОШЕНЦЕВ С. Ф., д. т. н., проф.

ШИЛОВ В. В., к. т. н., доц.

Редакция:

БЕЗМЕНОВА М. Ю.

ГРИГОРИН-РЯБОВА Е. В.

ЛЫСЕНКО А. В.

ЧУГУНОВА А. В.

Информация о журнале доступна по сети Internet по адресу <http://novtex.ru/IT>.

Журнал включен в систему Российского индекса научного цитирования.

Журнал входит в Перечень научных журналов, в которых по рекомендации ВАК РФ должны быть опубликованы научные результаты диссертаций на соискание ученой степени доктора и кандидата наук.

INFORMATION TECHNOLOGIES

INFORMACIONNYYE TEHNOLOGII

Vol. 21
2015
No. 12

THEORETICAL AND APPLIED SCIENTIFIC AND TECHNICAL JOURNAL

Published since November 1995

ISSN 1684-6400

CONTENTS

INTELLIGENT SYSTEMS AND TECHNOLOGIES

- Kharlamov A. A., Yermolenco T. V.** Neuronetwork Environment (Neuromorphic Associative Memory) for Negotiation of Information Complexity. Meaning Identification in Semistructured Information Corporuses. Part II. Information Processing in Hippocampus. World Model 883
- Lelekov S. G., Lyakh A. M.** The Adaptive Fuzzy Inference System for Biological Taxonomy 890
- Dykin V. S., Musatov V. Yu., Varegnikov A. S., Bolshakov A. A., Sysoev V. V.** Application of Genetic Algorithm to Define an Architecture of Artificial Neural Network for Processing of Multisensor Signal in Electronic Nose. 895

MODELING AND OPTIMIZATION

- Kukhareenko B. G., Solntseva-Chalei M. O.** Spectral Method with Polar Curvature in Use for Analysis of Multi-Dimensional Trajectory Clustering Results. 901
- Moiseev A. A.** Mathematical Model of Auction Sale with Lots. 906

CAD-SYSTEMS

- Stempkovskiy A. L., Telpukhov D. V., Solovyev R. A., Rukhlov V. S.** Test System for Comparing Algorithms which Increases the Fault Tolerance of Combinational Circuits 910
- Petrosyants K. O., Kharitonov I. A., Kozhukhov M. V., Sambursky L. M.** Compact SPICE Models of Bipolar Transistors and MOSFETs for Automation of Radiation-Hardened VLSIs Design 916

DIGITAL PROCESSING OF SIGNALS AND IMAGES

- Chervyakov N. I., Lyakhov P. A., Kalita D. I.** Selection of the Optimal RNS Moduli to Improve FIR Filters Performance 923

CRYPTOSAFETY INFORMATION

- Shcheglov K. A., Shcheglov A. Yu.** Informational System Attack Threat Modeling and Interpretation. Part I. Vulnerability Threat Modeling and Attack Threat Interpretation 930

Journal-in-journal NEUROTECHNOLOGIES

- Galushkin A. I.** Neurochips and Neuromorphic Computers: a Modelling Challenges. . . 942
- Gorbatkov S. A., Kasimova L. I.** Neural Network Iterative Method for Constructing Dynamic Models of the Probability of Bankruptcy Risk at Incomplete Data . . . 949
- Index of articles published in the journal "Information Technology" in 2015. 958

Editor-in-Chief:

Stempkovsky A. L., Member of RAS,
Dr. Sci. (Tech.), Prof.

Deputy Editor-in-Chief:

Ivannikov A. D., Dr. Sci. (Tech.), Prof.
Filimonov N. B., Dr. Sci. (Tech.), Prof.

Chairman:

Bychkov I. V., Member of RAS,
Dr. Sci. (Tech.), Prof.
Zhuravljov Yu. I., Member of RAS,
Dr. Sci. (Phys.-Math.), Prof.
Kuleshov A. P., Member of RAS,
Dr. Sci. (Tech.), Prof.
Popkov Yu. S., Corresp. Member of RAS,
Dr. Sci. (Tech.), Prof.
Rusakov S. G., Corresp. Member of RAS,
Dr. Sci. (Tech.), Prof.
Ryabov G. G., Corresp. Member of RAS,
Dr. Sci. (Tech.), Prof.
Soifer V. A., Corresp. Member of RAS,
Dr. Sci. (Tech.), Prof.
Sokolov I. A., Member of RAS,
Dr. Sci. (Phys.-Math.), Prof.
Suetin N. V.,
Dr. Sci. (Phys.-Math.), Prof.
Chaplygin Yu. A., Corresp. Member of RAS,
Dr. Sci. (Tech.), Prof.
Shakhnov V. A., Corresp. Member of RAS,
Dr. Sci. (Tech.), Prof.
Shokin Yu. I., Member of RAS,
Dr. Sci. (Tech.), Prof.
Yusupov R. M., Corresp. Member of RAS,
Dr. Sci. (Tech.), Prof.

Editorial Board Members:

Avdoshin S. M., Cand. Sci. (Tech.), Ass. Prof.
Antonov B. I.
Barsky A. B., Dr. Sci. (Tech.), Prof.
Vasenin V. A., Dr. Sci. (Phys.-Math.), Prof.
Vishnekov A. V., Dr. Sci. (Tech.), Prof.
Galushkin A. I., Dr. Sci. (Tech.), Prof.
Dimitrienko Yu. I., Dr. Sci. (Phys.-Math.), Prof.
Domrachev V. G., Dr. Sci. (Tech.), Prof.
Zaborovsky V. S., Dr. Sci. (Tech.), Prof.
Zagidullin R. Sh., Cand. Sci. (Tech.), Ass. Prof.
Zarubin V. S., Dr. Sci. (Tech.), Prof.
Karpenko A. P., Dr. Sci. (Phys.-Math.), Prof.
Kolin K. K., Dr. Sci. (Tech.)
Kulagin V. P., Dr. Sci. (Tech.), Prof.
Kureichik V. M., Dr. Sci. (Tech.), Prof.
Kukhareenko B. G., Cand. Sci. (Phys.-Math.)
Ljovich Ya. E., Dr. Sci. (Tech.), Prof.
Mikhailov B. M., Dr. Sci. (Tech.), Prof.
Nechaev V. V., Cand. Sci. (Tech.), Ass. Prof.
Poleschuk O. M., Dr. Sci. (Tech.), Prof.
Sokolov B. V., Dr. Sci. (Tech.)
Timonina E. E., Dr. Sci. (Tech.), Prof.
Uskov V. L. (USA), Dr. Sci. (Tech.)
Fomichev V. A., Dr. Sci. (Tech.), Prof.
Chermoshentsev S. F., Dr. Sci. (Tech.), Prof.
Shilov V. V., Cand. Sci. (Tech.), Ass. Prof.

Editors:

Bezmenova M. Yu.
Grigorin-Ryabova E. V.
Lysenko A. V.
Chugunova A. V.

Complete Internet version of the journal at site: <http://novtex.ru/IT>.

According to the decision of the Higher Certifying Commission of the Ministry of Education of Russian Federation, the journal is inscribed in "The List of the Leading Scientific Journals and Editions wherein Main Scientific Results of Theses for Doctor's or Candidate's Degrees Should Be Published"

УДК 519.1

А. А. Харламов, д-р техн. наук., ст. науч. сотр.; e-mail: kharlamov@analyst.ru,
Институт высшей нервной деятельности и нейрофизиологии РАН, г. Москва,
Т. В. Ермоленко, канд. техн. наук, нач. отдела, e-mail: etv@iai.dn.ua,
Институт проблем искусственного интеллекта, Украина, г. Донецк

Нейросетевая среда (нейроморфная ассоциативная память) для преодоления информационной сложности. Поиск смысла в слабоструктурированных массивах информации. Часть II. Обработка информации в гиппокампе. Модель мира

Ассоциативная память человека является средством для преодоления информационной сложности, возникавшей и возникающей постоянно в процессе его жизнедеятельности. Рассмотрены архитектура, свойства и функциональность ассоциативной памяти в целях ее использования для моделирования способности преодоления информационной сложности. Описан процесс перенормировки весовых характеристик образов событий, хранящихся в иерархии колонок коры полушарий большого мозга человека, который происходит в ламелях гиппокампа. Даются представления о модели мира человека, формирующейся на основе иерархий словарей разных уровней разных модальностей в коре. Показан изоморфизм лингвистической модели мира и многомодальной модели мира человека.

Ключевые слова: кортикоморфная ассоциативная память, структурная обработка информации, ассоциативная (однородная семантическая) сеть, переранжирование понятий, модель мира, классификация, обработка текстов

Введение

В первой из двух частей работы, которая была опубликована ранее, освещены вопросы формирования когнитивных сетей в мозге человека и использование технологии их формирования для автоматического анализа текстов. В ней были представлены механизмы формирования ассоциативных (однородных семантических) сетей, в том числе формирования словарей образов событий различной частоты встречаемости, а также выявления связей между этими образами во входной информации. В качестве примера обработки информации с использованием этих механизмов была представлена обработка текстовой информации. Во второй части работы описываются механизмы переранжирования весовых характеристик, выявленных на предыдущем этапе образов событий. Описана в структурных терминах модель мира человека, состоящая из трех компонентов: лингвистического и двух многомодальных. Показан изоморфизм многомодальных и лингвистического компонентов модели мира. Представлены механизмы сравнения и классификации текстов, в том числе естественно-языковых текстов.

1. Обработка информации в гиппокампе

Кроме коры больших полушарий другой структурой мозга, существенно важной для формирования модели мира, является гиппокамп. Ламели гиппокампа (сечения, ортогональные к длинной оси гиппокампа) ответственны за хранение информации о связях образов событий, хранящихся в колонках коры, в рамках целых ситуаций. Пирамидные нейроны (теперь уже) поля CA₃ *p*-й ламели гиппокампа формируют искусственную нейронную сеть Хопфилда [1], весовые характеристики которой хранят информацию об объединении образов событий, хранящихся в колонках коры, относящихся к конкретной ситуации, в рамках этой ситуации:

$$N_p = \cup_i B_i. \quad (1)$$

Знак крышки ($\wedge$) над соответствующими событиями в (1) отсутствует потому, что в ламелях гиппокампа в сеть объединяются не элементы словарей — фрагменты траекторий в многомерном пространстве, а их индексы.

Ламели гиппокампа получают информацию от колонок коры [2], причем и здесь работает ассоциативный принцип обращения к информации. Весь поток информации, поступающий через переключе-

ния из коры, приходит одновременно на все ламели гиппокампа. Но откликаются на этот поток только те ламели, которые содержат информацию о событиях, образы которых присутствуют во входном потоке. Отклик тем больше, чем сильнее ассоциация и чем больше вес образов событий в колонках коры.

На каждой итерации взаимодействия коры и гиппокампа поле CA_1 гиппокампа (как конкурентная сеть) формирует отклик только одной ламели гиппокампа — наиболее близкой к входной ситуации.

Но на этом дело не кончается: в результате отклика текущей ламели в коре в колонке, инициировавшей процесс, происходит дообучение (в результате так называемой долговременной потенциации). И на следующей итерации ассоциативная проекция той же ситуации на ламели гиппокампа оказывается измененной из-за этого дообучения, и следующий отклик ламелей изменяется.

После 15...20 итераций образы событий, включенных в ситуацию, в колонке коры изменяются из-за дообучения (которое связано с моделями ситуаций, хранящихся в ламелях гиппокампа). Вообще говоря, и модели ситуаций в ламелях гиппокампа изменяются также. То есть этот итеративный процесс переупорядочивает информацию в коре о событиях текущей ситуации в соответствии с имеющимися моделями ситуаций, хранящимися в гиппокампе, а эти модели ситуаций принимают к сведению информацию о текущей ситуации, отображенной в виде образов событий в коре.

А поскольку поле CA_3 гиппокампа работает в виде громадной автоассоциативной рекуррентной памяти по всей его длине и ширине, множество отдельных моделей ситуаций N_p , которые хранятся в ламелях p , совместно с образами событий, хранящихся в колонках коры, формируют единую семантическую сеть N на многомодальной модели мира, хранящейся в колонках коры:

$$N = \bigcup_p N_p. \quad (2)$$

При этом важна не дифференцировка хранимых образов, это обеспечивается в коре, а контекстуальные пространственно-временные связи образов в рамках целых ситуаций.

1.1. Семантическая сеть

Опишем более формально ассоциативную (однородную семантическую) сеть N , а также рассмотрим ее как некое подмножество метрического пространства.

Определение 1. Под семантической сетью N понимается множество несимметричных пар событий $\{<c_i c_j>\}$, где c_i и c_j — события, связанные между собой отношением ассоциативности (совместной встречаемости в некоторой ситуации):

$$N \approx \{<c_i c_j>\}.$$

В данном случае отношение ассоциативности несимметрично:

$$<c_i c_j> \neq <c_j c_i>.$$

Определение 2. Семантическая сеть, описанная таким образом, может быть представлена как множество так называемых звездочек $\{<c_i < c_j>>\}$:

$$N \approx \{z_i\} = \{<c_i < c_j>>\}.$$

Определение 3. Под звездочкой $<c_i < c_j>>$ понимается конструкция, включающая главное событие c_i , связанное с множеством событий-ассоциантов $<c_j>$, которые являются семантическими признаками главного события, отстоящими от главного события на одну связь. Связи направлены от главного события к событиям-ассоциантам.

Определение 4. Звездочка с единичными значениями весов событий-ассоциантов называется единичной звездочкой (звездочкой-ортом).

Определение 5. Звездочкой-подпространством называется звездочка, полученная на единичной звездочке введением весов событий (w_j — веса событий):

$$Z \approx <c_i < w_j c_j>>.$$

Определение 6. Весом w_j образа события c_j в сети является значение счетчика повторных появлений событий во входном тексте.

Семантическая сеть в терминах этих определений представляет собой декартово произведение подпространств, порождаемых всеми звездочками, входящими в семантическую сеть, полученными на единичных звездочках за счет введения весовых характеристик понятий-ассоциантов:

$$N = Z_1 \times Z_2 \times \dots \times Z_l.$$

Для введения в метрическом пространстве, определенном на семантической сети операций вложения и пересечения, разобьем звездочки на пары событий $<c_i c_j>$. Здесь c_i — главное событие звездочки, а $c_j \in \langle c_j \rangle$ — множество событий-ассоциантов события c_i . Связь направлена от c_i к c_j .

Все сказанное относится к когнитивной семантической сети, т. е. семантической сети, формируемой на нейронной сети как на субстрате [3]. Отчужденная от субстрата когнитивная семантическая сеть остается той же ассоциативной сетью, которая представлена в определениях 1—6. В этом случае множество событий $\{c_j\}$ — это индекс (квализлов A квазисловаря) квазитекста (в том числе и обычного текста), а сеть N — ассоциативная сеть, построенная на этом индексе.

1.2. Перенормировка семантических весов

При формировании сети на основе большого корпуса квазитекстов, каковыми являются все входные последовательности всех модальностей, получают некоторые весовые характеристики w_j образов событий c_j . При этом частота их встречаемости приближается к их смысловому весу. При анализе

малых по объему квазитекстов, каковыми являются отдельные ситуации, частота встречаемости уже не характеризует важности образа события.

По аналогии с перевзвешиванием образов событий в коре под воздействием результатов обработки информации в гиппокампе разумно каким-то образом менять весовые характеристики образов событий ассоциативной сети с учетом их связей с соседними вершинами сети. При такой перенормировке на каждой итерации образы событий, связанные с образами, имеющими большой вес, должны свой вес увеличивать. Другие события должны равномерно терять свой вес.

Сформированное первоначально статистическое представление квазитекста — сеть образов событий с их связями (в виде частоты встречаемости событий и связей между ними) — перенормируется, что позволяет перейти от частотного портрета квазитекста (сети с частотами) к ассоциативной сети ключевых образов событий квазитекста (сети с рангами этих событий — их смысловыми весами):

$$w_i(t+1) = \left(\sum_{i \neq j} w_i(t) w_{ij} \right) \sigma(\bar{E}),$$

здесь $w_i(0) = z_i$; $w_{ij} = z_{ij}/z_j$ и $\sigma(\bar{E}) = 1/(1 + e^{-k\bar{E}})$ — функция, нормирующая на среднее значение энергии всех вершин сети $\bar{E}$; z_i — частота встречаемости i -го слова в тексте; z_{ij} — частота совместной встречаемости i -го и j -го слов в фрагментах текста; t — номер итерации. Полученная числовая характеристика слов — их смысловой вес — характеризует степень их важности в тексте.

2. Автоматическое структурирование больших объемов разнородной информации

Используя приведенный выше формализм обработки информации, представленной в виде внутренне структурированных последовательностей (квазитекстов и естественных языковых текстов в том числе), мы можем структурировать произвольные потоки информации различных модальностей. Мы можем формировать словари $\{B_i\}$ образов событий разных модальностей разной частоты встречаемости (разной сложности). Мы можем формировать из входных последовательностей $\{A\}$ последовательности $\{C\}$ связей слов словарей. Таким образом, мы реализуем структурный анализ информации. Далее мы можем формировать весовые характеристики w_i слов этих словарей в зависимости от их смыслового (структурного) веса в конкретных квазитекстах (в естественных языковых текстах в том числе). Другими словами, мы можем формировать смысловой портрет квазитекста (в том числе и естественного языкового текста) в виде семантической сети со взвешенными (ранжированными) вершинами.

Пользуясь этими механизмами, мы можем перейти к пониманию структуры модели мира чело-

века, которая включает в себя лингвистический и многомодальные компоненты. И далее, мы можем сравнивать эти тексты по смыслу, вычисляя степень пересечения их семантических сетей. И, следовательно, классифицировать квазитексты, сравнивая их смысловые портреты со смысловыми портретами классов — моделей предметных областей.

2.1. Модель мира — иерархическая структура словарей образов событий разной сложности и разных модальностей

Рассмотрим формирование с помощью описанных механизмов представления о мире в терминах одной модальности. Рассмотрим иерархическую структуру словарей образов событий мира, имеющую вид, представленный на рис. 1, где на каждом уровне имеется множество словарей образов событий, связанных с образами событий следующего уровня по типу каждый с каждым. В каждом подсловаре $\{\hat{B}_i\}_{jkm}$ i — слово в подсловаре, j — номер подсловара на уровне, k — номер уровня, а m — номер модальности.

События внешнего (и внутреннего) мира, отображающиеся сенсорами в иерархическую структуру данной (m -й) сенсорной модальности, формируют в этой иерархической структуре иерархию словарей, слова которых оказываются связанными между собой синтаксическими последовательностями. При этом синтаксические последовательности, сформированные в процессе обработки предыдущего уровня, являются исходными для формирования словарей следующего уровня. Эта иерархия словарей является моделью мира в терминах m -й модальности.

Модель мира данной модальности — представляет собой объединение слов словарей $\hat{M}_m$. На верхнем (K -м) уровне в нем объединяются все слова $\hat{B}_{ijkK}$ словаря (подсловарей) верхнего уровня, а на всех более низких уровнях в него включаются (в соответствующие места — по ассоциативному принципу) слова $\hat{B}_{ijkm}$ словарей нижних уровней:

$$\hat{M}_m = \bigcup_{ij} \bigvee_{ijk < K} \hat{B}_{ijkm}. \quad (3)$$

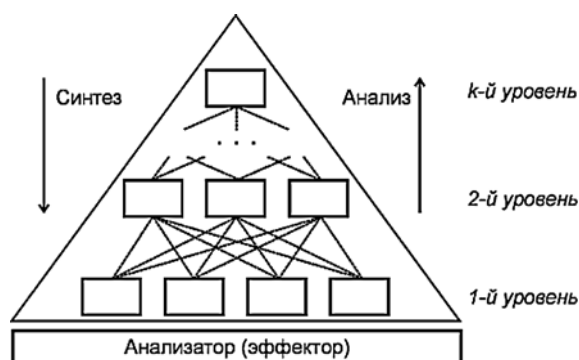


Рис. 1. Многоуровневая иерархическая структура из словарей образов событий одной модальности

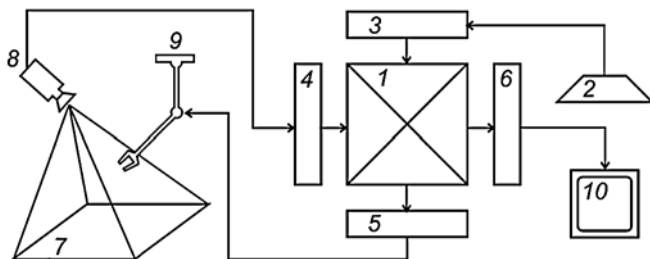


Рис. 2. Объединение иерархий для многомодального представления знаний в антураже задачи "Ханойская башня" (интегральный робот):

1 — многомодальная иерархия из словарей разных уровней разных модальностей; 2 — клавиатура; 3 — сенсорная подсистема ввода текстовой информации; 4 — сенсорная подсистема ввода зрительной информации, предварительно обработанной и развернутой, полученной с модели сцены 7, наблюдаемой светочувствительной матрицей 8; 5 — эффекторная подсистема управления манипулятором 9; 6 — эффекторная подсистема синтеза текста с выдачей на экран видеотерминала 10

Здесь $\cup$ — операция объединения, а операция $\vee$ обозначает включение на свое место в слова словаря более высокого уровня. Это включение аналогично логическому сложению последовательности, соответствующей слову словаря нижнего уровня, с синтаксической последовательностью (вложению слова словаря в соответствующую купюру синтаксической последовательности).

Объединив несколько иерархических структур, соответствующих разным сенсорным модальностям (рис. 2) со сформированными на них модальными моделями мира $\hat{M}_m$, мы получим многомодальную модель мира:

$$\hat{M} = \cup_m \hat{M}_m = \cup_{ijm} \hat{B}_{ijkm} \vee_{ijk < Km} \hat{B}_{ijkm}. \quad (4)$$

2.2. Формирование многомодального многоуровневого представления

Любое многомодальное представление можно рассматривать как квазитекстовое, поскольку оно имеет многоуровневую внутреннюю структуру. Поэтому рассмотрим формирование такого представления у человека на примере обработки текстовой информации. Язык, представленный в виде множества текстов, с помощью описанного выше формализма подвергается структурному анализу, в результате которого выявляются его словарные компоненты разных уровней.

При обработке текстов обычно рассматриваются несколько уровней обработки информации: морфологический, лексический, синтаксический и семантический. На каждом из уровней возможно формирование нескольких типов словарей. Мы рассмотрим только некоторые из них: на морфологическом уровне — словарь флексивных морфем; на лексическом уровне — словарь корневых основ; на синтаксическом уровне — словарь синтаксем, представляющих собой флексивную структуру синтаксических групп с выколотыми корневыми ос-

новами; на семантическом уровне — словарь попарной сочетаемости корневых основ [4].

Рассмотрим последовательные этапы формирования вышеперечисленных словарей. Сначала формируется словарь флексивных морфем $\{\hat{B}\}_1$, так как они являются наиболее часто встречающимися языковыми единицами.

На следующем этапе формируется словарь корневых основ слов. После того как сформировался словарь флексивных морфем, фильтрация словарем флексивных морфем множества текстов приводит к формированию словаря корневых основ $\{\hat{B}\}_2$, так как в результате взаимодействия множества текстов со словарем флексивных морфем возникает множество синтаксических последовательностей с купюрами вместо флексий — множество последовательностей корневых основ.

Далее формируется словарь синтаксического уровня $\{\hat{B}\}_3$. Этот словарь формируется фильтрацией фрагментов текста, соответствующих по длине синтаксической группе, через словарь корневых основ слов. Полученные при этом цепочки флексивных морфем кластеризуются на подклассы по их графемной структуре. Эти подклассы являются частями синтаксических классов, соответствующих основным синтаксическим группам.

После того как сформирован словарь синтаксем, формируется словарь попарной (смысловой) сочетаемости слов [4] $\{\hat{B}\}_4$, которая определяет семантику. Для простоты будем считать, что именно слова словарей этого уровня объединяются в семантическую сеть N :

$$N = \cup_i B_i,$$

здесь слова естественного языка B_i соответствуют словам-траекториям $\hat{B}_i$ в многомерном пространстве словарей $\{\hat{B}_i\}_4$.

2.3. Модель мира.

Языковой и многомодальные компоненты

Модель мира человека представляет собой совокупность модальных компонентов, которые являются множеством образов ситуаций, хранящихся в ламелях гиппокампа.

Однако существует более точно очерченная функциональность многомодальных компонентов модели мира, которая связана с латерализацией полушарий, зависящей от представительства в коре речевых функций. В левом полушарии под воздействием социума формируется лингвистическая модель мира $\hat{M}_l$ (3) в терминах естественного языка.

Таким образом, в субдоминантном (правом у правой) полушарии локализован многомодальный образный компонент модели мира $\hat{M}_R$ (4), а в доминантном (соответственно, левом у правой) полушарии — локализованы многомодальный схематический $\hat{M}_L$ (4) и лингвистический $\hat{M}_l$ (3) компоненты модели. Такое распределение не случайно:

в связи с наличием в левом полушарии лингвистической модели мира $\hat{M}_L$ социум через язык как сегментирующую функцию влияет на формирование левополушарного многомодального компонента $\hat{M}_L$. А образный компонент правого полушария $\hat{M}_R$ формируется под воздействием индивидуальной истории развития конкретного индивидуума (конечно, используя знания левополушарного компонента). При этом схематические образы событий левополушарной многомодальной схематической модели $\hat{M}_L$ наполняются содержанием — соответствующими образами событий правополушарной образной многомодальной модели $\hat{M}_R$.

Все три компонента имеют поуровневые связи и потому работают как единое целое:

$$\hat{M} = \hat{M}_L \cup \hat{M}_L \cup \hat{M}_R.$$

Ключевые понятия лингвистической модели мира соответствуют ключевым образам событий многомодальной модели, т. е. лингвистическая модель изоморфна многомодальной модели мира.

Таким образом, многомодальная модель, представленная в виде объединенной семантической сети $\hat{M}_L \cup \hat{M}_R$, имеет левополушарный языковой эквивалент (также в виде сети) $\hat{M}_L$: многомодальным образам событий мира в их взаимосвязях в ней сопоставляются понятия (слова, словосочетания) с их соответствующими связями. Поэтому операции на языковой модели можно считать отражением операций на многомодальной модели (мы можем оценить операции над многомодальной моделью, моделируя их на лингвистической модели). Для этого сопоставим многомодальной модели (4) соответствующую сеть (2). Сеть же для языковой модели (3) сформируем аналогично (1) и (2), но только для лингвистической модальности $m = l$:

$$N_{pl} = \bigcup_{ijklp} B_{ijklp}.$$

Здесь индекс l означает лингвистическую модальность:

$$N_l = \bigcup_p N_{pl}.$$

2.4. Сравнение семантических сетей. Классификация

Поскольку далее нам придется выявлять смысл текстов их соотношением с моделями предметных областей, рассмотрим вопросы сравнения текстов по смыслу и, далее, классификацию текстов. Все, что будет сказано ниже, в равной степени относится как к семантическим сетям и прочим упоминавшимся выше конструкциям в текстах (и квазитекстах), так и к когнитивным представлениям в многомерном (когнитивном) пространстве (которое моделируется колонками коры и ламелями гиппокампа).

Определение 7. Введем скалярное произведение на векторах c_i, c_j , где угол между векторами по-

нятий c_i, c_j : w_{ij} пропорционален весу связи от c_i к c_j : $w_{ij} \in (0...90^\circ)$.

Площадь треугольника s_j , построенного на векторах c_i, c_j , развернутых на угол w_{ij} относительно друг друга, будет использована для вычисления степени пересечения сначала звездочек, а потом семантических сетей как совокупностей звездочек.

Определение 8. Под пересечением двух звездочек понимается сумма по всем событиям-ассоциантам данного главного события звездочки пересечений площадей двух треугольников, построенных в плоскости векторов c_i, c_j , один из которых построен на векторах, развернутых на угол, пропорциональный связи $(w_{ij})_1$ между событиями в одной звездочке, а другой — на угол, пропорциональный связи $(w_{ij})_2$ между теми же событиями в другой, сравниваемой с первой, звездочке. В случае если в одной из звездочек пары, для которой считается пересечение

$$s_{12} = \langle c_{i_1} \langle c_{j_1} \rangle \rangle \cap \langle c_{i_2} \langle c_{j_2} \rangle \rangle = \sum_{j=1}^{\max(N_1 N_2)} (s_{j_1} \cap s_{j_2}),$$

не нашлось соответствующего события-ассоцианта, пересечение считается равным 0. Здесь N_1, N_2 — число ассоциантов в звездочках соответственно 1 и 2.

Определение 9. Под пересечением семантических сетей понимается сумма пересечений звездочек, включенных в эти сети (считая по главным понятиям):

$$S_{12} = \langle c_{i_1} c_{j_1} \rangle \cap \langle c_{i_2} c_{j_2} \rangle = \sum_{k=1}^{\max(M_1 M_2)} (s_{k_1} \cap s_{k_2}),$$

M_1, M_2 — число звездочек, входящих соответственно в семантические сети 1 и 2.

Определение 10. Под классификацией входной ситуации можно понимать отнесение семантической сети ситуации N к сети N_n (где $n = 1, \dots, N$ — число предметных областей) одной из предметных областей модели мира. Здесь объединение сетей $\bigcup_n N_n$ соответствует модели мира $\hat{M}_m$ m -й модальности. В идеальном случае семантическая сеть ситуации вкладывается в сеть соответствующей предметной области.

Используя операцию пересечения сетей N_1 и N_2 , определенную выше, мы можем оценивать степень подобия двух сетей $N_1 \cap N_2$ (рис. 3) и, тем самым, сравнивать по смыслу (по структуре) ситуации (их модели). Имея модели предметных областей в виде ассоциативных семантических сетей, мы можем классифицировать входные ситуации (описывающие их модели) вычислением степени совпадения (вложения) сети входной ситуации и сетей предметных областей (рис. 4), относя входную ситуацию к той предметной области, у которой степень

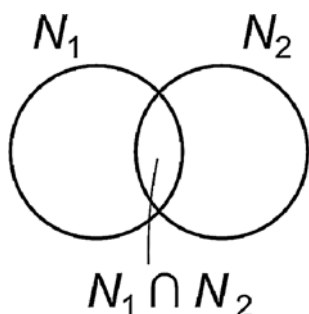


Рис. 3. Пересечение $N_1 \cap N_2$ двух сетей N_1 и N_2 , характеризующее степень их смыслового подобия

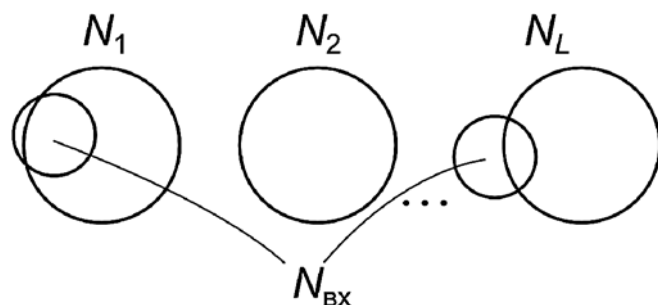


Рис. 4. Классификация входного текста путем выявления степени вложенности его семантической сети $N_{вх}$ в одну или несколько семантических сетей классов-рубрик — предметных областей $N_1, N_2, \dots, N_L$

совпадения сети входной ситуации с сетью предметной области окажется выше.

Далее мы можем абстрагироваться от входных ситуаций. Будем рассматривать их языковые модели. Пойдем еще дальше: перейдем к анализу текстов, на основе которых эти языковые модели формируются. Как и в случае классификации входных ситуаций, используя операцию пересечения сетей, мы можем оценивать степень подобия двух сетей и, тем самым, сравнивать по смыслу уже тексты. Имея модели предметных областей N_n в виде ассоциативных семантических сетей соответствующих тематических текстовых выборок, мы можем классифицировать входные тексты вычислением степени совпадения (пересечения/вложения) сети N входного текста и сетей предметных областей N_n , относя входной текст к той предметной области, у которой степень совпадения его сети с сетью предметной области окажется выше.

Покажем далее как, используя все выше перечисленные механизмы, подойти к понятию сложности чего-либо: некоторого явления (например, ситуации) в виде последовательности взаимосвязанных многомодальных образов, т. е. квазитекстов.

3. Сложность квазитекста как отражения некоторой ситуации

Попробуем оценить сложность входной информации, учитывая число модальностей m (отдельно

выделим лингвистическое представление l), представленных в ней, число формируемых уровней k словарей, число словарей j на уровнях, число слов i в словарях. Поскольку эта информация структурируется в процессе ее обработки в виде семантических сетей, удобно найти способ оценки ее сложности с учетом ключевых понятий сети, имеющих ранг выше заданного. Можно, например, определять число существенно важных тем в семантической сети, а также — число ассоциантов понятий в сети.

Определение 11. Под тематической структурой текста будем понимать минимальный древовидный подграф $T = \langle K, C, R, D, W, V \rangle$, извлеченный из сети N , где в качестве корневой вершины берется вершина с максимальным рангом, а остальные вершины ранжируются в соответствии с их смысловыми весами. Здесь K — корневая вершина, C — множество вершин в сети, R — множество связей этих вершин в семантической сети, N, D — конкретный способ связей вершин в конкретной сети, N, W и V — весовые характеристики вершин и связей соответственно. Минимальный древовидный подграф получается удалением слабых связей V (с весом ниже заданного порога h_p) и слабых вершин (также ниже заданного порога h_c).

Оперируя числом тем, подтем и подподтем в тематических структурах текстов, а также числом ассоциантов вершин семантической сети, можно оценивать сложность текстов и сравнивать тексты по сложности. Поскольку любая вершина сети имеет ближайшие соседи-ассоцианты, которые являются ее семантическими признаками, по числу ассоциантов можно определять сложность текста в шири-

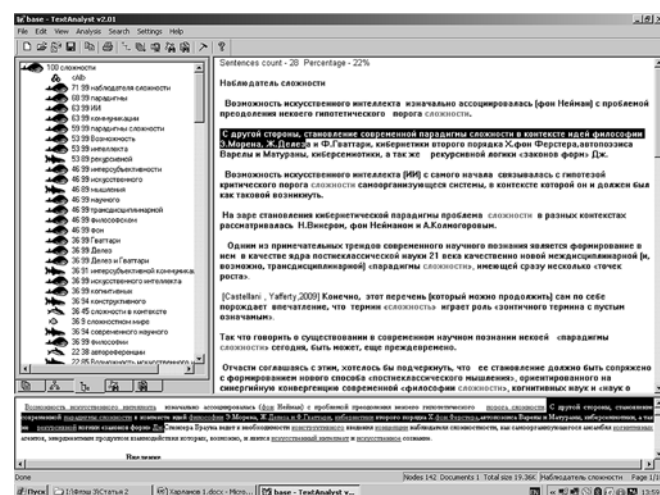


Рис. 5. Ближайшие ассоцианты конкретной вершины семантической сети, определяющие сложность текста в ширину (в левом окне) в тексте статьи В. И. Аршинова "Наблюдатель сложности как модель искусственного интеллекта" на графическом интерфейсе программы TextAnalyst. Вверху — корневая вершина, ниже справа — ассоцианты

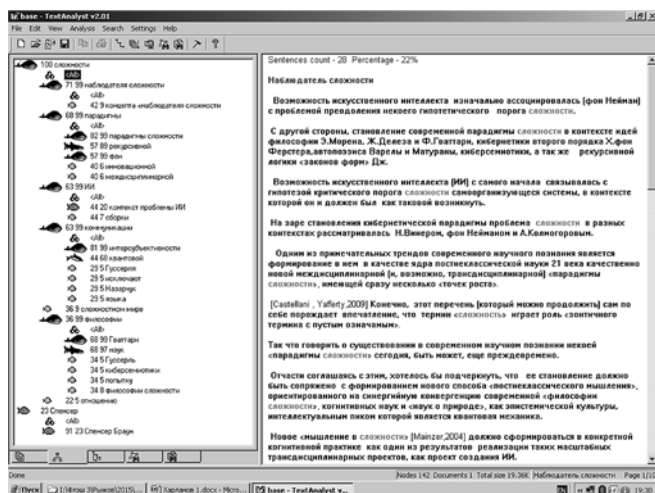


Рис. 6. Тематическая структура текста, определяющая сложность текста в глубину (в левом окне) в тексте статьи В. И. Аршинова "Наблюдатель сложности как модель искусственного интеллекта" на графическом интерфейсе программы TextAnalyst. Тематическое дерево имеет две корневые вершины — "Сложность" и "Спенсер". В первом дереве — семь подтем: "Наблюдатель сложности", "Парадигмы" и т. п. Наличие второй корневой вершины говорит о том, что в тексте есть отдельная тема, посвященная Спенсеру

ну (рис. 5), а по числу уровней подтем у главной темы — определять сложность текста в глубину (рис. 6).

Заключение

Представленные во второй части работы механизмы обработки информации в мозге человека,

в том числе в ламелях гиппокампа, связанные с переранжированием весовых характеристик образов событий, хранящихся в памяти, позволяют выявлять наиболее важные ключевые образы, что является наиболее существенным моментом при анализе информации. Изоморфизм лингвистического и много-модальных компонентов модели мира человека, которая строится на основе описанных в первой и второй частях работы механизмах, позволяет интерпретировать картину мира на основе анализа текстовых представлений. Описанные механизмы сравнения и классификации текстов являются основой для такой интерпретации. В работе также тезисно представлены соображения об анализе сложности текстового материала, которые будут подробно изложены в следующей публикации.

Работа выполнена при поддержке РГНФ (проект № 15-03-00860 "Методология управления сетевыми структурами в контексте парадигмы сложности").

Список литературы

1. Hopfield J. J. Neural networks and physical systems with emergent collective computational abilities // Proc. Natl. Acad. Sci. 1982, Vol. 79. P. 2554—2558.
2. Brown T. H., Zador A. M. Hippocampus / G. M. Shepherd (Ed.), The synaptic organisation of the brain. New York, Oxford: Oxford University Press, 1990. P. 346—388.
3. DARPA SyNAPSE Program. URL: <http://www.artificial-brains.com/darpa-synapse-program>.
4. Рахилина Е. В. Когнитивный анализ предметных имен: семантика и сочетаемость. М.: Русские словари, 2000.

A. A. Kharlamov, Senior Researcher, e-mail: kharlamov@analyst.ru,
Institute of Higher Nervous Activity and Neurophysiology of Russian Academy of Sciences, Moscow,
T. V. Yermolenco, Head of Department, e-mail: etv@iai.dn.ua,
Institute of Artificial Intelligence Problems, Ukraine, Donetsk

Neuronetwork Environment (Neuromorphyc Associative Memory) for Negotiation of Information Complexity. Meaning Identification in Semistructured Information Corpuses. Part II. Information Processing in Hippocampus. World Model

Human associative memory is a tool for negotiation of information complexity which is a result of its living. In the paper shown an architecture, properties and functions of associative memory for its using in negotiation of information complexity. A rearranging process of weight characteristics of input event images stored in columns hierarchy of hemisphere of human cortex which is fulfilled in hippocampus lamellas are shown too. A representation of human world model in cortex which are formed on base of several levels and several modalities vocabularies hierarchy is shown. Isomorphism of linguistic and multimodal human world models is represented too.

Keywords: korticomorphic associative memory, structural information processing, associative (homogeneous semantic) network, concept rearranging, world model, classification, text processing

References

1. Hopfield J. J. Neural networks and physical systems with emergent collective computational abilities, Proc. Natl. Acad. Sci., 1982, vol. 79, pp. 2554—2558.
2. Brown T. H., Zador A. M. Hippocampus. G. M. Shepherd (Ed.), The synaptic organisation of the brain. New York, Oxford: Oxford University Press, 1990, pp. 346—388.
3. DARPA SyNAPSE Program. URL: <http://www.artificial-brains.com/darpa-synapse-program>.
4. Rakhilina E. V. Kognitivnyi analiz predmetnikh imen: semantika i sochetaemost, M.: Russkiye slovari, 2000.

С. Г. Лелеков, канд. физ.-мат. наук, доц., e-mail: p1859@yandex.ru,
 Севастопольский государственный университет,
 А. М. Лях, канд. биол. наук, ст. науч. сотр., e-mail: antonlyakh@yahoo.com,
 Институт биологии южных морей, г. Севастополь

Нечеткий вывод в адаптивных таксономических экспертных системах

Семантическая неоднозначность естественного языка вызывает проблему представления и манипулирования знаниями в таксономических экспертных системах. Для представления признаков определяемых объектов предлагается использовать лингвистические переменные с нечеткими значениями. Рассматриваются способы организации нечеткого вывода в экспертных системах.

Ключевые слова: экспертные системы, лингвистические переменные, нечеткий вывод, биологическая таксономия, компьютерные определители

Введение

Любая идентификация (определение) организмов, в том числе компьютерная, основывается на данных о признаках объектов и их таксономическом статусе. Поэтому компьютерные таксономические базы данных (БД) и атласы организмов — это наиболее распространенный и активно используемый сетевой ресурс. Они содержат списки таксонов определенной группы, например, *algaebase.org* — таксономическая БД о водорослях, *marinespecies.org* — таксономическая БД морских организмов, *tolweb.org* — таксономическая БД всех организмов, *fishbase.org* — описание рыб Мирового океана и др. Эти сетевые ресурсы не являются экспертными системами, так как не содержат правил определения и не предоставляют интерфейс для идентификации. Они формируют базу таксономических знаний, которая может быть использована для создания компьютерных определителей.

Современные компьютерные определители применяют полиномические одно- или многовходовые ключи, заданные таблицами признаков. Процесс определения является пошаговым. Признаки выбираются алгоритмами системы на основе их диагностической ценности, которая рассчитывается по таблице признаков. Процедура идентификации продолжается до опознавания таксона или исчерпания признаков. Результатом является перечень таксонов, у которых признаки частично или полностью совпадают с введенными пользователем значениями. Интуитивно понятный механизм взаимодействия с пользователем позволяет корректно идентифицировать организмы и не специалисту.

Такой подход реализован в системах "Lucid" (*lucidcentral.org*), "Delta-IntKey" (*delta-intkey.com*), "Taxex"[1—4].

Однако существующие таксономические экспертные системы пока не находят широкого применения. Одной из причин этого является семан-

тическая неоднозначность естественного языка, используемого экспертами.

При составлении диагнозов таксонов эксперты используют своеобразную терминологию, которая вплетается в живой разговорный язык. Диагнозы часто содержат размытые формулировки, понятные автору, но не понятные стороннему читателю, например, признак "Форма головы" со значениями: "Большая массивная", "Небольшая", "Маленькая". Пользователь вынужден принимать терминологию эксперта, хотя его представление о сущности того или иного термина может отличаться от смысла, который вложил в него эксперт. Проблема усугубляется тем, что общение пользователя и эксперта проходит опосредованно через экспертную систему.

Частичным решением этой проблемы может стать использование адаптивных таксономических экспертных систем [5], в которые закладываются механизмы учета семантической неоднозначности естественного языка. Математической основой функционирования таких систем является теория нечетких множеств. Адаптивность, в этом контексте, следует понимать как приспособляемость системы к нечеткой интерпретации терминов пользователем и экспертом в процессе таксономической идентификации объекта.

Постановка задачи

В заданной предметной области имеется конечное множество визуально распознаваемых объектов. Все объекты описаны множеством признаков. Описания выполнены на естественном языке и, как следствие, содержат нечеткие, расплывчатые формулировки.

Требуется разработать автоматизированную диагностическую систему, способную в диалоговом режиме работы с пользователем идентифицировать предъявляемый ей объект путем последовательного выяснения значений признаков. Такие системы в

биологии получили название *адаптивных таксономических экспертных систем* (АТЭС).

Рассматривается решение задачи сочетания неоднозначности описаний биологических объектов, их толкования и способов организации логического вывода в рамках компьютерных адаптивных таксономических экспертных систем.

Эта задача включает две подзадачи:

- описание определяемых объектов в базе признаков;
- использование описаний в процессе логического вывода.

В статье предложено решение в основном второй подзадачи. Решение первой задачи более подробно рассмотрено в работе [6].

Описание определяемых объектов в базе признаков

База знаний АТЭС включает: описания признаков, описания таксонов и набор правил вывода заключительного диагноза.

Простейшим вариантом представления правил вывода является форма:

ЕСЛИ " α есть β ", ТО " γ есть δ ".

При этом " α есть β " и " γ есть δ " являются нечеткими высказываниями [7].

В рассматриваемом классе экспертных систем (ЭС) α является таксономическим признаком, β — его значением, γ — неизвестным определяемым объектом, δ — подмножеством известных системе таксонов. Наибольшие трудности при переносе знаний эксперта в базу знаний ЭС связаны с формированием машинного представления α и β .

Представление признаков реализуется в двух формах:

- внешней, в виде веб-страницы с текстом и рисунками, отражающими значения признака;
- внутренней, в которой учитывается семантическая неопределенность естественного языка.

Для внутреннего представления предлагается использовать аппарат нечетких множеств и понятие лингвистической переменной [4].

Лингвистическая переменная определяется как

$$L = (\chi, T, U, M),$$

где χ — название лингвистической переменной; T — совокупность лингвистических значений L , выраженных нечеткими переменными; U — универсальное множество (область определения T); M — семантические правила, которые каждому лингвистическому значению $x \in T$ ставят в соответствие нечеткое подмножество $M(x)$, которое может быть задано функцией принадлежности $\mu: U \rightarrow [0, 1]$.

Рассмотрим особенности представления таксономических признаков.

Все признаки можно разделить:

- на группу A , для которых на множестве U можно задать линейный порядок;

- на группу B , для которых на множестве U линейный порядок задать нельзя.

В группу A войдут:

- количественные признаки, например, признак "Число щетинок на первой паре торакокод" со значениями 1, 2, 3, 4, или признак "Длина тела";
- признаки, в которых неявно используется количественная шкала, например, признак "Длина антенн I" со значениями "Длинные, больше длины раковины", "Почти равные длине раковины", "Короткие, меньше длины раковины" или признак "Наличие выростов на голове" со значениями "Без выростов", "С незначительными выростами", "С выростами".

Последний пример иллюстрирует необходимость учета слов-модификаторов.

В русском языке модификаторы представлены в основном количественными наречиями, которые можно разбить на три группы [8].

В первую группу входят те, которые "размывают" нечеткие значения лингвистических переменных, например, *более-менее горячий, около пяти, где-то 12* и т. п.

Назовем эту группу группой типа "Равно" и включим в нее следующие модификаторы: *приблизительно, около, где-то, почти, примерно, вероятно, кажется, более или менее, скорее всего, по-моему*.

Во вторую группу типа "Больше — Меньше" войдут модификаторы, которые изменяют значение переменной в одну из сторон: *немного, чуть, значительно, очень, слегка, несколько, много, не более, не менее, сильно, слишком, едва*. Например: *немного горячий, почти холодный, очень горячий* и т. п.

В третью группу войдут те, которые акцентируют, уменьшают неопределенность нечетких значений лингвистической переменной, например, *конечно это...*. Варианты: *определенно, точно* и т. п.

Использование модификаторов приводит к необходимости расширения множества U в соответствии с принципом обобщения для нечетких множеств [7, 9].

Приведенные примеры предполагают линейный порядок элементов на множестве определения U нечетких значений лингвистических переменных заданием некоторой количественной меры. Примером может служить температура, радиус кривизны, скорость, количество и т. п.

Рассмотрим вторую группу признаков (B), где линейный порядок на множестве U либо не существует, либо его задать нельзя.

Причиной несущественности линейного порядка может быть наличие всего двух взаимно ортогональных значений лингвистической переменной: *да/нет, истина/ложь, он/не он* и т. п. Например, признак "Зубцы на кромке раковины моллюска" со значениями: "Есть", "Отсутствуют".

Линейный порядок на множестве U нельзя задать, когда одно или несколько значений выбира-

ется из конечного неупорядоченного множества. Например, значения признака "Форма раковины" — это неупорядоченное множество: "Треугольная", "Треугольно-эллиптическая", "Эллиптическая", "Круглая". То же самое справедливо для признака "Цвет пигментных пятен": "Желтый", "Бурый", "Сиреневый". В таких случаях формирование функций принадлежности может осуществляться, например, с помощью метода попарных сравнений [9].

Результатом описания признаков лингвистическими переменными является таблица, структура которой имеет классический вид: столбцы — признаки, строки — объекты, в ячейках — нечеткие значения признаков для соответствующих объектов.

Использование описаний в процессе логического вывода

Пусть для каждого объекта базы знаний имеется некоторая результирующая оценка — степень соответствия описания объекта в пространстве признаков данным, полученным от пользователя. В начале работы все результирующие оценки равны, например, единице. Другими словами, все объекты из базы знаний системы претендуют быть результатом идентификации.

Тогда последовательность действий экспертной системы при диагностировании объекта можно описать следующим алгоритмом.

1. Система выбирает некоторый признак, представленный лингвистической переменной, и демонстрирует пользователю его внешнее описание.

2. Пользователь выбирает одно или несколько значений признака; при этом он может использовать слова-модификаторы.

3. Система формирует нечеткое значение лингвистической переменной, соответствующей признаку.

4. Для каждого объекта, представленного в базе данных, система вычисляет оценку степени соответствия построенной лингвистической переменной значениям признаков объектов.

5. Полученная оценка учитывается в общей результирующей оценке.

6. Система повторяет п. 1 или завершает процесс по достижению некоторого критерия для результирующей оценки.

Пример. Рассмотрим работу алгоритма на примере.

1. Пусть в базе данных представлена информация о четырех объектах, и вначале результирующая оценка у всех объектов равна 1:

$$Res_1 = 1; Res_2 = 1; Res_3 = 1; Res_4 = 1.$$

2. Система спрашивает пользователя о "Форме раковины моллюска".

Предположим, формальное представление этого признака в виде лингвистической переменной следующее:

$$L_1 = ("Форма раковины", \{ "треугольная", "треугольно-эллиптическая", "эллиптическая", "круглая" \}, \{ "Объект 1", "Объект 2", "Объект 3", "Объект 4" \}),$$

$$M = \{ \mu("треугольная"), \mu("треугольно-эллиптическая"), \mu("эллиптическая"), \mu("круглая") \}.$$

Пользователь отвечает, что у его объекта форма раковины — треугольная, но он не совсем в этом уверен.

Рассмотрим действия системы.

Пусть функции принадлежности множества M признака L_1 имеют следующий вид:

- $\mu("треугольная") = \{0,9, 0,3, 0,5, 0,4\};$
- $\mu("треугольно-эллиптическая") = \{0,5, 0,8, 0,5, 0,7\};$
- $\mu("эллиптическая") = \{0,1, 0,5, 0,8, 0,4\};$
- $\mu("круглая") = \{0,2, 0,3, 0,5, 0,9\}.$

Так как выбрано значение признака "треугольная", то результирующую оценку для следующего шага определения Res_i^{j+1} можно вычислить по формуле [9]

$$Res_i^{j+1} = \min(Res_i^j, \mu_i(z)), \quad (1)$$

где z — значение признака.

Так как пользователь не совсем уверен в точности определения, необходима поправка

$$\mu_i^*(z) = \begin{cases} \mu_i(z)k, & \text{если } \mu_i(z)k \leq 1; \\ 1, & \text{если } \mu_i(z)k > 1. \end{cases}$$

Возможные значения коэффициента k приведены в таблице.

Значение μ_i признака	Сильно сомневаюсь	Не совсем уверен	Согласен	Почти уверен	Абсолютно уверен
Больше или равно 0,7	k^2	k	1	$1/k$	$1/k^2$
Меньше 0,7	$1/k^2$	$1/k$	1	k	k^2

Значение k выбирается из интервала $[0,90...0,95]$.

Выполняем коррекцию результирующей оценки. При $k = 0,95$ получим $Res_1 = 0,86$; $Res_2 = 0,32$; $Res_3 = 0,53$; $Res_4 = 0,42$.

3. Пользователю предъявляется признак "Зубцы на кромке раковины".

Формальное представление этого признака в виде лингвистической переменной, например, следующее:

$$L_2 = ("Зубцы на кромке раковины", \{ "есть", "нет" \}, \{ 1, 2, 3 \}),$$

$$M = \{ \mu("есть"), \mu("нет") \}.$$

Предположим, что функции принадлежности множества **M** признака L_2 имеют следующий вид:

$$\begin{aligned}\mu(\text{"есть"}) &= \{1,0, 0,505, 0,01\}; \\ \mu(\text{"нет"}) &= \{0,01, 0,505, 1,0\}.\end{aligned}$$

Пользователь отвечает, что у его объекта зубцы на кромке раковины, кажется, есть.

Так как использовано слово — модификатор "*кажется*", применим операцию размывания [8] к функции $\mu(\text{"есть"})$. Смысл этой операции — сгладить отличия в значениях функции принадлежности. Получим $\mu_{ans}(\text{"есть"}) = \{1,0, 0,93, 0,36\}$.

Пусть в таблице признаков признак L_2 на множестве объектов имеет следующие значения:

$$\begin{aligned}L_2(\text{Объект1}) &= \text{"есть"}; \\ L_2(\text{Объект2}) &= \text{"есть"}; \\ L_2(\text{Объект3}) &= \text{"нет"}; \\ L_2(\text{Объект4}) &= \text{"нет"}.\end{aligned}$$

Признак L_2 задается нечеткими переменными "есть", "нет". Для вычисления результирующей оценки Res_i^{j+1} найдем пересечение значений функций принадлежности этих переменных со значениями функции $\mu_{ans}(\text{"есть"})$. Поэлементное пересечение вычисляется нахождением минимальных значений из соответствующих пар по всем элементам множества **U**.

Тогда

$$\begin{aligned}\mathbf{d}_1 &= \mu_{ans}(\text{"есть"}) \cap \mu(\text{"есть"}) = \{1,0, 0,505, 0,01\}; \\ \mathbf{d}_2 &= \mu_{ans}(\text{"есть"}) \cap \mu(\text{"нет"}) = \{0,01, 0,505, 0,36\}.\end{aligned}$$

Для пересчета результирующей оценки выбираются максимальные значения векторов $\mathbf{d}_1$ и $\mathbf{d}_2$. Это 1,0 и 0,505 соответственно. Тогда в соответствии с формулой (1)

$$Res_i^{j+1} = \min(Res_i^j, d_{\max}) \quad (2)$$

и $Res_1 = 0,86$; $Res_2 = 0,32$; $Res_3 = 0,505$; $Res_4 = 0,42$.

4. Система спрашивает пользователя о "Длине антенн I".

Формальное представление этого признака в виде лингвистической переменной, например, следующее:

$$\begin{aligned}L_3 &= (\text{"Длина антенн I"}, \\ \{x_1 &= \text{"длинные, больше длины раковины"}, \\ x_2 &= \text{"почти равные длине раковины"}, \\ x_3 &= \text{"короткие, меньше длины раковины"}\}, \\ \mathbf{U} &= \{1, 2, 3, 4, 5, 6, 7, 8, 9\}, \\ \mathbf{M} &= \{\mu(x_1), \mu(x_2), \mu(x_3)\}.\end{aligned}$$

Функции принадлежности признака L_3 имеют следующий вид:

$$\begin{aligned}\mu(x_1) &= \\ &= \{0,01, 0,505, 1,00, 0,750, 0,50, 0,255, 0,01, 0,010, 0,01\}; \\ \mu(x_2) &= \\ &= \{0,01, 0,255, 0,50, 0,750, 1,00, 0,750, 0,50, 0,255, 0,01\}; \\ \mu(x_3) &= \\ &= \{0,01, 0,010, 0,01, 0,255, 0,50, 0,750, 1,00, 0,505, 0,01\}.\end{aligned}$$

Пользователь отвечает, что у объекта длина антенн I чуть больше длины раковины.

Этот ответ можно интерпретировать как среднее между значениями "Антенны I длинные, больше длины раковины" и "Антенны I почти равны длине раковины". Применяя операцию размывания к $\mu(x_1)$ и $\mu(x_2)$ и усредняя значения, получим:

$$\mu_{ans} = \{0,01, 0,2, 0,71, 1,0, 0,71, 0,3, 0,04, 0,005, 0,005\}.$$

Пусть в таблице признаков признак L_3 на множестве объектов имеет следующие значения:

$$\begin{aligned}L_3(\text{Объект1}) &= x_1; \\ L_3(\text{Объект2}) &= x_1 \text{ или } x_2; \\ L_3(\text{Объект3}) &= x_2 \text{ или } x_3; \\ L_3(\text{Объект4}) &= x_3.\end{aligned}$$

Для вычисления значения функции принадлежности L_3 (Объект2) воспользуемся операцией граничного объединения двух нечетких множеств $\mathbf{A} \oplus \mathbf{B}$ [9]. Результат определяется как нечеткое множество **D**, заданное на том же универсуме **U**, функция соответствия которого определяется по формуле

$$\mu_D(x) = \min\{\mu_A(x) + \mu_B(x), 1\}.$$

Тогда $\mu(x_1) \oplus (x_2) = \{0,02, 0,76, 1,0, 1,0, 1,0, 1,0, 0,51, 0,26, 0,02\}$.

Находим пересечения μ_{ans} с $\mu(x_1)$, $\mu(x_1) \oplus (x_2)$, $\mu(x_2)$, $\mu(x_3)$.

$$\begin{aligned}\mathbf{d}_1 &= \mu_{ans} \cap \mu(x_1) = \\ &= \{0,01, 0,20, 0,71, 0,75, 0,50, 0,26, 0,01, 0,01, 0,01\}; \\ \mathbf{d}_2 &= \mu_{ans} \cap (\mu(x_1) \oplus (x_2)) = \\ &= \{0,01, 0,20, 0,71, 1,0, 0,71, 0,3, 0,04, 0,005, 0,005\}; \\ \mathbf{d}_3 &= \mu_{ans} \cap \mu(x_2) = \\ &= \{0,01, 0,2, 0,50, 0,750, 0,71, 0,3, 0,04, 0,005, 0,005\}; \\ \mathbf{d}_4 &= \mu_{ans} \cap \mu(x_3) = \\ &= \{0,01, 0,01, 0,01, 0,255, 0,5, 0,3, 0,04, 0,005, 0,005\}.\end{aligned}$$

Максимальные значения составят:

$$\begin{aligned}\max(\mathbf{d}_1) &= 0,75; \\ \max(\mathbf{d}_2) &= 1,0; \\ \max(\mathbf{d}_3) &= 0,75; \\ \max(\mathbf{d}_4) &= 0,5.\end{aligned}$$

Выполняем коррекцию результирующих оценок по формуле (1)

$$Res_1 = 0,75; Res_2 = 0,32; Res_3 = 0,505; Res_4 = 0,42.$$

5. Пользователю предъявляется признак "Длина раковины"

$L_4 = (" \text{Длина раковины(ДР)}", \{x_1 = " \text{ДР у Объекта 1}", x_2 = " \text{ДР у Объекта 2}", x_3 = " \text{ДР у Объекта 3}", x_4 = " \text{ДР у Объекта 4}"\},$

$$U = [5,25], M = \{ \mu(x_1), \mu(x_2), \mu(x_3), \mu(x_4) \}.$$

Здесь функции принадлежности $\mu(x_i)$ имеют кусочно-линейную трапециевидную форму и задаются четверками $\langle a, b, c, d \rangle$ — абсциссами точек излома:

$$\mu(x_1) = \{8, 10, 12, 14\};$$

$$\mu(x_2) = \{12, 14, 16, 18\};$$

$$\mu(x_3) = \{12, 14, 15, 17\};$$

$$\mu(x_4) = \{10, 12, 15, 17\}.$$

Пользователь отвечает, что "Длина раковины равна примерно 11 мм".

Так как функции принадлежности представляют собой нечеткие интервалы, то ответ пользователя следует представить нечетким интервалом с треугольной функцией принадлежности (см. рисунок).

Ширина основания треугольника $a_{\max} - a_{\min}$ определяется степенью уверенности пользователя в ответе и зависит от его оценок: *уверен, не уверен, примерно, около, абсолютно уверен* и т. п.

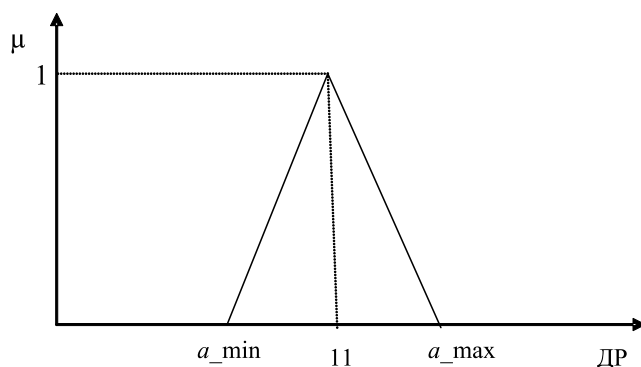
Значения $a_{\max}$ и $a_{\min}$ можно вычислить по формулам

$$a_{\max} = ans * (1 + w);$$

$$a_{\min} = ans * (1 - w),$$

где ans — числовое значение ответа пользователя (в примере оно равно 11), w — коэффициент не-уверенности/уверенности пользователя.

Вычисление поправочного коэффициента d_i для каждого объекта сводится к нахождению пересечения двух нечетких интервалов: трапециевидного из базы данных признаков и треугольного, построенного по ответу пользователя.



Ответ пользователя задан нечетким интервалом

Для данного примера при $w = 0,3$:

$$d_1 = 1; d_2 = 0,43; d_3 = 0,43; d_4 = 0,81.$$

Тогда оценки с учетом ответа на вопрос о значении признака L_4 составят:

$$Res_1 = 0,75; Res_2 = 0,32; Res_3 = 0,43; Res_4 = 0,42.$$

Заключение

Предложенный подход представления знаний и организации нечеткого вывода в адаптивных таксономических экспертных системах реализован в серии компьютерных определителей. Авторами разработан прототип адаптивной экспертной системы "Таксакейс" (www.taxakeys.org). На ее основе совместно с группой экспертов созданы интерактивные определители представителей кладоцер (Cladocera), копепода (Copepoda), отрядов кормового зоопланктона и рыб семейства собачковых (Blennidae) и сельдевых (Clupeidae) Севастопольского региона.

Исследование выполнено при финансовой поддержке РФФИ и Правительства города Севастополя в рамках научного проекта № 14-44-01566 "p_yug_a".

Авторы выражают благодарность В. В. Мельникову, С. А. Царину, А. В. Темных и М. И. Силакову за предоставленные материалы по конкретным группам объектов и помощь в наполнении базы знаний.

Список литературы

1. Лобанов А. Л., Кирейчук А. Г., Степаньянц С. Д., Смирнов И. С. Диагностические ключи: от текстовых дихотомических до компьютерных // Труды Зоологического института РАН. 2013. Приложение № 2. С. 249—268.
2. Adeeyo A. O., Ogunkunle A., Bello O. S. Review of Identification Aids in Biology: Past, Present and Future Challenges // Academic Journal of Interdisciplinary Studies MCSER Publishing, Rome-Italy. 2013. Vol. 2, N. 13. P. 27—34.
3. Dallwitz M. J. A Comparison of Interactive Identification Programs. URL: <http://delta-intkey.com/www/comparison/htm>
4. Lelekov S., Lyakh A. Taxex: Taxonomical Expert System — history of development and technology of identification / V. Berghe (Eds.) // Proceedings of "Ocean Biodiversity Informatics": an int. conf. on marine biodiversity data management. Hamburg, Germany, 29 Nov. — 1 Dec, 2004. Paris, UNESCO/IOC, VLIZ, BSH. 2007. P. 111—120.
5. Лях А. М., Лелеков С. Г. Адаптивная экспертная система для таксономической идентификации организмов // Экологическая безопасность прибрежной и шельфовой зон и комплексное использование ресурсов шельфа. Сб. научн. тр., Севастополь, 2014. Вып. 29. С. 107—113.
6. Лелеков С. Г., Лях А. М. Нечеткое описание признаков объектов в адаптивных таксономических экспертных системах // Вісник СевНТУ: зб. наук. пр. Серія: Автоматизація процесів та управління. Севастополь, 2014. Вип. 147/2014. С. 143—149.
7. Заде Л. А. Понятие лингвистической переменной и его применение к принятию приближенных решений. М.: Мир, 1976. 168 с.
8. Лелеков С. Г. Метод учета влияния модификаторов на нечеткие значения лингвистических переменных // Искусственный интеллект. 2013. № 4 (62). С. 454—462.
9. Леоненков А. В. Нечеткое моделирование в среде MATLAB и fuzzyTECH. СПб.: БХВ, 2003. 736 с.

S. G. Lelekov, Associate Professor, e-mail: p1859@yandex.ru,
Sevastopol State University, Sevastopol,
A. M. Lyakh, Senior Scientist, e-mail: antonlyakh@yahoo.com,
Institute of biology of the Southern seas, Sevastopol

The Adaptive Fuzzy Inference System for Biological Taxonomy

Taxonomic identification of natural objects requires qualified researchers (experts) or qualitative identifiers which present expert knowledge. Realization of taxonomic identifiers in the form of computer expert systems leads to the needs to transform taxonomic diagnosis to formalized computer description. Diagnosis contain specific terminology and fuzzed statements that hamper their understanding and obstruct mathematical formalization. For that reason taxonomic expert systems not widely used.

Adaptive expert systems partially solved that problem because they takes into account semantic ambiguity of natural language. Adaptability means that system adapts to interpretation of indistinct user and expert concepts. The theory of fuzzy sets is the mathematical basis of adaptive expert systems functioning. According to algorithm presented in the paper the expert system transforms expert description of organism features into linguistic variables, presents them to the user for organism identification, transforms user answers into another linguistic variables, and compare user linguistic variable with expert ones to find a correspondence of organisms description stored in the system knowledge base with the user object.

The proposed approach of knowledge representation and organization of fuzzy inference in adaptive taxonomic expert systems is implemented in a series of computer identifiers. The authors have developed a prototype of an adaptive expert system "Taxakeys" (taxakeys.org). On its basis, together with the group of experts, the interactive identifiers of Cladocera, Copepoda, zooplankton and two fish families Blennidae and Clupeidae of the Sevastopol region have been created.

Keywords: expert systems, linguistic variables, fuzzy inference, biological taxonomy, computer taxonomic keys

References

1. Lobanov A. L., Kirejchuk A. G., Stepan'janc S. D., Smirnov I. S. Diagnosticheskie kljuchi: ot tekstovyh dihotomicheskikh do komp'yuternyh, *Trudy Zoologicheskogo instituta RAN*, 2013, Prilozhenie, no. 2, pp. 249—268.
2. Adeeyo A. O., Ogunkunle A., Bello O. S. Review of Identification Aids in Biology: Past, Present and Future Challenges. *Academic Journal of Interdisciplinary Studies MCSER Publishing, Rome-Italy*, 2013, vol. 2, no. 13, pp. 27—34.
3. Dallwitz M. J. *A Comparison of Interactive Identification Programs*. URL: <http://delta-intkey.com/www/comparison/htm>
4. Lelekov S., Lyakh A. Taxex: Taxonomical expert system — history of development and technology of identification. V. Berghe (eds.). *Proceedings of "Ocean Biodiversity Informatics": an Int. conf. on marine biodiversity data management*. Hamburg, Germany, 29 Nov. — 1 Dec, 2004. Paris: UNESCO/IOC, VLIZ, BSH, 2007, pp. 111—120.
5. Lyakh A. M., Lelekov S. G. Adaptivnaja ekspertnaja sistema dlja taksonomicheskoy identifikacii organizmov. *Ekologicheskaja bezopasnost pribrezhnoj i shelfovoj zon i kompleksnoe ispolzovanie resursov shelfa*. Sb. nauchn. tr. Sevastopol, 2014, iss. 29, pp. 107—113.
6. Lelekov S. G., Lyakh A. M. Nechetkoe opisanie priznakov ob'ektov v adaptivnyh taksonomicheskikh jekspertnyh sistemah. *Visnik SevNTU: zb. nauk. pr. Serija: Avtomatizacija procesiv ta upravlinja*. Sevastopol, 2014, iss. 147/2014, pp. 143—149.
7. Zade L. A. *Ponjatije lingvisticheskoy peremennoj i ego primenenie k prinjatiju priblizhennyh reshenij*. Moscow: Mir, 1976. 168 p.
8. Lelekov S. G. Metod ucheta vlijanija modifikatorov na nechetkie znachenija lingvisticheskikh peremennyh, *Artificial intelligence*, 2013, no. 4 (62), pp. 454—462.
9. Leonenkov A. V. Nechetkoe modelirovanie v srede MATLAB i fuzzy TECH. SPb.: BHV, 2003, 736 p.

УДК 004.272.34

В. С. Дыкин, магистрант, **В. Ю. Мусатов**, канд. техн. наук, доцент, **А. С. Варезников**, аспирант,
А. А. Большаков, д-р техн. наук, профессор, **В. В. Сысоев**, д-р техн. наук, профессор
СГТУ, г. Саратов, e-mail: robsii@sstu.ru

Применение генетического алгоритма для определения архитектуры нейронной сети для прибора вида "электронный нос"

Рассматривается возможность применения генетических алгоритмов для выбора топологии искусственных нейронных сетей в системе распознавания газоаналитического устройства вида "электронный нос". Поиск топологии осуществлен оптимизацией по критериям увеличения процента правильного распознавания и уменьшения вычислительных затрат с использованием массива заранее заданных вероятных значений параметров архитектуры нейронной сети. Оптимизируемые характеристики: число скрытых слоев, количество нейронов в каждом слое и объем обучающей выборки. Распознавание проведено в системе Matlab нейронной сетью прямого распространения с функцией обучения Левенберга-Марквардта. Полученные результаты подтверждают работоспособность созданного алгоритма поиска топологии искусственной нейронной сети.

Ключевые слова: "электронный нос", генетические алгоритмы, мультисенсорная микросистема, анализ газов, распознавание образов

Введение

При построении газоаналитических приборов существуют два принципиально разных подхода. Первый заключается в применении аналитических инструментов химического анализа (спектрометры, хроматографы), при котором газовая смесь идентифицируется как набор составляющих ее компонентов, измеряемых с большой точностью. Во втором подходе смесь идентифицируется как целое, следуя аналогии с работой биологического механизма обоняния [1]. Поэтому приборы, работающие согласно этому принципу, получили название "электронный нос". Они основаны на наборе слабоселективных сенсоров, совокупный сигнал которых обрабатывается методами распознавания образов в режиме реального времени для классификации газовых смесей [2]. Селективность отдельных сенсоров к определенным газам не играет определяющей роли, но из "перекрестной" чувствительности можно извлечь информацию о виде газовой смеси и, при необходимости, о концентрации отдельных ее составляющих [3].

Среди множества методов распознавания образов, применяемых при создании приборов вида "электронный нос", наибольший интерес вызывают искусственные нейронные сети (ИНС), которые позволяют проводить различную предобработку сенсорных сигналов и классифицировать их в соответствующие группы [4]. Для успешной работы ИНС необходимо обеспечить достаточную обучающую выборку и выбрать архитектуру сети, наиболее подходящую для решения задачи.

Следует отметить, что в большинстве случаев задача выбора ИНС при использовании в приборах "электронный нос" решается, как правило, эмпирическим способом [5]. В то же время с 70-х гг. прошлого века известны генетические алгоритмы (ГА) [6], представляющие метод поиска глобального экстремума целевой функции. Эти методы отражают естественную эволюцию решения задач, которая включает естественный отбор и наследование. В настоящей работе исследованы возможности применения ГА к выбору архитектуры ИНС, решающей задачу распознавания газов обработкой сенсорных сигналов прибора "электронный нос" с наименьшей погрешностью.

1. Описание объекта исследования

1.1. Прибор вида "электронный нос" на основе мультисенсорного чипа. Принципиальное устройство прибора "электронный нос" представлено на рис. 1 (см. вторую сторону обложки) [7]. Прибор состоит из трех модулей: набора газовых сенсоров, системы предобработки данных и системы обработки информации (ИНС или другой математический метод распознавания или классификации [8]).

Газовые сенсоры обеспечивают физический уровень взаимодействия с газом и преобразуют химические реакции и другие взаимодействия сенсорного материала с газовой фазой в измеряемый электрический (в некоторых случаях, оптический) сигнал [9]. При составлении наборов сенсоров каждой газовой смеси соответствует векторный сигнал [10], который может быть обработан методами распознавания образов. В зависимости от выбранного метода распознавания сенсорные сигналы могут быть подвергнуты предварительной обработке для лучшего выделения признаков, характеризующих тестовый газ (рис. 1, см. вторую сторону обложки).

Перспективным направлением при конструировании приборов "электронный нос" является применение мультисенсорных систем, сформированных из однотипных датчиков, расположенных на одном кристалле [11]. В этом случае датчики мультисенсорной системы имеют единый тип сигнала, а вариация свойств и выходных характеристик достигается на основе изменения внутренних параметров [12]. Наибольшее распространение в таких системах получили полупроводниковые датчики хеморезистивного типа [8]. Принцип их действия основан на изменении поверхностного сопротивления хеморезистора при адсорбции молекул газа на поверхности полупроводника. Такой датчик формируется, как правило, на диэлектрической подложке нанесением электродов, между которыми помещается газочувствительный полупроводниковый элемент (как правило, оксид металла).

В данной работе применяли мультисенсорный чип, сформированный на основе пленки SnO_2 [11]. Расположение электродов, как показано на рис. 2 (см. вторую сторону обложки), разделяет сплошной слой SnO_2 на 38 сегментов на площади 4×8 мм. Чип размещен в 120-штырьковый корпус (PGA-120, Куосега Со., Япония) на керамических столбиках.

Для создания неоднородности свойств хеморезистивных элементов в чипе применен переменный нагрев подложки, для чего только одна сторона корпуса закреплена на трубках керамическим клеем. Электрические соединения электродов, терморезисторов и нагревателей выполнены с использованием золотых проволочек толщиной 38 мкм, приваренных ультразвуковой сваркой.

В работе исследована возможность использования ГА для выбора архитектуры ИНС, которая осуществляет распознавание векторных сигналов, генерированных рассмотренной выше линейкой хеморезистивных элементов чипа, к воздействию различных газовых смесей с наименьшей погрешностью.

1.2. Принцип работы ГА и взаимодействие с ИНС.

В описании ГА [6] используются термины, заимствованные из генетики и техники. Их основой служит модель естественной эволюции и методы

Таблица 1

Соответствие обозначений в ГА и ИНС [6]

ГА	ИНС
Ген	Бит в строке решения
Хромосома	Упорядоченная последовательность генов
Особь	Решение, представленное хромосомой
Популяция	Множество особей в текущем поколении
Генотип	Набор возможных хромосом особи
Фенотип	Декодированное из хромосомы значение иско- мых параметров
Поколение	Популяция особей на данной итерации алго- ритма
Потомки	Поколение, созданное на основе исходного по- коления на данной итерации

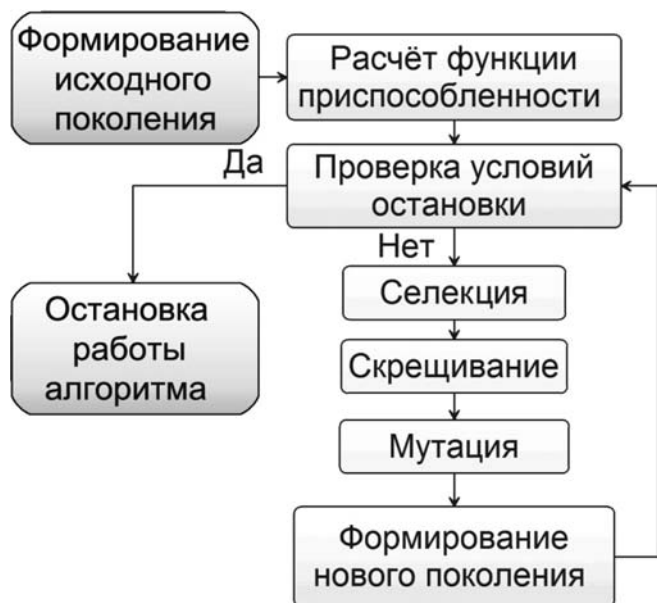


Рис. 3. Блок-схема функционирования ГА

случайного поиска. Искомое решение представляется в виде двоичной строки, в которой в двоичном коде закодированы значения оптимизируемых параметров. Соответствие обозначений в ГА и ИНС приведены в табл. 1.

Классический ГА использует двоичное кодирование параметров и состоит из последовательности операций, представленных на рис. 3.

Существуют различные способы взаимодействия ГА и ИНС [7], например:

- ИНС применяют для поддержки ГА: сеть формирует исходную популяцию, а ГА проводит поиск по кратчайшему пути;
- ГА применяют для поддержки ИНС: 1) ГА подбирает параметры, либо преобразует пространство параметров, используемых ИНС для классификации; 2) ГА подбирает правила обучения, либо параметры, влияющие на процесс обучения ИНС; 3) ГА обучает ИНС через оптимизацию весов сети.

Выбор способа взаимодействия ГА и ИНС обоснован тем, что точных алгоритмов выбора архитектуры ИНС не существует [5]. Определение оптимального числа нейронов возможно по теореме Арнольда — Колмогорова — Хехт-Нильсена [14], однако она имеет рекомендательный характер. В практических задачах полученное таким образом число нейронов может не соответствовать оптимальной топологии.

В настоящей работе рассмотрена ИНС прямого распространения, которая является широко используемым средством распознавания образов [4]. Для таких ИНС имеется большой выбор алгоритмов обучения, оптимизированных с точки зрения потребления ресурсов. Для примера рассмотрена многослойная ИНС с сигмоидальной функцией активации, в которой поиск архитектуры сводится к выбору топологии и связанному с ней объему оптимальной обучающей выборки.

Для ИНС с различным числом синаптических связей размер оптимальной обучающей выборки варьируется. При этом объем выборки меньше оптимального приводит к некорректному обучению, а превышение оптимального объема — к снижению сходимости. Сходимость — способность распознавать образы, не участвовавшие в обучении. Критерий оптимальности заключается в минимальной погрешности распознавания при решении поставленной задачи ИНС.

2. Описание результатов экспериментов и их анализ

2.1. Экспериментальное получение векторных сигналов прибора "электронный нос" при воздействии газов. Экспериментальные результаты по воздействию газовых смесей на мультисенсорный чип получены с использованием установки, изображенной на рис. 4.

Векторный сигнал мультисенсорного чипа считывался электронно-измерительным блоком KAMINA

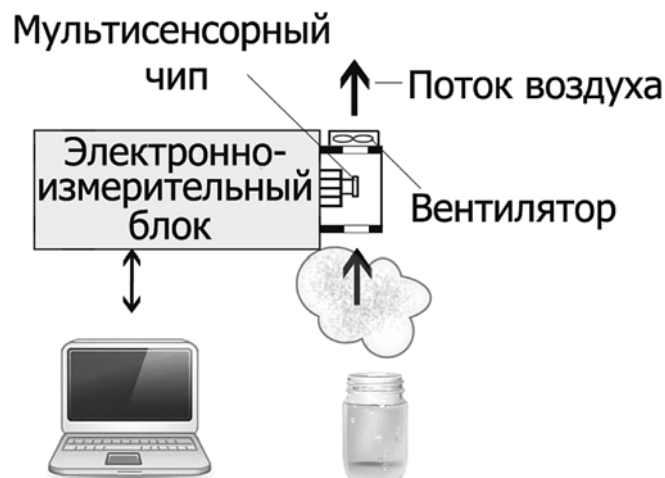


Рис. 4. Схема экспериментальной установки по получению сигналов прибора "электронный нос" при воздействии различных паров

(KIT, Германия) [11], который подключен через интерфейс RS232 к ПК. Измерение сопротивлений сенсорных сегментов проводили между парой электродов со скоростью опроса 30 мс на сегмент. Среднюю температуру поверхности чипа поддерживали на уровне 350 °С. Полученные данные сохраняли на компьютере. Объем измерительной камеры, в которой установлен образец, составлял около 50 см³. Напуск тестового газа (в смеси с чистым лабораторным воздухом) осуществлялся в проточном режиме; между тестовыми газами проводили напуск чистого лабораторного воздуха. Время экспозиции образца каждого газа составляло 1 мин.

В качестве тестовых газов использованы пары этанола, ацетона, пропанола и аммиака. Открытый резервуар с соответствующими растворами помещали в поток воздуха, нагнетаемого встроенным вентилятором в измерительную камеру. Данная установка полностью симулировала условия реального применения устройства.

2.2. Реализация ГА. В настоящей работе применен ГА на основе двоичного кодирования хромосомы с размером популяции, равной 16. Используемый метод отбора — способ рулетки (*chart roulette selection*) [13]. Вероятность мутации принята равной 10. Процедура скрещивания заключается в нахождении точки скрещивания случайным образом и обменом частями хромосом до и после выбранной точки для двух родительских особей. Оптимизация, проводимая ГА, включает оценку значений ошибки обучения ИНС и минимизацию числа нейронов, составляющих скрытые слои ИНС. Результативность ИНС повышается при снижении ошибки обучения. Минимизация числа нейронов осуществляется дополнительным слагаемым целевой функции, значение которого пропорционально числу нейронов. В настоящей работе целевая функция определена на основе обобщения и математического моделирования экспериментальных данных в виде

$$y = e + N_1k_1 + N_2k_2, \tag{1}$$

где e — евклидова норма ошибки, которая определяется как разность ожидаемого и действительного результатов выхода сети; N_1 — число нейронов в первом скрытом слое; N_2 — число нейронов во втором скрытом слое; k_1, k_2 — коэффициенты, которые подбирают так, чтобы максимальные значения N_1 и N_2 обеспечивали значение целевой функции, равное половине порога останова алгоритма, определяемого как компромиссное значение между временем поиска и результативностью. Эмпирически найденные коэффициенты равны $k_1 = 0,0078$, $k_2 = 0,0156$. Изменение этих значений требуется лишь при смене способа обучения ИНС.

При построении ИНС, как правило, выбирается один или два скрытых слоя нейронов [5]. Критерием минимального числа скрытых слоев является



Рис. 6. Хромосома, использованная для поиска топологии ИНС

нелинейная разделимость пространства признаков. Этот случай описывает так называемая задача включающего "ИЛИ". Максимальное число скрытых слоев, в общем случае, не ограничено. Однако на практике увеличение числа скрытых слоев вызывает существенное повышение вычислительных затрат при процедуре обучения. В нашем случае максимальное число нейронов в слое выбирали исходя из экспериментальных результатов. Выбор осуществлен таким образом, что при дальнейшем увеличении числа нейронов результативность убывает, а вычислительная нагрузка чрезмерно возрастает (рис. 5, см. вторую сторону обложки).

Исходный объем данных — 581 выборка, из которых для обучения использовались 100, 150, 200, 250, 300, 350, 400 или 450 выборок. С учетом описанных выше критериев сформировано исходное множество вероятных значений архитектуры ИНС, из которых составлена хромосома (рис. 6).

2.3. Анализ полученных результатов и работы ГА. Работа ГА по подбору параметров ИНС для обработки мультисенсорного векторного сигнала проверена кроссвалидацией. При этом контрольной выборкой являлись экспозиции к тестовым газам, не участвующим в обучающей выборке, результат отображен в табл. 2.

Для определения точности распознавания использован расчет суммы разностей желаемого и действительного откликов по каждому нейрону выходного слоя с установленным порогом 0,2, где отклик со значением 1 одного из выходных нейронов соответствует 100 %-ной вероятности присут-

Таблица 2

Результат работы алгоритма			
Выборка, шт.	N_1 , нейронов	N_2 , нейронов	Распознавание, %
400	8	7	98,01
400	8	7	89,15
400	4	5	99,73
450	4	7	99,98
450	5	6	99,55

ствия соответствующего газа, а отклик, равный 0,8 (1 – 0,2), соответствует 0 %-ной вероятности присутствия согласно выражению:

$$\begin{cases} r_i = (|y_i - d| - 0,2) \cdot 5, & \text{при } |y_i - d| < 0,2; \\ r_i = 0, & \text{при } |y_i - d| \geq 0,2. \end{cases}$$

$$R = \left(\sum_{i=1}^5 r_i \right) \cdot 100 \%. \quad (2)$$

Обучение ИНС зависит от случайной величины инициализации весовых коэффициентов, поэтому рассмотрено минимальное число экспериментов, обеспечивающих статистически значимый результат. Минимальное число экспериментов установлено по критерию согласия Колмогорова [15]. Для

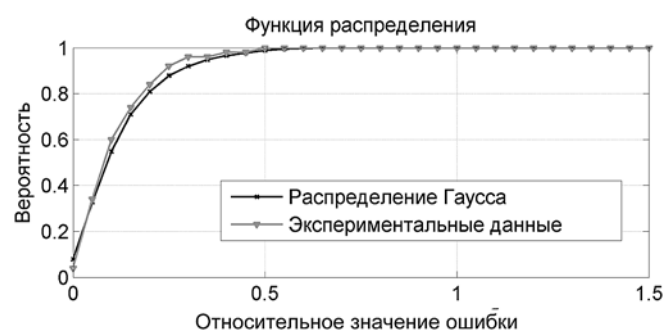


Рис. 7. Функция распределения ошибки при обучении ИНС. По оси абсцисс отложена разность желаемого и действительного откликов

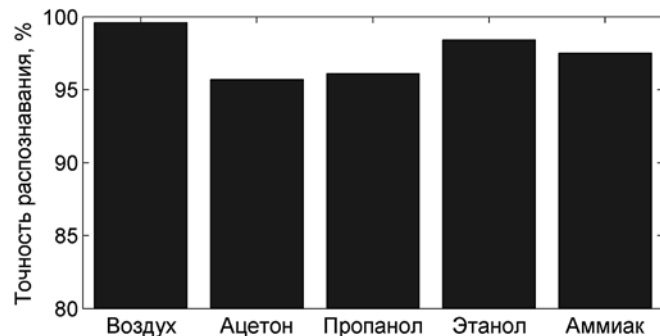


Рис. 8. Усредненные результаты контрольного тестирования

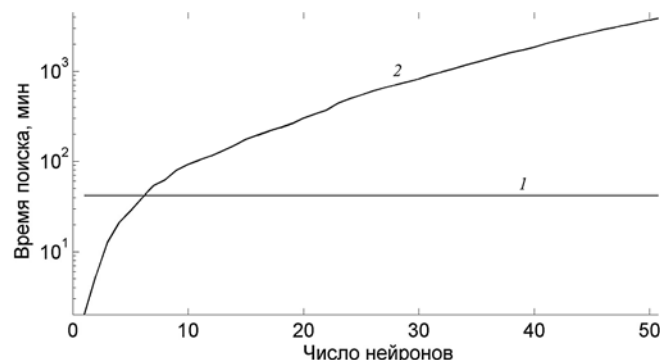


Рис. 9. Сравнение времени работы ГА и времени перебора для нахождения оптимальной конфигурации ИНС, состоящей из двух слоев. По оси абсцисс отложено число нейронов в первом слое ИНС

50 испытаний значение максимального отклонения $F_{\text{эмфир}}$ от $F_{\text{теор}}$ оказалось меньше расчетного (максимальное отклонение 0,064, гамма-распределение, 50 точек, уровень доверия 0,9) (рис. 7). Аналогичные результаты получены и при других выборах данных.

Важной характеристикой ИНС является способность к обобщению, т. е. генерирование правильных сигналов при подаче на ее вход данных, не участвовавших в процессе обучения. Поэтому способность обученной ИНС к обобщению проверяли контрольным тестированием. Проведена серия контрольных тестирований по анализу мультисенсорного векторного сигнала к каждой из тестовых газовых смесей. Для примера усредненные результаты по 50 тестированиям приведены на рис. 8.

Как видно из рис. 8, тестовые газы успешно распознаются ИНС, конфигурацию которой определил ГА.

В результате применения ГА удалось существенно сократить время поиска подходящей топологии ИНС по сравнению с методом простого перебора различных конфигураций ИНС с установленным ограничением по отношению вычислительных затрат к результативности. Например, на рис. 9 показано время поиска топологии ИНС методом перебора, выполненного на ПК (ЦП Intel Core i3, 2,4 ГГц, 2 Гбайт ОЗУ). Как видно из рис. 9, затраченное время (при проведении 50 экспериментов для каждого значения числа нейронов) составляет 4313 мин. При этом использование ГА на идентичном множестве исходных параметров позволило найти оптимальную ИНС за 42 мин, что значительно меньше времени, затраченного при использовании метода перебора.

Заключение

Алгоритм подбора конфигурации ИНС с использованием ГА показал работоспособность и возможность применения для обработки векторного сигнала мультисенсорного чипа в составе прибора вида "электронный нос" с высоким процентом правильного распознавания тестовых газовых смесей. Применение предложенного способа для определения значений параметров ИНС обеспечивает существенное преимущество по вычислительным затратам, времени вычисления характеристики ее конфигурации по сравнению, например, с поиском полным перебором.

Определение архитектуры ИНС с использованием ГА показало в рассмотренном приложении достаточно хороший результат распознавания и может быть рекомендовано к использованию при проектировании газоаналитических приборов вида "электронный нос".

Работа частично поддержана в рамках госзадания Минобрнауки РФ, договор № 8.236.2014/К.

Список литературы

1. Ганшин В. М., Фесенко А. В., Чебышев А. В. От обонятельных моделей к "электронному носу": новые возможности параллельной аналитики // Спец. техника. 1999. № 1—2. С. 5—19.
2. Сысоев В. В., Мусатов В. Ю. Газоаналитические приборы "электронный нос" Саратов: Изд-во Сарат. гос. техн. ун-та, 2011. 100 с.
3. Рембеза С. И. Нужен ли человечеству искусственный нос? // Природа. 2005. № 2. С. 5—12.
4. Осовский С. Нейронные сети для обработки информации / Пер. с польс. И. Д. Рудинского. М.: Финансы и статистика, 2004. 343 с.
5. Хайкин С. Нейронные сети: полный курс, 2-е изд.: пер. с англ. М.: Вильямс, 2006. 1104 с.
6. Гладков Л. А., Курейчик В. В., Курейчик В. М. Генетические алгоритмы: уч. пособие. М.: ФИЗМАТЛИТ, 2006. 320 с.
7. Althainz P., Goschnick J., Ehrmann S. et al. Multisensor microsystem for contaminants in air // Sensors and Actuators B. 1996. Vol. 33. P. 72—76.
8. Stetter J. R., Penrose W. R. Understanding chemical sensors and chemical sensor arrays (Electronic Noses): Past, Present, and Future // Sensors Update. 2002. Vol. 10. Issue 1. P. 189—229.
9. James D., Scott S. M., Ali Z., O'Hare W. T. Chemical Sensors for Electronic Nose Systems // Microchim. Acta. 2005. N. 149. P. 1—17.
10. Scott S. M., James D., Ali Z. Data analysis for electronic nose systems // Microchim. Acta. 2007. N. 156. P. 183—207.
11. Sysyov V., Kiselev I., Frietsch M., Goschnick J. Temperature Gradient Effect on Gas Discrimination Power of an Oxide Thin-film Sensor Microarray // Sensors. 2004. N. 4. P. 37—46.
12. Сысоев В. В., Кучеренко Н. И., Кисин В. В. Текстурированные пленки оксида олова для микросистем распознавания газов // Письма в ЖТФ. 2004. Т. 30, № 18. С. 14—20.
13. Рутковская Д., Пилиньский М., Рутковский Л. Нейронные сети, генетические алгоритмы и нечеткие системы / Пер. с польс.: И. Д. Рудинский. М.: Горячая линия-Телеком, 2006. 140 с.
14. Большаков А. А., Каримов Р. Н. Методы обработки многомерных данных и временных рядов. М.: Горячая линия. Телеком, 2007. 522 с.
15. Колемаев В. А., Староверов О. В., Турундаевский В. Б. Теория вероятностей и математическая статистика. М.: Высшая школа, 1991. 400 с.

V. S. Dykin, Indergraduate Student, V. Yu. Musatov, Associated Professor,
A. S. Varegnikov, Graduate Student, A. A. Bolshakov, Professor, V. V. Sysyov, Professor,
SSTU, Saratov, robsii@sstu.ru

Application of Genetic Algorithm to Define an Architecture of Artificial Neural Network for Processing of Multisensor Signal in Electronic Nose

The application of genetic algorithm to finding a topology of an artificial neural network as a part of a gas-analytical device of "electronic nose" kind is considered for odor pattern recognition. The search of the topology has been carried out via optimization to increase the recognition rate and to reduce the computing cost by genetic algorithms as a criterion for predefined possible parameters values. The optimized parameters are the number of hidden layers, the number of neurons in each layer and the training set size. The algorithm code and data processing have been realized in MATLAB software. The odor pattern recognition has been produced by feed-forward artificial neural network with Levenberg-Marquardt training method. The received results approve the efficiency of a proposed method to ANN topology finding in electronic nose devices.

Keywords: "electronic nose", genetic algorithm, gas sensor, multisensor microarray, gas analysis, pattern recognition

References

1. Ganshin V. M., Fesenko A. V., Chebishev A. V., Ot obonyatelnih modeley k elektronnomu nosu: novye vozmozhnosti paralelnoi analitiki, *Spec. tehnika*, 1999, no. 1—2, pp. 5—19.
2. Sysyov V. V., Musatov V. Yu., *Gazoanaliticheskie pribory vida "elektronniy nos"*, Saratov, Sarat. gos. nehn. un-t, 2011, 100 p.
3. Rembeza S. I. Nujen li chelovechestvu iskustvenny nos? *Priroda*, 2005, no. 2, pp. 5—12.
4. Osovsky S., *Neyronnie seti dlya obrabotki informacii*. Per. s polskogo Rudinsky I. D., Moscow, Finansy i statistika, 2004. 343 p.
5. Haykin S. *Neyronnie seti: polnyi kurs, 2 izd.: per s angl.*, Moscow, Vilyams, 2006, 1104 p.
6. Gladkov L. A., Kureychik V. V., Kureychik V. M. *Geneticheskie algoritmy, uch. posobie*, Moscow.: FIZMATLIT, 2006, 320 p.
7. Althainz P., Goschnick J., Ehrmann S. et al, Multisensor microsystem for contaminants in am, *Sensors and Actuators B*, 1996, vol. 33, pp. 72—76.
8. Stetter J. R., Penrose W. R., Understanding Chemical Sensors and Chemical Sensor Arrays (Electronic Noses): Past, Present, and Future, *Sensors Update*, 2002, vol. 10, issue 1, pp. 189—229.
9. James D., Scott S. M., Ali Z., O'Hare W. T. Chemical Sensors for Electronic Nose Systems, *Microchim. Acta*, 2005, no. 149, pp. 1—17.
10. Scott S. M., James D., Ali Z. Data analysis for electronic nose systems, *Microchim Acta*, 2007, no. 156, pp. 183—207.
11. Sysyov V., Kiselev I., Frietsch M., Goschnick J. Temperature Gradient Effect on Gas Discrimination Power of an Oxide Thin-film Sensor Microarray, *Sensors*, 2004, no. 4, pp. 37—46.
12. Sysyov V. V., Kucherenko N. I., Kisin V. V. Texturirovannye plenki oxide olova dlya mikrosistem raspoznavaniya gazov (Textured film tin oxide gas detection for microsystems), *Pisma v Jurnal tekhnicheskoi fiziki*, 2004, vol. 30, no. 18, pp. 14—20.
13. Rutkovskaya D., Pilinsky M., Rutkovsky L. *Neyronnie seti, geneticheskie algoritmy i nechetkie systemi*, per. s polskogo, Rudinsky I. D., Moscow, Goryachaya liniya-Telekom, 2006, 140 p.
14. Bolshakov A. A., Karimov R. N. *Metody obrabotki mnogomernykh dannykh i vremennykh ryadov* (Multidimensional data and time series processing methods), Moscow, Goryachaya liniya. Telecom, 2007, 522 p.
15. Kolemaev V. A., Staroverov O. V., Turunsaevsky V. B. *Teoriya veroyatnostey i matematicheskaya statistika*, Moscow, Vyshaya shkola, 1991, 400 p.

МОДЕЛИРОВАНИЕ И ОПТИМИЗАЦИЯ MODELING AND OPTIMIZATION

УДК 519.246

Б. Г. Кухаренко, канд. физ.-мат. наук, ст. науч. сотр., вед. науч. сотр.,
Институт машиноведения РАН, г. Москва, e-mail: kukharenko@imash.ru,

М. О. Солнцева-Чалей, аспирант,
Московский физико-технический институт (ГУ), e-mail: solnceva.chalei@gmail.com

Спектральный метод с использованием полярной кривизны для анализа результатов кластеризации многомерных траекторий

Геометрический метод используется для получения модели кластеризации в виде объединения аффинных подпространств. Подобие набора векторов оценивается многомерным тензором близости векторов, который развертывается в матрицу близости (подобия) векторов и анализируется спектральным методом. В качестве примера рассматривается кластеризация траекторий движения самолетов при посадке в аэропорту. Оценка близости в кластере по мере косинуса обнаруживает потенциально посторонние траектории полета самолетов в зоне риска.

Ключевые слова: анализ данных, многомерные траектории, модель объединения аффинных подпространств, полярная кривизна, спектральная кластеризация

Введение

При анализе наборов существенно многомерных векторов нельзя использовать методы визуализации, обеспечивающие оценку числа кластеров. Поэтому все алгоритмы кластеризации имеют ту же проблему, что и алгоритм K -средних. Примером является кластеризация в пространстве многомерных векторов $\mathbb{R}^D$ (где $D \gg 1$) на основе генеративных моделей, в которых алгоритм ожидания-максимизации правдоподобия — Expectation-Maximization (EM) используется для обучения смеси распределений. Во-первых, чтобы оценить это параметрическое распределение, необходимо сделать предположение о Гауссовом распределении в каждом определяемом кластере. Во-вторых, логарифм правдоподобия может иметь много локальных минимумов, и, следовательно, требуются многочисленные запуски, чтобы получить приемлемое решение. Позже алгоритм K -средних был обобщен, чтобы определять K наилучшим образом аппроксимирующих d -мерных аффинных подпространств для набора векторов в $\mathbb{R}^D$, т. е. прототипом для набора векторов становится аппроксимирующее аффинное подпространство, а не центроид, как в алгоритме K -средних.

Альтернативой является применение спектральных методов кластеризации, в которых используются главные собственные вектора матрицы близости (подобия), основанной на Евклидовом расстоянии между многомерными векторами. Спектральные методы успешно применяют к задачам

сегментирования изображений (размерность пространства характеристик для пикселей изображения ≤ 7), но их применимость ограничена парными мерами близости (подобия) при формировании матрицы близости (подобия). Однако геометрические задачи требуют анализа выборки более двух многомерных векторов, чтобы оценить их меру подобия. При решении таких задач определяется вероятность принадлежности к одному и тому же кластеру для набора векторов (а не пары), что приводит к многомерному тензору близости (подобия). Ниже показано, что спектральный метод кластеризации с использованием многомерного тензора близости (подобия) векторов обеспечивает точную кластеризацию для существенно многомерных данных, представляющих траектории движения самолетов при посадке в аэропорту (данные в открытом доступе на сайте <https://c3.nasa.gov/dashlink/resources/132/>).

1. Спектральная кластеризация с тензором близости на основе полярной кривизны

Задача гибридного линейного моделирования (*hybrid linear modeling*) предполагает, что набор данных (многомерных векторов) достаточно хорошо аппроксимируется объединением аффинных подпространств (*flats*), и необходимо одновременно оценить параметры каждого из аффинных пространств и ассоциацию этих многомерных векторов с аффинными подпространствами [1]. Аффинное d -мерное

подпространство является подмножеством пространства векторов $\mathbb{R}^D$ и характеризуется решением линейной системы уравнений $\mathbb{F} = \{\mathbf{z} | \mathbf{z} \in \mathbb{R}^D, \mathbf{F}^T \mathbf{z} = \boldsymbol{\gamma}\}$, где $\mathbf{F} \in \mathbb{R}^{D \times (D-d)}$, $\boldsymbol{\gamma} \in \mathbb{R}^{1 \times (D-d)}$ (например, 0-мерное аффинное подпространство (0-flat) — точка; 1-мерное (1-flat) — плоскость; $(D-1)$ -мерное $((D-1)\text{-flat})$ — гиперплоскость). В настоящей работе рассматривается специальный случай гибридного линейного моделирования, когда все аффинные подпространства имеют одинаковую размерность $d \geq 0$ ($d\text{-flats clustering}$) [2]. Используется определенный в работе [3] многомерный тензор аффинности (близости) (*affinity tensor*) для набора векторов и алгоритм спектральной кластеризации [4, 5]. Для каждого $(d+2)$ векторов из набора данных назначается аффинная мера и в результате формируется (многомерный) тензор аффинности (близости) порядка $(d+2)$. Развертывание этого тензора близости в матрицу близости (подобия) обеспечивает применение спектральной кластеризации [4, 5].

Пусть d и D — целые числа, такие, что $0 \leq d < D$. Для каждого $(d+2)$ различных векторов-столбцов $\mathbf{z}_1, \dots, \mathbf{z}_{d+2} \in \mathbb{R}^D$, $V_{d+1}(\mathbf{z}_1, \dots, \mathbf{z}_{d+2})$ обозначает объем выпуклой оболочки, образованной $(d+1)$ векторами — симплекса $(d+1)$ -го порядка $((d+1)\text{-мерного обобщения треугольника или } (d+1)\text{-мерного тетраэдра})$ [6]. В каждой вершине $\mathbf{z}_i$ полярный синус определяется как

$$\text{psin}_{\mathbf{z}_i}(\mathbf{z}_1, \dots, \mathbf{z}_{d+2}) = \frac{(d+1)! V_{d+1}(\mathbf{z}_1, \dots, \mathbf{z}_{d+2})}{\prod_{j=1, d+2, j \neq i} \|\mathbf{z}_j - \mathbf{z}_i\|}, \quad i = \overline{1, d+2}. \quad (1)$$

Пусть $\text{diam}(\mathbb{Z})$ обозначает диаметр набора векторов $\mathbb{Z} = \{\mathbf{z}_1, \dots, \mathbf{z}_{d+2}\}$. Полярная кривизна (*polar curvature*) набора из $(d+2)$ векторов определяется в работах [2, 7] как

$$c_p(\mathbf{z}_1, \dots, \mathbf{z}_{d+2}) = \text{diam}(\{\mathbf{z}_1, \dots, \mathbf{z}_{d+2}\}) \left(\sum_{i=1}^{d+2} (\text{psin}_{\mathbf{z}_i}(\mathbf{z}_1, \dots, \mathbf{z}_{d+2}))^2 \right)^{1/2}. \quad (2)$$

При $d = 0$ полярная кривизна совпадает с Евклидовым расстоянием. Введем обозначения для набора индексов $I = \{1, \dots, 2d\}$ и матрицы из векторов-столбцов $\mathbf{Z}_I = [\mathbf{z}_1 \dots \mathbf{z}_{2d}]$. С учетом (1) c_p (2) принимает вид

$$c_p(\mathbf{z}_1, \dots, \mathbf{z}_{d+2}) = \max_{i, j \in I} \|\mathbf{z}_j - \mathbf{z}_i\| \left(\frac{1}{d+2} \sum_{j \in I} \frac{\det(\mathbf{Z}_I^T \mathbf{Z}_I + \mathbf{1})}{\prod_{k \in I, k \neq j} \|\mathbf{z}_j - \mathbf{z}_k\|^2} \right)^{1/2}.$$

Числитель $\det(\mathbf{Z}_I^T \mathbf{Z}_I + \mathbf{1})$ является, с точностью до множителя, квадратом объема симплекса $(d+1)$ -го порядка, сформированного $(d+2)$ векторами

$\{\mathbf{z}_1, \dots, \mathbf{z}_{d+2}\}$. Следовательно, полярную кривизну можно рассматривать, как объем симплекса $(d+1)$ -го порядка, нормированного в каждой вершине, усредненного по вершинам и затем масштабированного диаметром симплекса $(d+1)$ -го порядка. Если $(d+2)$ векторов выбраны из одного и того же аффинного подпространства, то полярная кривизна $c_p \approx 0$ и, следовательно, аффинность (близость) ≈ 1 . Вместе с тем, когда вектора выбраны из объединения аффинных подпространств, то полярная кривизна велика и аффинность (близость) ≈ 0 .

Алгоритм K аффинных подпространств разделяет набор данных $\mathbb{Z} = \{\mathbf{z}_1, \dots, \mathbf{z}_N\} \subset \mathbb{R}^D$ на K поднаборов (кластеров) $\mathbb{C}_1, \dots, \mathbb{C}_K$, каждый из которых лучше всего аппроксимируется его d -мерным аффинным подпространством $\mathbb{F}_k$, $k = \overline{1, K}$. При заданных K и d этот алгоритм минимизирует целевую функцию

$$e = \sum_{k=1}^K \min_{\mathbb{F}_k} \sum_{\mathbf{z}_j \in \mathbb{C}_k} \|\mathbf{z}_j - P_{\mathbb{F}_k} \mathbf{z}_j\|^2,$$

где $P_{\mathbb{F}_k} \mathbf{z}_j$ — проекция вектора $\mathbf{z}_j$ на d -мерное аффинное подпространство $\mathbb{F}_k$, $k = \overline{1, K}$. На практике минимизация целевой функции выполняется итеративно, как в алгоритме K -средних [8]. То есть после инициализации K d -мерных аффинных подпространств (например, они могут быть выбраны случайным образом) повторяются два шага до достижения сходимости: 1) назначаются кластеры в соответствии с минимальным расстоянием до аффинных подпространств, определенных на предыдущей итерации; 2) для этих вновь полученных кластеров посредством анализа главных компонент — *Principal Component Analysis* (PCA [9]) вычисляются d -мерные аффинные подпространства с минимальной средней квадратичной ошибкой. Эта процедура очень быстрая и гарантированно сходится, по крайней мере, к локальному минимуму. Однако на практике локальный минимум, к которому сходится алгоритм K аффинных подпространств, гораздо хуже глобального минимума целевой функции. В результате, этот алгоритм не такой точный, как более ранние алгоритмы гибридного линейного моделирования, и даже при моделировании поверх линейных подпространств (в противоположность общим аффинным подпространствам) он часто дает сбой, когда, или d достаточно велико (например $d \geq 10$), или имеется значительная составляющая посторонних векторов [10, 11].

Ниже предполагается, что набор векторов $\mathbb{Z} = \{\mathbf{z}_1, \dots, \mathbf{z}_N\} \subset \mathbb{R}^D$ выбран из объединения K d -мерных аффинных подпространств $\mathbb{F}_k$, $k = \overline{1, K}$ (возможно с шумом и посторонними векторами), где $K > 1$ и N велико. Используя полярную кривизну c_p (2) с фиксированной константой моделирования $\sigma > 0$, конструируется тензор аффинности

(близости) $\mathcal{A}$ порядка $(d + 2)$ для различных $(d + 2)$ векторов $\mathbf{z}_{i_1}, \dots, \mathbf{z}_{i_{d+2}} \in \mathbb{Z}$ с компонентами

$$\mathcal{A}(i_1, \dots, i_{d+2}) = \exp(-(c_p(\mathbf{z}_{i_1}, \dots, \mathbf{z}_{i_{d+2}}))^2 / (2\sigma^2)). \quad (3)$$

Выбор оптимального значения параметра σ об-суждается в работе [2]. В выражении (3) тензор близости (подобия) $\mathcal{A}$ порядка $(d + 2)$ имеет раз-мерность $N \times N \times \dots \times N$. Однако в настоящей ра-боте используется только матричное представле-ние тензора $\mathcal{A}$ (3), которое обозначается $\mathbf{A}$ и назы-вается матрицей близости (подобия). Размерность матрицы $\mathbf{A}$ равна $N \times N^{d+1}$. Для каждого $i = \overline{1, N}$ строка i матрицы $\mathbf{A}$ (т. е. $\mathbf{A}(i, :)$) развертывается из слоя i тензора $\mathcal{A}$ (3) (т. е. $\mathcal{A}(i, :, \dots, :)$), следуя не-которому произвольному, но фиксированному по-рядку, например лексикографическому порядку последних $(d + 1)$ индексов [12]. Это упорядочение несущественно для настоящего рассмотрения, по-скольку как показано в работах [4, 5], в спектраль-ной кластеризации используется произведение

$$\mathbf{W} = \mathbf{A}\mathbf{A}^T, \quad (4)$$

которое не зависит от порядка индексов. Опреде-ление матрицы $\mathbf{A}$ и умножение этой матрицы боль-шой размерности на результат ее транспонирова-ния (чтобы вычислить $\mathbf{W}$ (4)) выполнить сложно. Возможное решение состоит в однородной выборке, т. е. в случайной выборке и вычислении небольшого поднабора столбцов $\mathbf{A}$, чтобы провести оценку $\mathbf{W}$ (4) [3, 13]. Обозначая посредством $\mathbf{A}(:, j)$ столбец j ма-трицы $\mathbf{A}$, записываем $\mathbf{W}$ (4) в следующем виде:

$$\mathbf{W} = \sum_{j=1}^{N^{d+1}} \mathbf{A}(:, j)\mathbf{A}(:, j)^T. \quad (5)$$

Следовательно, $\mathbf{W}$ — сумма N^{d+1} матриц ранга 1. Пусть $j_1, \dots, j_c$ — c целых чисел, случайно выбран-ных из интервала $[1, N^{d+1}]$. Как показано в работе [13], матрица $\mathbf{W}$ (5) аппроксимируется следующим образом:

$$\mathbf{W} \approx \sum_{t=1}^c \mathbf{A}(:, j_t)\mathbf{A}(:, j_t)^T. \quad (6)$$

В работе [2] алгоритм спектральной кластериза-ции с матрицей близости на основе полярной кри-визны (2) — *Spectral Curvature Clustering* (SCC) фор-мирует матрицу парных весов $\mathbf{W}$ (6) из аппроксими-рованной матрицы близости (подобия) $\mathbf{W} = \mathbf{A}_c\mathbf{A}_c^T$, и применяет спектральную кластеризацию [4] для определения K кластеров $\mathbb{C}_1, \dots, \mathbb{C}_K$. Для того чтобы улучшить эти кластеры, алгоритм SCC затем по-вторно выбирает векторы из кластеров $\mathbb{C}_k, k = \overline{1, K}$, в пределах небольшой полосы вокруг каждого из их аппроксимирующих d -мерных аффинных подпро-странств $\mathbb{F}_k, k = \overline{1, K}$. Эта процедура повторяется до достижения сходимости и называется итеративной выборкой (*iterative sampling*) [2]. Сходимость изме-

ряется ошибкой ортогональных наименьших квад-ратов — *Orthogonal Least Squares* (OLS) для d -мер-ных аффинных подпространств $\mathbb{F}_k, k = \overline{1, K}$, аппрок-симирующих кластеры $\mathbb{C}_1, \dots, \mathbb{C}_K$ в виде

$$e_{\text{OLS}} = \sum_{k=1}^K \sum_{\mathbf{z}_j \in \mathbb{C}_k} \|\mathbf{z}_j - P_{\mathbb{F}_k} \mathbf{z}_j\|^2, \quad (7)$$

где $P_{\mathbb{F}_k} \mathbf{z}_j$ — проекция вектора $\mathbf{z}_j$ на d -мерное аф-финное подпространство $\mathbb{F}_k, k = \overline{1, K}$ (может быть получена анализом главных компонент (PCA) [8]).

Алгоритм спектральной кластеризации с матри-цей близости (подобия) SCC на основе полярной кривизны представлен ниже [2].

Вход: Набор векторов $\mathbb{Z}$, внутренняя размер-ность (*intrinsic dimension*) d , число K d -мерных аф-финных подпространств, число выбираемых столб-цов c (по умолчанию = 100K).

Выход: K непересекающихся кластеров $\mathbb{C}_1, \dots, \mathbb{C}_K$ и ошибка e_{OLS} .

Шаги:

1. Случайным образом выбираются c поднабо-ров из $\mathbb{Z}$, каждый из которых содержит в точности $(d + 1)$ различных векторов.

2. Вычисляются полярная кривизна (2) каждого поднабора и каждого из оставшихся $(N - d - 1)$ векторов в $\mathbb{Z}$, эти $(N - d - 1)c$ значений полярной кривизны сортируются по возрастанию и форми-руют вектор $\mathbf{c}_p$.

3. **for** $q = 1: (d + 1)\mathbf{do}$

- Используется (3) с $\sigma = \mathbf{c}_p(Nc/K^q)$ для вычис-ления c выбранных столбцов $\mathbf{A}$. Используя эти c столбцов, формируется матрица $\mathbf{A}_c \in \mathbb{R}^{N \times c}$ (6).
- Вычисляется матрица $\mathbf{D} = \text{diag}(\mathbf{A}_c(\mathbf{A}_c^T \mathbf{1}))$, где $\mathbf{1}$ — вектор из единиц, и эта матрица исполь-зуется для нормировки матрицы $\mathbf{A}_c$: $\tilde{\mathbf{A}}_c = \mathbf{D}^{-1/2} \mathbf{A}_c$.
- Формируется матрица $\mathbf{U}$, столбцы которой — K старших левых сингулярных векторов $\tilde{\mathbf{A}}_c$.
- К строкам $\mathbf{U}$ применяется алгоритм K -средних (возможно, нормированным на единичную длину), и они разделяются на K кластеров.
- Эти кластеры (найденные) используются для группировки векторов набора $\mathbb{Z}$ в K поднабо-ров, и вычисляется соответствующая ошиб-ка e_{OLS} (7).

end for

Регистрируются K поднаборов $\mathbb{C}_1, \dots, \mathbb{C}_K$ набора $\mathbb{Z}$, которые соответствуют наименьшей ошибке e_{OLS} (7) в приведенном выше цикле.

4. Из каждого найденного $\mathbb{C}_k, k = \overline{1, K}$, выби-раются вектора в пределах небольшой полосы во-круг каждого из их OLS-аппроксимирующих d -мер-ных аффинных подпространств $\mathbb{F}_k, k = \overline{1, K}$, и вы-полняются шаги 2 и 3, чтобы найти K новых кла-стеров. Итерации повторяются до достижения сходимости.

2. Численный эксперимент

В настоящей работе анализ тонкой структуры кластеров траекторий движения самолетов, полученных методом полиномиальных регрессий [14], выполняется с помощью алгоритма спектральной кластеризации с тензором близости (подобия) на основе полярной кривизны (2).

Используются траектории 117 самолетов, идущих на посадку в международном аэропорту и зарегистрированных радаром TRACON 1 января 2006 года (<https://c3.nasa.gov/dashlink/resources/132/>). Начало координат совпадает с положением радара, интервал времени между точками регистрации составляет около 5 с. В работе учитываются только 160 последних точек каждой траектории, что позволяет исключить случайные маневры самолетов перед заходом на посадку. Эти траектории в трехмерном пространстве представлены в работе [14]. Пять кластеров траекторий самолетов (рис. 1, см. третью сторону обложки), выделяются в результате применения метода полиномиальных регрессий. Каждый кластер соответствует определенному "посадочному" паттерну или маршруту. Распределение числа траекторий по кластерам следующее: 16 траекторий в розовом кластере; 13 — в зеленом; 3 — в синем; 37 — в черном и 38 — в красном кластерах.

На рис. 2 (см. третью сторону обложки) показаны проекции траекторий в анализируемых кластерах (см. рис. 1) на координатные оси x , y и z в последовательные моменты времени k регистрации радаром. Линии тренда по методу полиномиальных регрессий (выделены жирными линиями) представляют обобщенную форму траекторий в каждом кластере [14]. Сходство траекторий движения самолетов в кластерах отражает несколько типичных маршрутов посадки ("посадочных" паттернов).

В работах [15, 16] по модели линейных и нелинейных динамических систем в четырех из пяти рассматриваемых кластеров при рассмотрении проекций траекторий на координатные оси x , y , z выявляется их тонкая (неоднородная) структура.

Восстановление двух первых столбцов исходных данных после разложения сингулярных чисел (svd-разложения) и сокращение числа сингулярных чисел до двух показано на рис. 3 (в коде MATLAB исходные данные — переменная x).

На рис. 4 (см. третью сторону обложки) для первичного красного кластера (см. рис. 1 и рис. 2) для x -компоненты траекторий при внутренней размерности аффинных подпространств $d = 5$ показаны три новых подкластера: красный, синий и зеленый. На рис. 4, а показана кластеризация алгорит-

```
[N,D] = size(X);  
[U,S] = svds(X - repmat(mean(X,1),N,1),6);  
data = U(:,1:2).*repmat(transpose(diag(S(1:2,1:2))),  
N,1);
```

Рис. 3. Восстановление исходных данных после разложения сингулярных чисел и сокращение их числа до двух

мом SCC в координатах: $data_1 = data(:, 1)$ и $data_2 = data(:, 2)$ (см. рис. 3). На рис. 4, в алгоритм SCC в общей модели 5-мерных аффинных подпространств ошибочно относит розовую траекторию к красному подкластеру, и в этом подкластере по мере косинуса она оказывается кандидатом на постороннюю. В результате нормальная траектория зеленого подкластера идентифицируется как посторонняя (показана желтым цветом на рис. 4, в). В синем подкластере голубым цветом показан кандидат на постороннюю траекторию. Классификация алгоритмом LSCC, предложенным в работе [2] (см. рис. 4, б), показана в координатах двух первых столбцов восстановленных данных (см. рис. 3). На рис. 4, г алгоритм LSCC в модели 5-мерных линейных подпространств правильно относит эту траекторию к зеленому подкластеру, но в этом подкластере она оказывается кандидатом на постороннюю (показана желтым цветом). Теперь в красном подкластере по мере косинуса правильно определяется кандидат на постороннюю траекторию (показана розовым цветом).

Заключение

В работе показано, как кластеризация траекторий движения самолетов при посадке в аэропорту осуществляется геометрическим методом, при котором набор многомерных векторов моделируется объединением d -мерных аффинных подпространств, заменяемых центроидами как прототипами кластеров. Поэтому учитывается не бинарное, а d -арное подобие, и принадлежность рассматриваемого набора траекторий (многомерных векторов) к кластеру оценивается тензором (многомерным) близости порядка $(d + 2)$, который разворачивается в матрицу близости (подобия) векторов, обеспечивая эффективное использование спектрального метода. Оценка близости в кластере (по мере косинуса) обнаруживает потенциально посторонние траектории движения самолетов в зоне риска.

Список литературы

1. Ma Y., Yang A. Y., Derksen H., Fossum R. Estimation of sub-space arrangements with applications in modeling and segmenting mixed data // SIAM Review. 2008. Vol. 50, N. 3. P. 413–458.
2. Chen G., Lerman G. Spectral curvature clustering (SCC) // International Journal on Computer Vision. 2009. Vol. 81, N. 3. P. 317–330.
3. Govindu V. A tensor decomposition for geometric grouping and segmentation // Proc. of the 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05). 20–26 June 2005, San Diego, CA. Washington, DC; IEEE Computer Society. 2005. V. 1. P. 1150–1157.
4. Ng A., Jordan M., Weiss Y. On spectral clustering: Analysis and an algorithm / Eds. Dietterich T., Becker S. Ghahramani Z. Advances in Neural Information Processing Systems. Cambridge, MA: MIT Press. 2002. V. 14. P. 1–8.
5. Luxburg U. A tutorial on spectral clustering // Statistics and Computing. December 2007. Vol. 17, Is. 4. P. 395–416.
6. Daverman R. J., Sher R. B. Handbook on Geometric Topology. Amsterdam: Elsevier Science BV, 2002, 1133 p.

7. **Whitehouse J. T.** Generalized Sines, Multiway Curvatures, and the Multiscale Geometry of d-Regular Measures. PhD Dissertation. University of Minnesota: Faculty of the Graduate School, 2009.
8. **Kroonenberg P. M.** Applied Multiway Data Analysis. Hoboken, NJ: Wiley, 2008. 557 p.
9. **Jolliffe I. T.** Principal Component Analysis. New York, Berlin, Heidelberg: Springer, 2002. 406 p.
10. **Zhang T., Szlam A., Lerman G.** Median K -flats for hybrid linear modeling with many outliers // Proc. of the 2009 IEEE 12th International Conference on Computer Vision, Sept. 27 — Oct. 4, 2009. Washington, DC: IEEE Computer Society 2009. P. 234—241.
11. **Zhang T., Szlam A., Wang Y., Lerman G.** Hybrid linear modeling via local best-fit flats // International Journal of Computer Vision. 2012. Vol. 100, N. 3. P. 217—240.
12. **Bader B., Kolda T.** Matlab Tensor Classes for Fast Algorithm Prototyping. Technical Report SAND2004-5187. Albuquerque, NM: Sandia National Laboratories, 2004.

13. **Drineas P., Kannan R., Mahoney M.** Fast Monte Carlo algorithms for matrices I: Approximating matrix multiplication // SIAM Journal on Computing. 2006. Vol. 36, N. 1. P. 132—157.
14. **Кухаренко Б. Г., Солнцева М. О.** Кластеризация управляемых объектов на основе сходства их многомерных траекторий // Информационные технологии. 2014. № 5. С. 3—7.
15. **Кухаренко Б. Г., Солнцева М. О.** Анализ результатов кластеризации многомерных траекторий посредством моделей линейных динамических систем // Информационные технологии. 2015. Т. 21, № 2. С. 104—109.
16. **Кухаренко Б. Г., Солнцева М. О.** Применение моделей нелинейных динамических систем для анализа результатов кластеризации многомерных траекторий // Информационные технологии. 2015. Т. 21, № 5. С. 341—345.

B. G. Kukharensko, e-mail: kukharenskobg@hotmail.com,

Leading Research Scientist, Blagonravov Mechanical Engineering Research Institute of the RAS,

M. O. Solntseva-Chalei, Graduate Student, e-mail: solntseva.chalei@gmail.com,

Moscow Institute of Physics and Technology (SU)

Spectral Method with Polar Curvature Used for Analysis of Multi-Dimensional Trajectory Clustering Results

Problem of clustering trajectories is pre-conditioned by a need to organize motion of objects under control. The polynomial regression method is the best approach to trajectory cluster selection, which estimates a form of general trajectory of each cluster. For a set of sufficiently heterogeneous trajectories, the defined clusters are heterogeneous also. For the inhomogeneous cluster, its polynomial regression is a too strong abstraction. To demonstrate the cluster heterogeneity (and, thus, non full clustering trajectories) a method of dimension reducing is in need. As K d-flats algorithm is a generalization of K -means algorithm where d-dimensional best fit affine sets replace centroids as the cluster prototypes, a similar modification of spectral clustering method is in use in paper. This is geometric method to obtain vector clustering model in form of affine space union. Vector set similarity is estimated by multi-dimensional affinity tensor, for which matrix representation is in use. Next it is analyzed by spectral method. As example, clustering results of airplane fly trajectories in an airport space is under study. Defining the cluster central trajectory with affinity estimation by cosine measure in use gives an opportunity to determine the cluster outlying trajectories in potential, which represent fly trajectories located in a risk zone.

Keywords: data mining, multi-dimensional trajectories, model of affine space union, polar curvature, spectral clustering

References

1. **Ma Y., Yang A. Y., Derksen H., Fossum R.** Estimation of sub-space arrangements with applications in modeling and segmenting mixed data, *SIAM Review*, 2008, vol. 50, no. 3, pp. 413—458.
2. **Chen G., Lerman G.** Spectral curvature clustering (SCC), *International Journal on Computer Vision*, 2009, vol. 81, no. 3, pp. 317—330.
3. **Govindu V.** A tensor decomposition for geometric grouping and segmentation, *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. 20—26 June 2005, San Diego, CA, Washington, DC: IEEE Computer Society, 2005, vol. 1, pp. 1150—1157.
4. **Ng A., Jordan M., Weiss Y.** On spectral clustering: Analysis and an algorithm / Dietterich T., Becker S. Ghahramani Z., eds. *Advances in Neural Information Processing Systems*. Cambridge, MA: MIT Press. 2002, vol. 14, pp. 1—8.
5. **Luxburg U.** A tutorial on spectral clustering, *Statistics and Computing*, December 2007, vol. 17, is. 4, pp. 395—416.
6. **Daverman R. J., Sher R. B.** Handbook on Geometric Topology. Amsterdam: Elsevier Science BV, 2002, 1133 p.
7. **Whitehouse J. T.** Generalized Sines, Multiway Curvatures, and the Multiscale Geometry of d-Regular Measures. PhD Dissertation. University of Minnesota: Faculty of the Graduate School, 2009.
8. **Kroonenberg P. M.** Applied Multiway Data Analysis. Hoboken, NJ: Wiley, 2008. 557 p.

9. **Jolliffe I. T.** Principal Component Analysis. New York, Berlin, Heidelberg: Springer, 2002, 406 p.
10. **Zhang T., Szlam A., Lerman G.** Median K -flats for hybrid linear modeling with many outliers, *Proceedings of the 2009 IEEE 12th International Conference on Computer Vision*, Sept. 27 — Oct. 4, 2009. Washington, DC: IEEE Computer Society, 2009, pp. 234—241.
11. **Zhang T., Szlam A., Wang Y., Lerman G.** Hybrid linear modeling via local best-fit flats, *International Journal of Computer Vision*, 2012, vol. 100, no. 3. P. 217—240.
12. **Bader B., Kolda T.** Matlab Tensor Classes for Fast Algorithm Prototyping. Technical Report SAND2004-5187. Albuquerque, NM: Sandia National Laboratories, 2004.
13. **Drineas P., Kannan R., Mahoney M.** Fast Monte Carlo algorithms for matrices I: Approximating matrix multiplication, *SIAM Journal on Computing*, 2006, vol. 36, no. 1, pp. 132—157.
14. **Кухаренко Б. Г., Солнцева М. О.** Кластеризация управляемых объектов на основе сходства их многомерных траекторий, *Информационные технологии*, 2014, no. 5, pp. 3—7.
15. **Кухаренко Б. Г., Солнцева М. О.** Анализ результатов кластеризации многомерных траекторий посредством моделей линейных динамических систем, *Информационные технологии*, 2015, no. 2, pp. 104—109.
16. **Кухаренко Б. Г., Солнцева М. О.** Применение моделей нелинейных динамических систем для анализа результатов кластеризации многомерных траекторий, *Информационные технологии*, 2015, vol. 21, no. 5, pp. 341—345.

А. А. Моисеев, канд. техн. наук, ст. науч. сотр., e-mail: slow.coach@yandex.ru,
ГосНИИ химмотологии, Москва

Модель продаж стандартными партиями

Рассматривается метод оптимизации продаж мелких партий товара в ситуации плавающей цены на него. Целью продаж в конечном итоге является максимизация прибыли, и предлагаемые мероприятия по оптимизации продаж подчинены этому критерию. В качестве метода реализации товара рассматривается его распродажа мелкими фиксированными долями — лотами, в ходе которой темп роста прибыли определяется текущей ценой партии и темпом сделок. В качестве товара могут выступать, например, партии акций или валюты, обладающие свойствами стандартности, серийности и ликвидности и продаваемые по биржевому курсу в рамках сделок по схеме SPOT (at a spot — на месте, немедленная поставка товара с немедленной оплатой). Оптимизационная процедура определяет текущее предложение товара и блокирует продажи в случае снижения цены ниже допустимого уровня. При разработке этой процедуры используется предположение, что запас товара ограничен и имеется также ограничение на время проведения торгов. Цена продажи предполагается плавающей, однако при ее падении ниже минимально допустимой продажи блокируются. В случае же, если разность текущей и минимальной цен превышает некоторую величину, продажи осуществляются в предельном темпе. Предусматривается также активизация продаж в конце торгов. В рамках этой работы учитывается также влияние встречных покупок товара на тех же условиях. Процедура реализована в форме трехпозиционного широтно-импульсного регулятора, формирующего команды на продажу, покупку и приостановление активности. Проведенные в рамках разработки численные эксперименты показывают, что организованная таким образом оптимизационная процедура обнаруживает поведение, качественно соответствующее интуитивно разумному поведению на торгах. Вероятно при условии предварительной настройки к условиям конкретных торгов, она могла бы быть использована для моделирования динамики котировок и в перспективе послужить аналитической основой для создания торгового алгоритма.

Ключевые слова: численный эксперимент, идентификационная модель, торговый алгоритм, широтно-импульсный регулятор, оптимизация, торги, лот, SPOT, транзакция, ликвидность

Целью продаж в конечном итоге является максимизация прибыли и все мероприятия по оптимизации продаж должны быть, естественно, подчинены этому критерию. Возможным методом реализации товара является его распродажа мелкими фиксированными долями — партиями, в ходе которой темп роста прибыли определяется текущей ценой партии и темпом сделок. В качестве товара могут выступать, например, партии акций или валюты, обладающих свойствами стандартности, серийности и ликвидности [1] и продаваемые по биржевому курсу в рамках сделок по схеме SPOT [2]. В рамках этой работы учитывается также влияние встречных покупок товара на тех же условиях [3].

Оптимизирующая процедура

При разработке оптимизирующей процедуры будем исходить из предположения, что запас товара ограничен и имеется также ограничение на время проведения торгов. Цена продажи x предполагается плавающей, однако при ее падении ниже минимально допустимой продажи блокируются. В случае если разность текущей и минимальной цен превышает некоторое значение Δ продажи осуществляются в предельном темпе.

Возможная схема оптимизации продаж приведена на рис. 1. Ее основой является генератор импульсов [4] разблокировки продаж **unlock** с адаптивно изменяемой скважностью, определяемой величиной

$$d = \min\left(1, \frac{\Delta}{x - x_0 f}\right),$$

где f — значение штрафной функции, определяемой ниже; x_0 — начальное значение минимальной цены.

Этот генератор выступает в качестве широтно-импульсного модулятора, управляющего генерато-

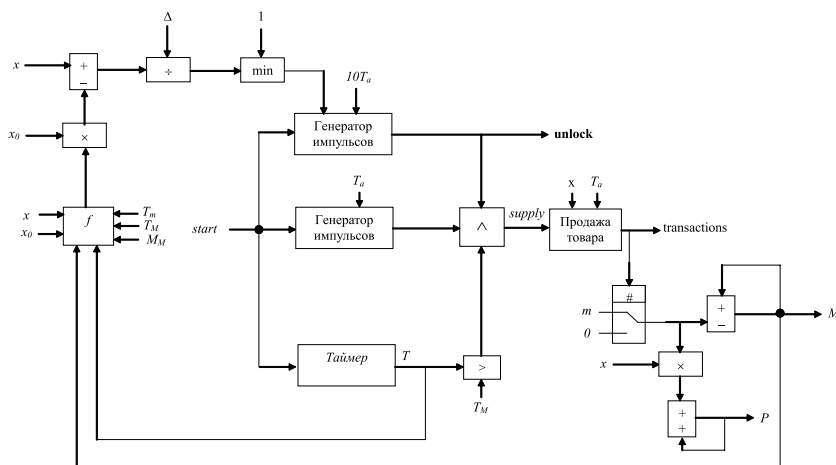


Рис. 1. Схема оптимизации продаж

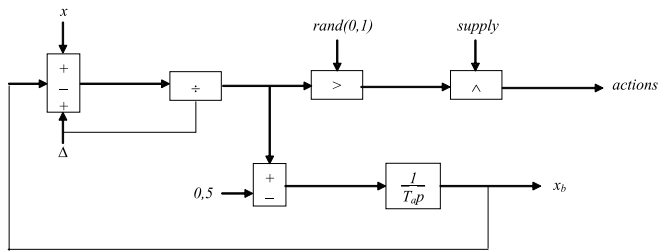


Рис. 2. Модель решения о покупке товара

ром импульсов предложения товара в предельном темпе. Длительность этих импульсов соответствует длительности совершения сделки T_a , а период следования определяется темпом сделок и в рассматриваемой модели составляет $\sim 10T_a$. Модуляция осуществляется путем логического перемножения выходных импульсов генераторов с формированием на выходе потока предложений $supply$.

Принятие решения о покупке при наличии предложения моделируется процедурой, отображенной на рис. 2. Покупка осуществляется, если стандартное случайное число $rand$, равномерно распределенное в интервале $(0,1)$, больше порога, образованного относительным превышением текущей цены x минимальной цены покупателя $x_b - \Delta$, где x_b — максимальная допустимая цена. Для моделирования вариации последней в ходе продаж на вход формирующего интегратора подается центрированное превышение минимальной цены. Признак превышения случайным числом относительного превышения логически перемножается с признаками предложения $supply$, формируя поток сделок $actions$. Этот поток управляет мультиплексором, входящим в состав калькулятора текущего запаса товара. При подаче импульса, соответствующего сделке, запас товара M уменьшается на объем партии m . Запас товара и объем партии выражаются в процентах от начального запаса товара M_M .

Пуск продаж осуществляется командным импульсом $start$ (см. рис. 1), который запускает оба генератора и таймер, рассчитывающий текущее время операции T . Завершение продаж осуществляется по истечении заданного времени операции T_M .

При увеличении превышения текущей цены x над текущей минимальной ценой x_0 скважность адаптивного модулятора снижается и он формирует более длительные импульсы. Средний темп продаж при этом возрастает. Если же указанное превышение превзойдет значение Δ , продажи разблокируются полностью и осуществляются в предельном темпе. Уменьшение превышения напротив ведет к увеличению скважности и снижению длительности разблокирующих импульсов. Средний темп про-

даж при этом убывает, стремясь к нулю при приближении текущей цены к минимальной.

Расчет прибыли P осуществляется путем ее прогрессивного наращивания на величину $x \cdot m$ в случае, если имеет место продажа партии, т. е. при $actions = 1$. Расчет прибыли выполняется с нулевого значения сумматором, охваченным положительной обратной связью.

Наложенные ограничения на время операций и запас товара требуют обеспечения плавающего уровня минимальной цены. Для решения этой задачи предлагается использовать мультипликативную штрафную функцию f [5], вида $f = 1 + \Delta f_T + \Delta f_M$, схема формирования которой отображена на рис. 3.

Влияние истечения времени торгов учитывается компонентой Δf_T , определяемой соотношением:

$$\Delta f_T = \frac{y}{1-y};$$

$$y = \max\left(0, \frac{T - T_m}{T_M - T_m}\right),$$

где $T_m \sim 0,7 \div 0,8 T_M$ — время перехода к завершающей стадии торгов.

Эта компонента позволяет снизить минимальную цену и тем самым активизировать продажи перед окончанием торгов.

Влияние наличного запаса товара учитывается компонентой Δf_M , основу которой составляет оценка разности модулей текущего темпа убывания относительного запаса товара и половины максимального темпа убывания указанного запаса. Последний находится экспериментально при полностью разблокированных продажах. Текущий темп убывания формируется на выходе форсирующего устройства с постоянной времени, равной 1. Эта величина подается на вход интегратора, формирующего компоненту Δf_M , причем вход интеграто-

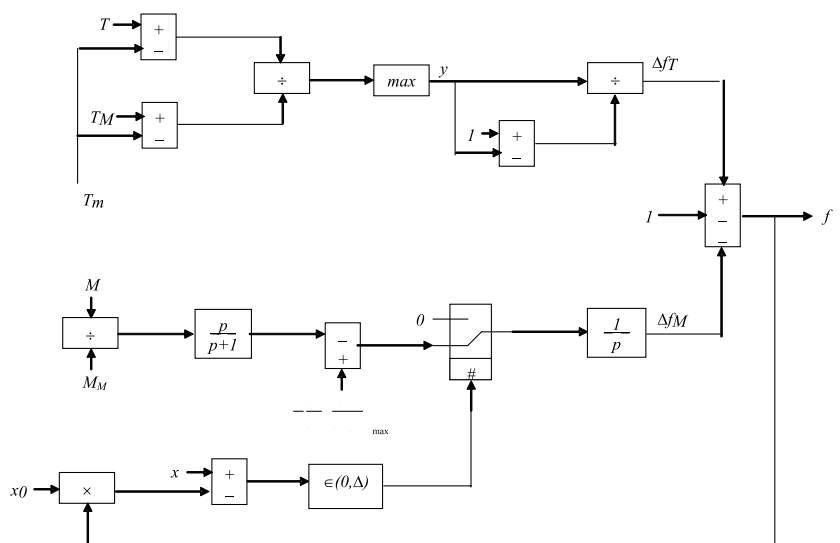


Рис. 3. Схема формирования штрафной функции

ра обнуляется (т. е. выход фиксируется), если разность текущей цены x и плавающей максимальной цены $x_0 f$ лежит в диапазоне $(0, \Delta)$. Построенная таким образом компонента позволяет варьировать уровень минимальной цены и, следовательно, объем продаж в ситуации значительного отклонения текущей цены от минимальной, а также отчасти разблокировать продажи в ситуации значительного снижения текущей цены.

Результаты численных экспериментов

Результаты численных экспериментов по исследованию временной зависимости продаж с использованием построенного оптимизатора приведены на рис. 4–6. На рис. 4 отображена временная зависимость продаж в случае благоприятной конъюнктуры цен, когда имеет место тенденция их роста с кратковременными колебаниями. На рис. 4, а отображена динамика текущей и допустимой цены, а на рисунке 4, б — динамика запаса товара и текущей прибыли. Их анализ показывает, что наблюдается некоторый рост минимальной цены, а продажи происходят практически в предельном темпе. Некоторое снижение текущей цены имеет место в конце торгов, однако увеличение темпа продаж, вызванное уменьшением допустимой цены, компенсирует влияние этого снижения.

Временная зависимость продаж при колеблющейся конъюнктуре отображена на рис. 5. В соответствии с рис. 5, а, в этом случае отсутствуют тенденции к росту или понижению цены и ее изменения имеют вид кратковременных колебаний. Темп продаж в этом случае возрастает на участках роста

цены и снижается на участках ее убывания. Рост темпа продаж в конце торгов вызывается, как и в первом случае, снижением допустимой цены. Аналогичную динамику обнаруживает прибыль, отображенная на рис. 5, б, результирующая величина которой существенно ниже, чем в первом случае.

Динамика продаж при неблагоприятной конъюнктуре, когда имеет место тенденция к их снижению, отображена на рис. 6. В соответствии с рис. 6, а

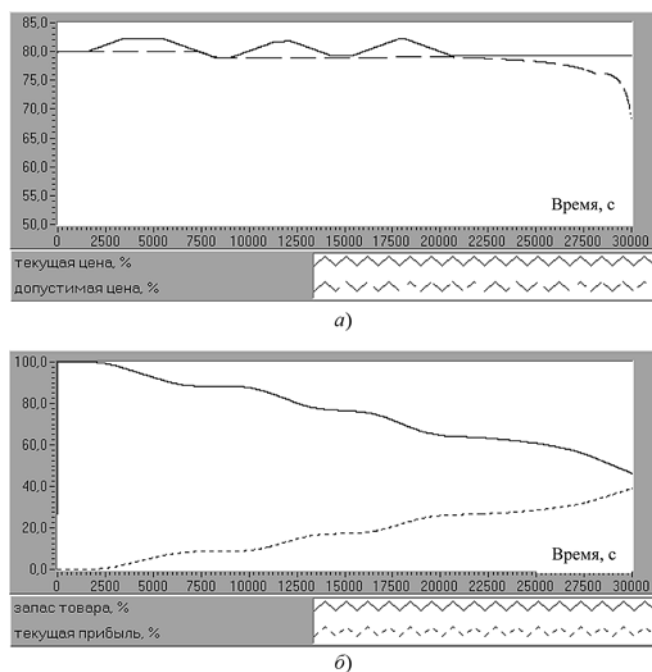


Рис. 5. Временная зависимость продаж при колеблющейся конъюнктуре

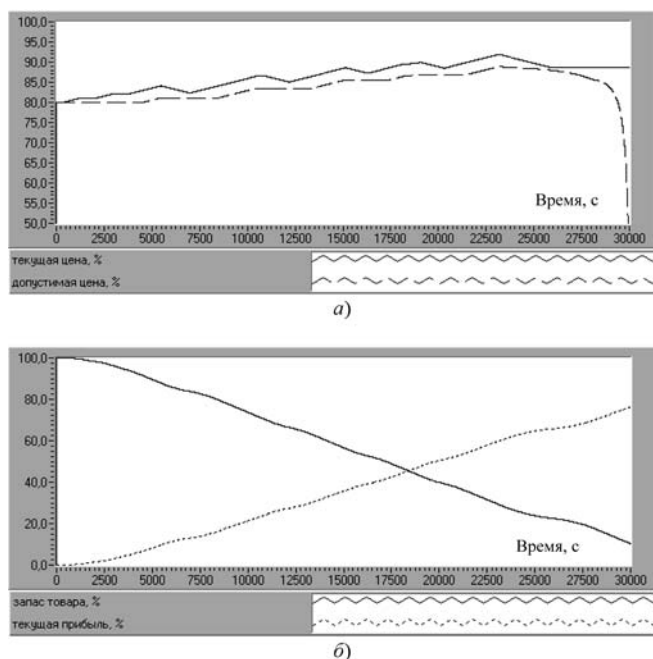


Рис. 4. Временная зависимость продаж при благоприятной конъюнктуре

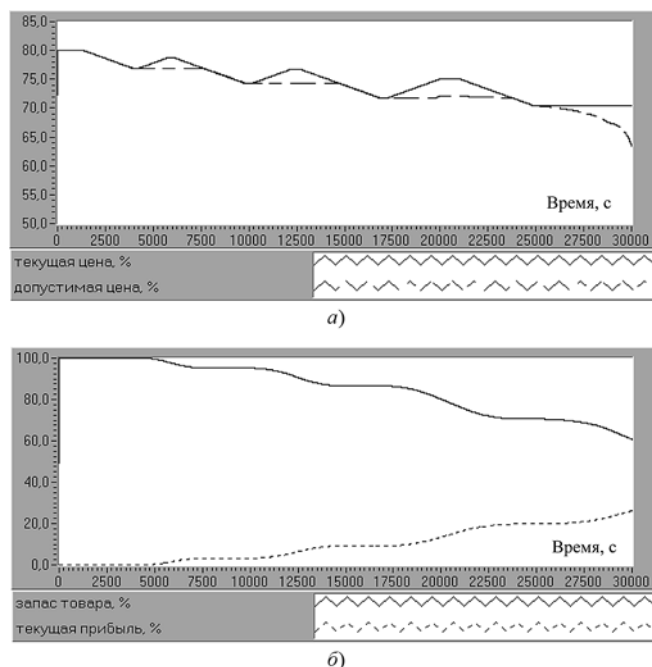


Рис. 6. Временная зависимость продаж при неблагоприятной конъюнктуре

продажи, как и во втором случае, происходят в основном на участках роста цены и блокируются на участках ее убывания. Рост темпа продаж в конце торгов вызывается, как и ранее, снижением допустимой цены. Результирующая прибыль, отображенная на рисунке 6, б, несколько ниже, как и следовало ожидать, чем во втором случае.

Проведенные численные эксперименты показывают, что оптимизационная процедура обнаруживает поведение, качественно соответствующее интуитивно разумному поведению на торгах. Вероятно, при условии предварительной настройки к условиям конкретных торгов, она могла бы быть использована для моделирования динамики коти-

ровок и в перспективе послужить аналитической основой для создания торгового алгоритма.

Список литературы

1. Миркин Я. М. Ценные бумаги и фондовый рынок. М.: Перспектива, 1995, 550 с.
2. Рынок ценных бумаг и биржевое дело / под. ред. О. И. Дегтяревой и др., М.: Юнити, 2004, 501 с.
3. Моисеев А. А. Виртуализация квалификационных испытаний // Промышленные АСУ и контроллеры. 2015. № 10. С. 40.
4. Моисеев А. А. Проблемно-ориентированное моделирование технологических процессов в интерактивном синтезе алгоритмов автоматического управления // Промышленные АСУ и контроллеры. 2015. № 2. С. 26.
5. Васильев Ф. П. Методы оптимизации. М.: Факториал, 2002. 824 с.

A. A. Moiseev, PhD., Senior Researcher, e-mail: slow.coach@yandex.ru
National Research Institute of Chimnotology

Mathematical Model of Auction Sale with Lots

Optimization method of lots sale considered in situation of volatile price. Usual sale aim is profit maximization and proposed optimization procedure must follow to this criterion. Lots sale is considered here as method of assets realization and profit rate defined with fluent price and transactions rate. Stock or currency lots are standard liquid assets and can be on sale in form of SPOT dealing. Optimization procedure determines fluent offer in situation when assets reserve and tender time are limited. Assets price is volatile but procedure can lock the sales in case of intolerable prices drop. It provides also maximal sales rate in case when difference between fluent and minimal price exceeds some threshold. Procedure provides also the sales activation at the tender completion and counter purchases on the same conditions. Procedure implemented in form of three — positional pulse-duration controller which generates commands on sales, purchases and activity suspension. Performed numerical experiments showed that optimization procedure provides behavior, which corresponds to intuitively reasonable one. Probably at preliminary adjustment to specific tender conditions such a procedure could be used for simulation of quotation dynamics. Besides it could be used in sight as a base for trade algorithm development.

Keywords: numerical experiment, simulation model, trade algorithm, pulse-duration controller, optimization, tender, lot, SPOT, transaction, liquidity

References

1. Mirkin Ya. Tsennye bumagi i fondowy rynek (Securities and stock market), Moscow, Perspektiva, 1995, 550 p.
2. Securities market and activities at stock exchanges, edited by Degtiareva O. I. ea, Moscow, Unity, 2004, 501 p.
3. Moiseev A. A. Virtualization of qualification tests, *Industrial ASC and controllers*, 2015, no. 10, p. 40.
4. Moiseev A. A. Probleme-oriented modeling of technological processes at interactive synthesis of control algorithms, *Industrial ASC and controllers*, 2015, no. 2, p. 26.
5. Vasiliev F. P. *Metody optimizatsii (Optimization methods)*, Moscow, Factorial, 2002, 824 p.

СИСТЕМЫ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ CAD-SYSTEMS

УДК 004.4'242

А. Л. Стемповский, д-р техн. наук, академик РАН, директор,
Д. В. Тельпухов, канд. техн. наук, нач. отдела, **Р. А. Соловьев**, канд. техн. наук, науч. сотр.,
В. С. Рухлов, аспирант, мл. науч. сотр., e-mail: do1p@ya.ru,
Институт проблем проектирования в микроэлектронике РАН

Тестовая система для сравнения алгоритмов, увеличивающих надежность комбинационных схем

Предложено описание системы, состоящей из большого набора тестов и программы судьи (judge program). Система используется для оценки эффективности работы произвольных алгоритмов защиты логических или арифметических схем без обратных связей. При разработке системы использовался опыт подготовки задач для соревнований по спортивному программированию. Приведены подробности по разработке закрытой системы тестов и по методике начисления очков. Раскрыты детали работы программы судьи. Разработанная тестовая система свободно доступна онлайн в системе SPOJ для всех специалистов по надежным схемам. Система ранжирует алгоритмы и выстраивает их в список по числу полученных очков. Чем больше очков получил алгоритм, тем более надежные схемы он сгенерировал на основе исходных.

Ключевые слова: надежность, резервирование, логические схемы, спортивное программирование, тестирование

Введение

Задача увеличения надежности микроэлектронных схем является одной из ключевых в некоторых отраслях промышленности, например в космической отрасли [1]. Множество исследователей занимаются проблемами защиты логических и арифметических схем [2, 17]. Разработано большое число разнообразных алгоритмов для этой задачи [3–5]. Методы увеличения надежности комбинационных схем в большинстве своем основаны на введении структурной избыточности, также есть методики, основанные на ресинтезе, изменении структуры элементов голосования на более надежные и т. д. Однако в большинстве своем сравнение разработанных методов проводится с архаичным методом Triple Modular Redundancy (TMR) [6]. В некоторых работах не учитывается влияние надежности декодеров на общую надежность схем, так как считается, что декодеры или мажоритарные элементы не подвержены сбоям [7, 8]. Как следствие, разработчикам микроэлектронных устройств трудно определить, какой из методов использовать для защиты микросхем, в связи с отсутствием явной классификации методов по их эффективности. Поэтому зачастую выбор делается в пользу TMR — как самого простого и при этом довольно эффективного метода. В работе предлагается система, которая будет ранжировать различные алгоритмы по их эффективности, и разработчики микроэлектронных схем

смогут использовать методики, более продвинутые, чем TMR.

В работе приводится детальное описание методики создания набора тестов, которые будут выступать своеобразным эталоном при сравнении различных алгоритмов защиты логических схем от сбоев. Реализация тестирования сделана на базе онлайн системы оценки решений SPOJ [9, 10], которая разработана специально для ранжирования эффективности алгоритмов. Мы следуем по пути крупных Интернет-соревнований, которые регулярно проводят Netflix, Яндекс, Google, TopCoder и другие крупные IT-гиганты на своих площадках.

Предполагается, что систему будут использовать различные группы исследователей, работающие над задачами защиты комбинационных схем. Для использования тестовой системы потребуется написать программу, реализующую разработанный метод и соответствующую формальным параметрам, которые будут описаны ниже. Ценность тестовой системы будет тем выше, чем больше исследователей пришлют свои программные реализации алгоритмов для защиты схем. Программа на первом месте рейтинга будет считаться наиболее эффективной. Таким образом, независимые исследователи получат возможность сравнить свои результаты с результатами других групп, не вступая в непосредственный контакт.

Формализованное описание задачи

Формализованное описание задачи для использования в автоматической системе ранжирования должно быть одновременно максимально простым и максимально полным. Система работает онлайн без вмешательства человека, поэтому нужно однозначно определить:

- набор и формат входных данных;
- набор и формат данных, который должен выдать алгоритм пользователя, т. е. выходные данные;
- состав тестовых последовательностей;
- систему начисления очков.

При этом все данные должны быть в машиночитаемом виде.

Входные данные. Поскольку мы ориентируемся на современные средства разработки с проектированием на основе стандартных ячеек, то схемы нужно задать в виде набора логических ячеек и связей между ними. Технологические библиотеки могут содержать сотни элементов. Чтобы не усложнять чтение входных данных и последующее решение задачи, выберем только небольшую часть базовых ячеек, которые заведомо присутствуют в технологических библиотеках. Пусть их будет шесть основных из набора: инвертор (INV), логическое "И", логическое "ИЛИ", "И-НЕ", "ИЛИ-НЕ" и логическое "ИЛИ" (рис. 1). Предполагается, что разработанные алгоритмы могут быть легко адаптированы для случая наличия в библиотеке сотен логических элементов, в том числе многовыходовых.

Наряду с остальными параметрами в технологической библиотеке, такими как площадь (S), каждая стандартная ячейка может характеризоваться, в том числе и вероятностью отказа (P). Скорее всего, это нелинейный параметр, который может зависеть от ряда факторов, но для простоты мы будем считать, что такой параметр задан однозначно — одним числом для одной ячейки.

Обычно заниматься увеличением надежности начинают после проектирования схемы, когда из-

вестен ее первоначальный размер. Из теории кодирования также известно, что чем больше избыточность, тем большей надежности можно достичь, однако производственные нормы накладывают ограничения на максимальный размер проекта. В задаче будем считать, что для каждой схемы нам задано число K — во сколько раз площадь защищенной схемы может превышать исходную. Мы также полагаем, что нас интересуют методы, альтернативные мажорированию, которые увеличивают площадь начальной схемы в разы. То есть K изменяется в довольно широких пределах от 2 и выше.

Чтобы разработчики алгоритма не занимались разработкой парсера для входных данных, подготовим их сразу в машиночитаемом формате. На первой строке укажем число тестовых схем Z . Далее следует Z схем, для каждой из которых сначала заданы требуемые параметры технологической библиотеки:

- в первой строке K — максимальная избыточность схемы по площади. На размер K требуется наложить некоторые ограничения, в нашем тестовом наборе $2 \leq K \leq 20$;
- в следующих шести строках указаны по два числа: площадь S ($1 \leq S \leq 100$) и вероятность сбоя q ($0 \leq q \leq 20$) в процентах, параметры каждого из логических элементов в следующем порядке: INV, AND, OR, NAND, NOR, XOR.

Затем идет описание самой схемы в следующем формате:

- число входов схемы: I ($0 < I \leq 250$), затем следует I названий входных узлов схемы (не более 20 символов в каждом), разделенных пробелами;
- число выходов схемы: O ($0 < O \leq 150$), затем следует O названий выходных узлов схемы (не более 20 символов в каждом), разделенных пробелами;
- число логических вентилях в схеме N , после чего следуют N строк с описанием каждого вентиля. Описание каждого вентиля начинается с его типа (одного из шести: INV, AND, OR, NAND, NOR, XOR), далее идут имена входных узлов, затем следует имя выходного узла. Входных узлов ровно два для каждого вентиля, кроме инвертора, где входной узел один. Выходной узел всегда один для всех вентилях.

В целом, описание схем напоминает структурный Verilog, но сделанный в более удобном для чтения из программ виде.

Выходные данные. Программа, реализующая алгоритм защиты, должна на основе исходной логической схемы (ИЛС) вывести модифицированную логическую схему (МЛС), которая выполняет ту же самую логическую функцию. То есть при любом наборе входных данных значения на всех выходах ИЛС и МЛС должны быть одинаковыми. Соответственно у МЛС должны быть в наличии все входные и выходные узлы, определенные в ИЛС. МЛС должна состоять из шести элементов, опре-

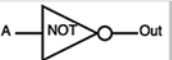
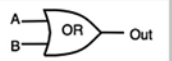
Название	Описание	Операция	Изображение
INV	Инвертор	$OUT = \neg A$	
AND	Логическое "И"	$OUT = A \& B$	
OR	Логическое "ИЛИ"	$OUT = A B$	
NAND	Инвертированное "И"	$OUT = \neg(A \& B)$	
NOR	Инвертированное "ИЛИ"	$OUT = \neg(A B)$	
XOR	Исключающее "ИЛИ"	$OUT = A \wedge B$	

Рис. 1. Базовые логические элементы

деленных в библиотеке, и ее площадь не должна превышать площадь более чем в K раз.

Формат вывода результирующей схемы похож на формат входных данных.

На первой строке число N — число логических элементов в МЛС. Далее следует N строк с описанием логических вентилях. Описание каждого вентиля начинается с его типа, далее идут имена входных узлов, затем — имя выходного узла.

Такую схему требуется вывести для каждого из Z тестов.

Описание и генерация схем. Поскольку задача поставлена для защиты комбинационных схем, то в наборе из тестовых схем используются только схемы без обратных связей. Для тестов было принято решение использовать схемы из известного бенчмарка ISCAS85 [11], а также несколько автоматически сгенерированных схем, поскольку набор ISCAS85 состоит всего из 11 схем, что недостаточно для полноценной тестовой системы. Некоторые схемы из набора ISCAS85 используются несколько раз, но с различными параметрами вероятности сбоя и избыточности.

В реальности ошибки в схемах происходят крайне редко. Поскольку процесс моделирования ошибок будет запускаться много раз, то нам желательно пропускать те случаи, когда схема работает верно. Иначе мы рискуем потратить на моделирование очень много времени. Поэтому для сочетания "библиотека плюс схема" необходимо подобрать вероятности таким образом, чтобы процесс моделирования не проходил впустую, т. е. чтобы хотя бы одна ошибка вносилась в схему достаточно часто и чтобы ошибок в схеме было не слишком много. Идеальным будет случай распределения ошибок, близкого к нормальному, пик которого находится около значения одна ошибка на схему.

В ходе исследований была разработана формула генерации вероятности появления ошибки для ячейки, чтобы выполнялись условия моделирования, близкие к требуемым:

$$P = \frac{1}{kN}, \quad (1)$$

где k — некий случайный коэффициент от 1 до 2; N — число логических элементов в схеме.

Для одной из схем с библиотекой, вероятности ошибок в которой получены по формуле (1), распределение числа ошибок во время моделирования Монте-Карло имеет вид, представленный на рис. 2.

Число ошибок на схему для всего тестового набора (даже для самых больших схем) редко превышает 12, и пик приходится на одиночную ошибку (SEU — single event upset) [20].

Для генерации случайных схем был разработан специализированный алгоритм. Для его реализации используется функция, которая принимает на



Рис. 2. График числа ошибок в зависимости от кратности для одного из тестов на 5000 запусков метода Монте-Карло

вход два числа W и H — желаемую "длину" и "высоту" логической схемы.

1. Число входов и выходов схемы берется как целое случайное число на промежутке от $H/2$ до H .

2. Генерируемая схема имеет W уровней, на каждом из которых число логических элементов считается по формуле $1 + \text{rand}(0, H - 1)$.

3. Тип логического элемента выбирается случайным образом из набора из шести элементов: INV, AND, NAND, AND, NOR, OR, XOR.

4. Соединения между элементами выбираются следующим образом. В массив `wires` помещаем все входные узлы. Далее двигаемся последовательно по уровням от входов к выходам схемы. Для входов элементов из каждого уровня случайным образом выбираем два различных элемента из массива `wires` (для инвертора один элемент). По окончании этого процесса для заданного уровня добавляем в массив `wires` все выходные узлы элементов из данного уровня.

Повторяем процесс для всех уровней, включая выходные узлы схемы.

Система начисления очков. Чтобы сравнить два алгоритма защиты схемы между собой на одной схеме, требуется сгенерировать защищенную версию исходной схемы каждым из алгоритмов, затем промоделировать обе эти схемы. Очевидно, что более эффективным будет алгоритм, который сгенерировал схему, число отказов на выходах которой в результате моделирования будет меньше.

Для оценки алгоритмов будем использовать метод Монте-Карло [12] и метрику COF (*correct output factor*) [13]. Метод Монте-Карло — это статистический метод, который характеризуется числом запусков моделирования TN . Чем больше TN , тем более точными получаются результаты моделирования. Обычно используют TN от 1000 и выше. По итогам каждого отдельного моделирования возможны два результата:

- на всех выходах схемы получены значения, совпадающие с эталоном, т. е. сбоя не было или сбой не привел к некорректному функционированию схемы;

- хотя бы на одном выходе результат отличается от эталона, т. е. в схеме произошел сбой, и он привел к неверному значению на выходе схемы.

Обозначим число сбойных тестов как ET . Тогда значение COF можно рассчитать по формуле

$$COF = \frac{TN - ET}{TN}. \quad (2)$$

Из формулы (2) видно, что значение COF лежит на промежутке от 0 до 1. Чем ближе значение к единице, тем более эффективно защищена схема. Надо также отметить, что в обычных условиях даже для схемы без защиты значение COF не равно 0. И даже для наиболее эффективно защищенных схем значение COF не равно единице из-за возможности сбоя на элементе, который подсоединен к выходу.

Чтобы сравнить эффективность алгоритмов защиты, недостаточно сравнить их работу только на одной схеме и при одних условиях работы. Для этого требуется некоторый набор тестовых схем. Так как значение COF нормировано на интервале от 0 до 1, то для сравнения достаточно посчитать эффективность SCORE как сумму COF_i для всех TN тестов:

$$SCORE = \sum_{i=0}^{TN} COF_i. \quad (3)$$

Из формулы (3) видно, что значение SCORE находится на интервале от 0 до N .

Задача в системе SPOJ

Система SPOJ работает без участия человека и оценивает присланные решения в автоматическом режиме. Поэтому задача требует полной формализации, а также защиты от накруток в системе начисления очков.

Для реализации задачи в системе SPOJ требуются:

- Input.txt — набор входных данных в текстовом формате;
- Output.txt — набор шаблонных выходных данных. Для нашей задачи этот файл не требуется, так как шаблон для выхода отсутствует;
- Judge.c — программа "Судья" на языке C (строго говоря программу "Судья" можно писать на любом языке, который поддерживает SPOJ).

На вход программы "Судья" подаются:

- *spoj_p_in — входные данные задачи (input.txt);
- *spoj_p_out — выходные данные задачи (output.txt);
- *spoj_t_out — выходные данные тестируемой программы;
- *spoj_t_src — исходный код тестируемой программы.

На выходе программа "Судья" может выдавать следующие данные:

- *spoj_score — число очков, которые заработала программа (обязательный параметр);

*spoj_p_info — информация о моделировании для автора задачи (по сути, закрытая информация);

*spoj_u_info — информация о моделировании для автора алгоритма (открытая для автора решения информация).

В нашей программе "Судья" используются: spoj_p_in — отсюда читаются исходные схемы в формате, описанном в разделе "Входные данные", файл spoj_t_out, который был сгенерирован тестируемой программой. В этом файле находятся описания для схем, защищенных от сбоев в формате, описанном в разделе "Выходные данные".

На выходе программа выдает файл spoj_score, с числом очков, полученным тестируемой программой в формате float, например 12.046564, а также служебную информацию с подробностями о работе программы в файле spoj_p_info, который доступен для просмотра только администраторам. Эта информация включает: число очков за каждый отдельный тест, число ошибок, внедренных в каждом конкретном тесте, используемую площадь в процентах от максимально возможной и время, затраченное на моделирование каждого теста.

Пример служебной информации для одного теста для числа итераций Monte Carlo $TN = 5000$:

Score for Test #2: 0.564800 [Err tests: 2670, Incorrect: 2176]
Error distribution: [0 — 2330] [1 — 1870] [2 — 662] [3 — 124]
[4 — 11] [5 — 3]
Area used: 354.0 from 1451.4 (24.39 %)
Time for test modeling: 0.003 sec

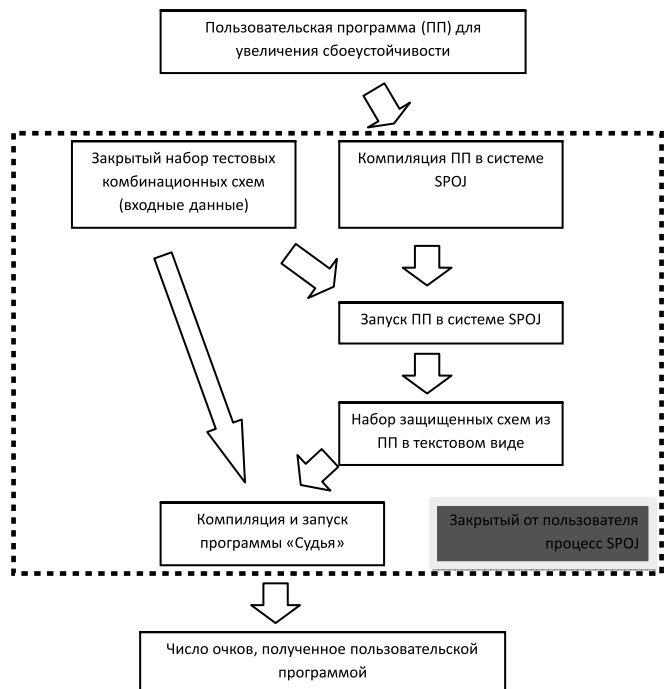


Рис. 3. Поток данных в системе SPOJ

Рейтинг участников для задачи в системе SPOJ

Место	Дата	Пользователь	Результат	Время	Память	Язык программирования
1	2015-02-18 11:49	Sergey	21,509933	12,49 с	10 Мбайт	C
2	2014-11-27 12:58	Flago	11,552267	32,98 с	24 Мбайт	PHP
3	2014-08-09 00:22	Robert Gerbicz	0,1058	2,36 с	11 Мбайт	C++4.3.2
4	2014-08-09 02:08	Mitch Schwartz	0,1058	0,1 с	1,9 Мбайт	BF

Последовательность работы программы "Судья" для каждого отдельного теста:

1. Чтение тестовой схемы.
2. Чтение схемы, сгенерированной тестируемой программой.
3. Проверка условия, что площадь тестируемой схемы не превышает требуемую.
4. Тестирование по методу Монте-Карло, при котором определяется, что схемы работают одинаково при случайных тестовых последовательностях на входах. В случае обнаружения ошибки программа "Судья" выдает сообщение об ошибке.
5. Моделирование неисправностей по методу Монте-Карло и подсчет значения COF.

Исходный код программы "Судья" доступен по ссылке [14]. Задача, оформленная с текстом для неспециалистов, доступна для решения по ссылкам [18, 19]. Решения принимаются более чем на 40 языках программирования. Поток данных в системе показан на рис. 3.

Ограничения в системе SPOJ

В процессе реализации системы тестирования мы столкнулись с некоторыми ограничениями. Процесс моделирования методом Монте-Карло оказался довольно затратным по времени. В итоге, чтобы программа "Судья" работала в приемлемые сроки, пришлось:

- снизить общее число тестов с планируемых 2000 до примерно 400;
- уменьшить число итераций Монте-Карло с 10 000 до 5000 для маленьких схем и до 3000 для больших схем;
- оптимизировать код программы "Судья".

После всех этих манипуляций удалось сократить время работы программы "Судья" с нескольких минут до примерно 50...60 с.

Сама система SPOJ накладывает ограничения (правда не строгие) на размер исходного кода. В данный момент он ограничен 50 Кбайт, а также на время выполнения тестируемой программы. Администраторы системы не рекомендуют ставить очень большие значения. Сейчас время выполнения ограничено 50 с.

Метод Монте-Карло является недетерминированным и зависит от генератора случайных чисел, поэтому число полученных очков, хотя и незначительно, но может варьироваться для схожих решений. Также существует вероятность, что тестируе-

мая схема, не эквивалентная исходной, пройдет тест на эквивалентность (но вероятность астрономически мала).

Одним из простейших решений задачи является вариант — вывести исходные схемы в качестве ответа. Выяснилось, что значение SCORE при этом получается достаточно большим по отношению к теоретическому максимуму, и разница между этим значением и реальными решениями получается не очень существенной. Поскольку нас интересуют только алгоритмы, в которых происходит улучшение надежных характеристик схемы, то было принято решение изменить формулу расчета очков (3) на следующую:

$$SCORE = \left(\sum_{i=0}^{TN} COF_i \right) - DEF,$$

где DEF — это суммарное значение COF для исходной схемы без защиты. В нашем случае DEF получилось равным 276. В случае если SCORE равен отрицательному числу, то такое решение получает 0 очков. После этого изменения все реальные решения задачи стали визуально сильно отличаться от стандартных решений (см. таблицу).

Чтобы облегчить авторам алгоритмов задачу, мы упростили формат входных данных, сократили число требуемых для обработки параметров задачи (уменьшили число стандартных ячеек и оставили площадь и вероятность отказа в виде цифровых коэффициентов), а также подготовили:

1. Программу-пример на языке C, которая читает входные данные и выдает схему без изменений на выход [15].
2. Набор тестов, сходный с набором тестов на сервере для локального тестирования алгоритмов [16].
3. Программа "Судья", которая используется на сервере для отладки своей программы с точки зрения эффективности решения [14].

Исследование выполнено за счет гранта Российского научного фонда (проект № 14-19-01036).

Список литературы

1. Niranjana S., Frenzel J. F. A comparison of fault-tolerant state machine architectures for space-borne electronics // Reliability, IEEE Transactions on. 1996. Vol. 45, N. 1. P. 109—113.
2. Choudhury M. R., Mohanram K. Reliability Analysis of Logic Circuits // IEEE transactions on computer-aided design of integrated circuits and systems. 2009. Vol. 28, N. 3.

3. **Huang C.** Robust Computing with Nano-scale Devices. Springer, 2010. 180 p.
4. **Han J., Leung E., Liu L., Lombardi F.** A Fault-Tolerant Technique using Quadded Logic and Quadded Transistors // *IEEE Transactions on VLSI Systems*. 2014. P. 1—5.
5. **Poolakkaparambil M., Mathew J., Jabir A.** Multiple Bit Error Tolerant Galois Field Architectures Over GF(2^m) // *Electronics*. 2012. N. 1. P. 3—22.
6. **Shubham C. A., Kolte M. T.** Fault Tolerant and Correction System Using Triple Modular Redundancy // *International Journal of Emerging Engineering Research and Technology*. 2014. Vol. 2. P. 187—191.
7. **Vial J., Bosio A., Girard P.** et al. Using TMR architectures for yield improvement // *Defect and Fault Tolerance of VLSI Systems*, 2008 in DFTVS'08. IEEE International Symposium on. IEEE. 2008. P. 7—15.
8. **Ebrahimi M., Miremadi S. G., Asadi H.** ScTMR: A scan chain-based error recovery technique for TMR systems in safety-critical applications // *Design, Automation & Test in Europe Conference & Exhibition (DATE)*. 2011. IEEE, 2011. P. 1—4.
9. **Онлайн** система оценки решений SPOJ. URL: <http://spoj.com>
10. **Описание** системы SPOJ. URL: <http://www.spoi.com/info/>
11. **Bryan D.** The ISCAS'85 benchmark circuits and netlist format. North Carolina State University. 1985. P. 4.

12. **Соболь И. М.** Численные методы Монте-Карло. М.: Наука, 1973. 312 с.
13. **Стемпковский А. Л., Тельпухов Д. В., Соловьев Р. А., Соловьев А. Н., Мячиков М. В.** Моделирование возникновения неисправностей для оценки надежностных характеристик логических схем // *Информационные технологии*. 2014. № 11. С. 30—36.
14. **Исходный** код программы "судья". URL: <http://vscscripts.ru/docs/zrely/Judge.zip>
15. **Программа** пример на языке Си. URL: <http://vscscripts.ru/docs/zrely/prog.cpp>
16. **Набор** тестов для локального тестирования алгоритмов. URL: <http://vscscripts.ru/docs/zrely/input.zip>
17. **Lynch J. D.** Stochastic fault simulation of triple-modular redundant asynchronous pipeline circuits: Thesis (Ph. D.). — Oregon Health & Science University. 2009.
18. **Русская** версия задачи. URL: <http://www.spoj.com/ZELARCH/problems/ZRELY/>
19. **Английская** версия задачи. URL: <http://www.spoj.com/problems/ZRELY1/>
20. **Описание** типа ошибки. URL: <http://radhome.gsfc.nasa.gov/radhome/see.htm>

A. L. Stempkovskiy, Academician, Director, **D. V. Telpukhov**, Ph. D., Head of the Department, **R. A. Solovyev**, Ph. D., Researcher of the Department, **V. S. Rukhlov**, Junior Researcher, IPPM RAS

Test System for Comparing Algorithms which Increases the Fault Tolerance of Combinational Circuits

Description of the system consisting of a large set of tests and a judge program is proposed. The system is used for performance evaluation of arbitrary algorithms of protection of logic and arithmetic circuits without feedback. While developing the system, experience in preparing tasks for sports programming competitions was used. The paper presents the details of the closed system of tests development and of the scoring methodology. Details of the judge program work are revealed. The developed test system is freely available online from the SPOJ system for all reliability professionals. The system ranks the algorithms and arranges them in a list according to their total score. The more points the algorithm received, the more reliable circuits based on the initial version it generated.

Keywords: reliability, redundancy, logic circuits, sports programming, testing

References

1. **Niranjan S., Frenzel J. F.** A comparison of fault-tolerant state machine architectures for space-borne electronics, *Reliability, IEEE Transactions on*, 1996, vol. 45, no. 1, pp. 109—113.
2. **Choudhury M. R., Mohanram K.** Reliability Analysis of Logic Circuits, *IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEM*, 2009, vol. 28, no. 3.
3. **Huang C.** Robust Computing with Nano-scale Devices, Springer, 2010, 180 p.
4. **Han J., Leung E., Liu L., Lombardi F.** A Fault-Tolerant Technique using Quadded Logic and Quadded Transistors. *IEEE Transactions on VLSI Systems*, 2014, pp. 1—5.
5. **Poolakkaparambil M., Mathew J., Jabir A.** Multiple Bit Error Tolerant Galois Field Architectures Over GF(2^m), *Electronics*, 2012, no. 1, pp. 3—22.
6. **Shubham C. A., Kolte M. T.** Fault Tolerant and Correction System Using Triple Modular Redundancy, *International Journal of Emerging Engineering Research and Technology*, 2014, vol. 2, pp. 187—191.
7. **Vial J., Bosio A., Girard P.** et al. Using TMR architectures for yield improvement. *Defect and Fault Tolerance of VLSI Systems*, 2008, DFTVS'08. IEEE International Symposium on. IEEE, 2008, pp. 7—15.
8. **Ebrahimi M., Miremadi S. G., Asadi H.** ScTMR: A scan chain-based error recovery technique for TMR systems in safety-critical

- tical applications. *Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2011, IEEE, 2011, pp. 1—4.
9. **http://spoj.com**
10. **http://www.spoj.com/info/**
11. **Bryan D.** The ISCAS'85 benchmark circuits and netlist format. North Carolina State University. 1985.
12. **Sobol' I. M.** Chislennyye metody Monte-Karlo. Moscow: Nauka, 1973. 312 p.
13. **Stempkovskiy A. L., Tel'puhov D. V., Solov'ev R. A., Solov'ev A. N., Mjachikov M. V.** Modelirovanie vozniknoveniya neispravnostej dlja ocenki nadezhnostnyh harakteristik logicheskikh shem, *Informacionnye tehnologii*, 2014, no. 11, pp. 30—36.
14. **http://vscscripts.ru/docs/zrely/Judge.zip**
15. **http://vscscripts.ru/docs/zrely/prog.cpp**
16. **http://vscscripts.ru/docs/zrely/input.zip**
17. **Lynch J. D.** Stochastic fault simulation of triple-modular redundant asynchronous pipeline circuits: Thesis (Ph. D.). Oregon Health & Science University, 2009.
18. **http://www.spoj.com/ZELARCH/problems/ZRELY/** — russkaja versija zadachi
19. **http://www.spoj.com/problems/ZRELY1/** — anglijskaja versija zadachi
20. **http://radhome.gsfc.nasa.gov/radhome/see.htm**

К. О. Петросянц^{1, 2}, д-р техн. наук, проф., **И. А. Харитонов**¹, канд. техн. наук, проф.,
М. В. Кожухов¹, аспирант, **Л. М. Самбурский**^{1, 2}, канд. техн. наук, доц., e-mail: kpetrosyants@hse.ru,

¹ Национальный исследовательский университет "Высшая школа экономики"

(Московский институт электроники и математики), г. Москва

² Институт проблем проектирования в микроэлектронике РАН, г. Москва

Схемотехнические SPICE-модели биполярных и МОП-транзисторов для автоматизации проектирования радиационно стойких БИС

Возможности коммерческих SPICE-симуляторов электронных схем расширены в новую область — расчет схем космической электроники. С этой целью в библиотеки моделей коммерческих версий программы SPICE включен набор моделей биполярных и МОП-транзисторов, учитывающих воздействие различных видов радиации (гамма-лучей, протонов, нейтронов, электронов, одиночных заряженных частиц). Приведены примеры расчета биполярных и МОП-интегральных схем с учетом требований радиационной стойкости.

Ключевые слова: биполярные транзисторы (БТ), гетеропереходные биполярные транзисторы (ГБТ), МОП-транзисторы, ионизирующее излучение, SPICE-модели, системы автоматизированного проектирования (САПР), схемотехническое моделирование, радиационная стойкость

Введение

Современный рынок экстремальной электроники включает в себя приборы, микросхемы и системы на кристалле и в корпусе, которые работают в аппаратуре, подверженной воздействию внешних факторов, в частности радиации. Это спутниковые системы, включенные в глобальную коммуникационную инфраструктуру, космические системы обработки и передачи изображений для климатических, топографических работ, системы обнаружения и исследования элементарных частиц, системы обработки медицинских изображений, системы контроля и управления объектов атомной энергетики, системы вооружения и военной техники и др. Очевидно, что этот рынок в будущем будет только расширяться.

В большинстве случаев обычные электронные приборы, используемые в составе экстремальных систем, быстро деградируют и выходят из строя в результате воздействия радиационного излучения. Поэтому к надежности электронной компонентной базы предъявляются чрезвычайно высокие требования, которые обеспечиваются за счет использования особо надежных конструктивно-технологических решений, специальных средств защиты и специальных, более "жестких" методов проектирования приборов и схем с повышенной надежностью. С этой целью разработчиками используются системы радиационно стойкого проектирования "hardening-by-design".

Коммерческие программы, построенные на базе платформы SPICE, широко используются в системах автоматизированного схемотехнического проектирования [1—4]. Однако возможности этих программ при проектировании схем, работающих в экстремальных условиях эксплуатации, существенно

ограничены, поскольку заложенные в них модели не учитывают различные виды внешних воздействий: радиацию, температуру, электромагнитное излучение и др.

В настоящей работе сделана попытка снять существующие ограничения в части учета радиационных эффектов. Предложен набор компактных схемотехнических моделей биполярных и МОП-транзисторов, учитывающих различные виды радиационных воздействий. Модели встроены в библиотеки моделей коммерческих SPICE-подобных программ, что позволило распространить возможности этих программ на задачи проектирования радиационно стойких интегральных схем (ИС), удовлетворяющих жестким требованиям по показателям надежности, сбоеустойчивости и срока службы в составе бортовой аппаратуры.

1. Методология разработки SPICE-моделей компонентов, учитывающих радиационные эффекты

Анализ отечественных и зарубежных работ по созданию SPICE-моделей биполярных транзисторов (БТ) и МОП-транзисторов (МОПТ), учитывающих радиационные эффекты [5—10], показал, что существующие модели, во-первых, узкоспециализированы и направлены на решение частных задач, во-вторых, учитывают, как правило, только один вид радиационного воздействия, в-третьих, эквивалентные схемы и уравнения, вводимые для учета радиационных эффектов, отличаются большим разнообразием и содержат большое число формальных коэффициентов, которые или трудно определяются из результатов традиционных измерений, или требуют специальных дополнительных измерений. Для работы с такими моделями, как правило,

требуется специальная подготовка пользователя, кроме того, большое разнообразие частных моделей со своим математическим описанием, системой параметров и специфической методикой экстракции параметров создает серьезные трудности при включении этих моделей в библиотеки моделей коммерческих или промышленных версий программы схемотехнического проектирования SPICE.

В настоящей работе перечисленные выше проблемы решены за счет создания одной унифицированной модели для всех разновидностей БТ и унифицированной модели для разновидностей МОПТ. При этом каждая унифицированная модель для БТ и для МОПТ учитывает влияние радиационных эффектов всех видов: нейтронов, протонов, электронов, гамма-квантов, импульсного излучения и одиночных заряженных частиц (ОЯЧ). Каждая унифицированная модель состоит из центрального ядра, которое включает в себя существующие стандартные SPICE-модели БТ или МОПТ, и дополнительной электрической подсхемы (отдельно для БТ и отдельно для МОПТ), учитывающей все виды радиационного воздействия и подключаемой к центральному ядру. Дополнительные подсхемы состоят из стандартных элементов электрических цепей, параметры этих элементов определяются по стандартным методам измерений характеристик на тестовых приборах, доступным и понятным разработчикам ИС. Такая организация моделей резко упрощает процесс схемотехнического проектирования ИС с учетом радиационных эффектов.

Основные этапы разработки SPICE-моделей электронной компонентной базы (ЭКБ) с учетом радиационных эффектов представлены на рис. 1. Для создания моделей используется комбинация двух универсальных методов: 1) макромодельный подход, заключающийся в подключении схемных элементов к стандартной SPICE-модели полупроводникового прибора; 2) введение в уравнения схе-

мотехнической модели дополнительных математических выражений для описания радиационно-зависимых параметров.

Для космической электроники наиболее важны следующие виды радиационного воздействия: полная поглощенная доза гамма-излучения, протоны, нейтроны, электроны, одиночные частицы. Физические эффекты, вызванные воздействием этих видов, включаются в стандартные модели биполярных и МОП-транзисторов, изготовленных по различным технологиям: обычные Si БТ, GaAs гетеропереходные биполярные транзисторы (ГБТ), SiGe ГБТ, обычные МОПТ, МОПТ со структурой "кремний на изоляторе/сапфире" (КНИ/КНС), МОП-транзисторы с двойной диффузией (ДМОПТ), полевые транзисторы с управляющим *p-n*-переходом (JFET), MESFET и HEMT на основе GaAs и др.

Все перечисленные выше разновидности биполярных и МОП-транзисторов, изготовленные на основе кремний-германия (SiGe) и арсенида галлия (GaAs), обладают повышенными показателями по радиационной стойкости. По оценкам специалистов Американского космического агентства NASA, Европейского космического агентства, Роскосмоса и Росатома, они включены в список перспективных компонентов для производства радиационно стойких схем для космической и другой специальной аппаратуры [16, 17].

2. Унифицированная SPICE-модель биполярного транзистора с учетом радиационных эффектов

При выборе полупроводникового прибора для использования в электронной аппаратуре специального назначения необходимым этапом является оценка воздействия различных видов радиации на степень деградации его электрических параметров.

В качестве примера на рис. 2 для биполярного *n-p-n*-транзистора интегральной схемы операцион-



Рис. 1. Схема стратегии разработки SPICE-моделей компонентов с учетом радиационных эффектов

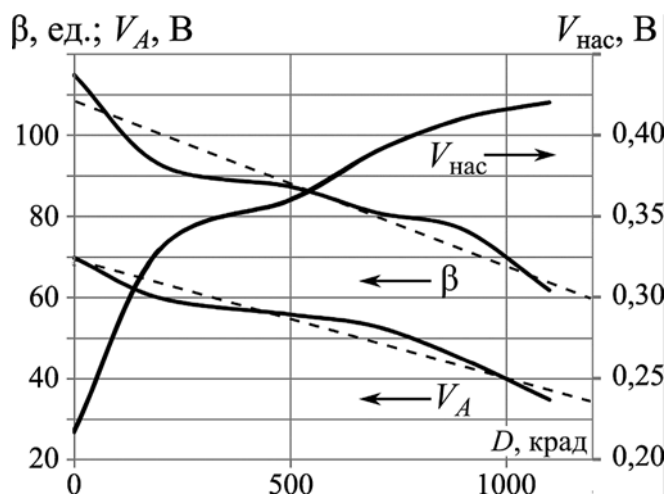


Рис. 2. Зависимости напряжения Эрли V_A , напряжения насыщения $V_{нас}$ и коэффициента усиления по току β от поглощенной дозы D электронного излучения для интегрального биполярного *n-p-n*-транзистора

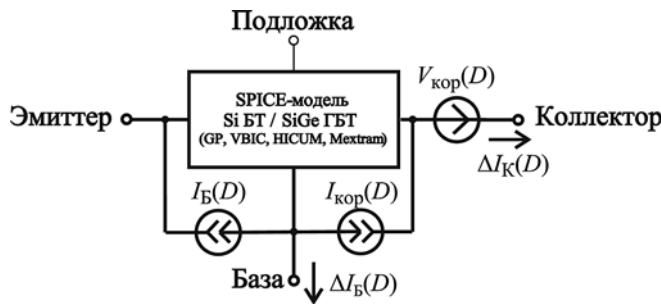


Рис. 3. Компактная SPICE-модель биполярного транзистора, учитывающая радиационные эффекты

ного усилителя [18] приведены зависимости основных параметров: напряжения Эрли V_A , напряжения насыщения $V_{нас}$ и коэффициента усиления по току β от поглощенной дозы электронного излучения.

Видно, что все важные для БТ параметры β , V_A , $V_{нас}$ существенным образом деградируют.

Для каждого вида облучения деградация характеристик БТ должна с достаточной точностью описываться его SPICE-моделью.

В нашем случае при использовании макромоделного подхода транзистор описывается одной из известных стандартных SPICE-моделей БТ. Эта модель является ядром, к которому подключена под-схема, содержащая дополнительные схемные эле-менты, учитывающие влияние разных видов ра-диационных излучений (рис. 3).

В макромодели для учета влияния воздействия разных видов статических излучений (нейтроны, протоны, электроны, гамма- и рентгеновские лучи и др.) используются следующие схемные элементы: источник радиационно-индуцированного тока базы $I_B(D)$, описывающий спад усиления в области малых токов после облучения; источник тока $I_{kop}(D)$,

учитывающий уменьшение тока насыщения кол-лектора после полученной дозы D ; источник на-пряжения $V_{kop}(D)$, учитывающий эффект увеличе-ния напряжения $V_{нас}$ в результате облучения.

Дополнительные схемные элементы SPICE мак-ромодели Si БТ или SiGe ГБТ для учета влияния различных видов радиации описываются следую-щими уравнениями [5, 19]:

$$I_B(D) = I_{sd}(1 + K_d D) \left(\exp \left[\frac{U_{БЭ}}{n_{ed}} \right] - 1 \right) + I_{ss \max} (1 - \exp[-K_s D]) \left(\exp \left[\frac{U_{БЭ}}{n_{es}} \right] - 1 \right); \quad (1)$$

$$I_{kop}(D) = I_d(D) \exp \left[\frac{U_{КЭ}}{n_{es}} \right]; \quad I_d(D) = v + g \exp[-hD]; \quad (2)$$

$$V_{kop}(D) = \sqrt{CD} \left(1 - \exp \left[\frac{U_{КЭ} B}{n_{es}} \right] \right) = \sqrt{\frac{aD}{1 + f \exp S}} \left(1 - \exp \left[\frac{U_{КЭ} B}{n_{es}} \right] \right), \quad (3)$$

где I_{sd} , $I_{ss \max}$, K_d , K_s , n_{ed} , n_{es} , v , g , h , a , f , S , B — под-гоночные коэффициенты; D — поглощенная доза излучения или поток частиц; $U_{БЭ}$ — напряжение ба-за-эмиттер, $U_{КЭ}$ — напряжение коллектор-эмиттер.

Уравнения (1)–(3) являются общими для всех видов радиации. Для протонов, электронов, гамма- и рентгеновских лучей в качестве влияющего фак-тора используется доза D ; для учета влияния ней-тронного излучения необходимо в уравнениях (1)–(3) заменить дозу D на значения интегрально-го потока нейтронов Φ .

Разработанная макромодел в отличие от других известных моделей [5–7] обладает большей гибко-

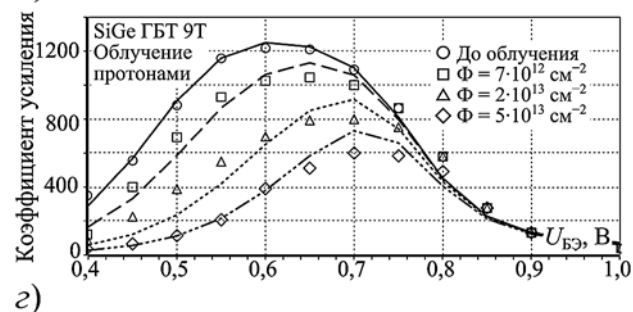
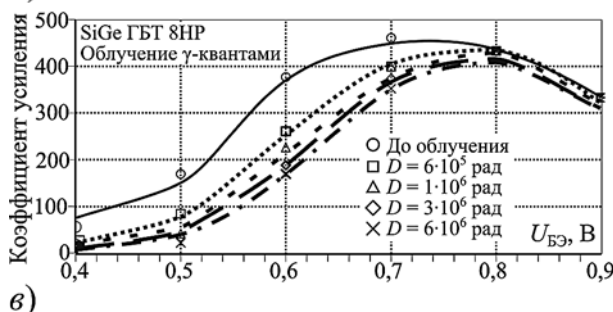
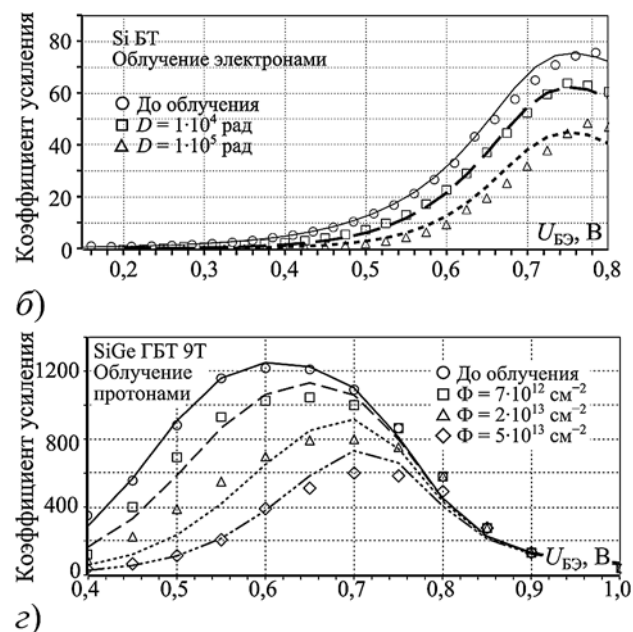
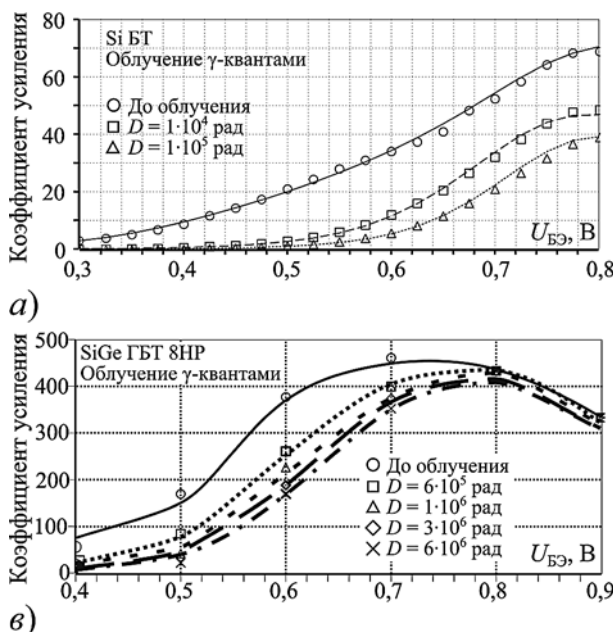


Рис. 4. Сравнение смоделированного и измеренного коэффициента усиления биполярных транзисторов до и после облучения: а — Si БТ, γ -излучение; б — Si БТ, электроны; в — SiGe ГБТ 8HP, γ -излучение; г — SiGe ГБТ 9T, протоны

стью при описании изменения тока базы, что позволяет описывать зависимости любой сложности с приемлемой точностью. Также разработанная макромодель учитывает изменение напряжения насыщения и напряжения пробоя на выходных характеристиках после воздействия радиации. В аналогичных работах учет влияния различных видов проникающей радиации на выходные характеристики либо учитывается частично, либо не учитывается совсем.

Ниже приведены примеры использования разработанной SPICE-макромодели для расчета параметров Si БТ и SiGe ГБТ, подвергнутых воздействию нейтронного, протонного, электронного и гамма-излучений. Для моделирования были выбраны транзисторы:

- интегральный *n-p-n* Si БТ, изготовленный по отечественной технологии аналоговых ИС с параметрами: коэффициент усиления по току $\beta = 70$, граничная частота $f_{гр} = 5,1$ ГГц и максимальная частота $f_{max} = 1,9$ ГГц;
- SiGe ГБТ компании IBM, выполненный по технологии 8HP с проектной нормой 120 нм с параметрами: $\beta = 450$, $f_{гр} = 200$ ГГц, $f_{max} = 100$ ГГц;
- SiGe ГБТ компании IBM, выполненный по технологии 9T с проектной нормой 120 нм с параметрами: $\beta = 1250$, $f_{гр} = 350$ ГГц, $f_{max} = 250$ ГГц.

На рис. 4 приведены результаты измерения и моделирования с помощью разработанной модели коэффициента усиления вышеупомянутых транзисторов до и после воздействия различных видов радиации. Расхождения между измеренными и смоделированными характеристиками Si БТ и SiGe ГБТ не превышают 10–15 %, что является удовлетворительным для прогнозирования параметров и характеристик БТ, подвергнутых воздействию радиации.

Пример использования модели БТ для схемотехнического моделирования приведен на рис. 5, где для различных уровней потока нейтронов показаны результаты измерения и моделирования АЧХ (относительного изменения коэффициента усиления по напряжению K_u) и переходной характеристики выходного напряжения операционного усилителя, реализованного на базе биполярного базового матричного кристалла (БМК) KB1451HN2-4. Погрешность моделирования электрических характеристик облученных схем с помощью макромодели Si БТ/SiGe ГБТ не превышает 25 %, что является удовлетворительным для практики показателем при моделировании динамических характеристик.

3. Унифицированная SPICE-модель МОП-транзистора с учетом радиационных эффектов

Модель рассмотрим на примере КМОП-транзисторов со структурой "кремний на изоляторе/сапфире" (КНИ/КНС КМОПТ), которые обладают повышенной радиационной стойкостью и являются перспективными электронными компонентами для проектирования радиационно стойких ИС.

Разработаны две макромодели BSIMSOI-RAD [14, 20, 21] и EKV-RAD [22, 23] для описания субмикронных КНИ/КНС КМОП-транзисторов с учетом эффектов, возникающих при воздействии статического и импульсного радиационного потока, отдельных ядерных частиц (ОЯЧ). Эквивалентные схемы макромодели BSIMSOI-RAD и EKV-RAD представлены на рис. 6.

Основной элемент макромодели — верхний транзистор $M_{верх}$ (описывается стандартной моделью BSIMSOI v3.2 или EKV v2.64 с радиационно-зависимыми параметрами, см. рис. 6, а) — отражает по-

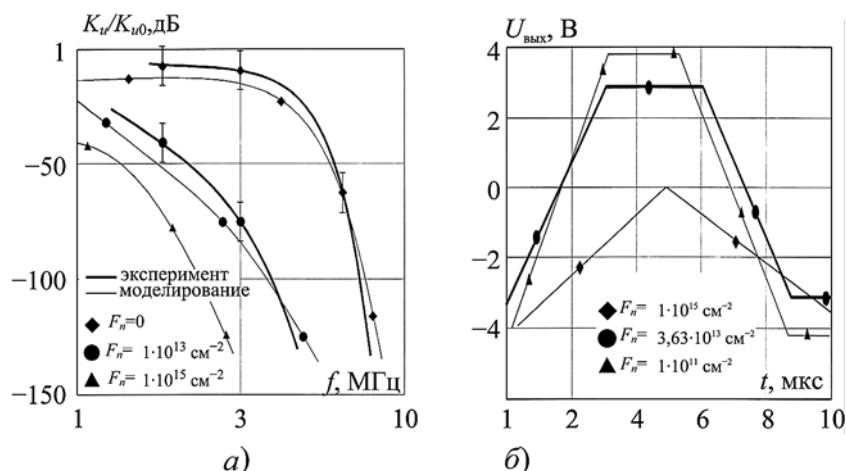


Рис. 5. Сравнение смоделированной и измеренной АЧХ (а) и переходной характеристики (б) операционного усилителя для разных значений потока нейтронов F_n

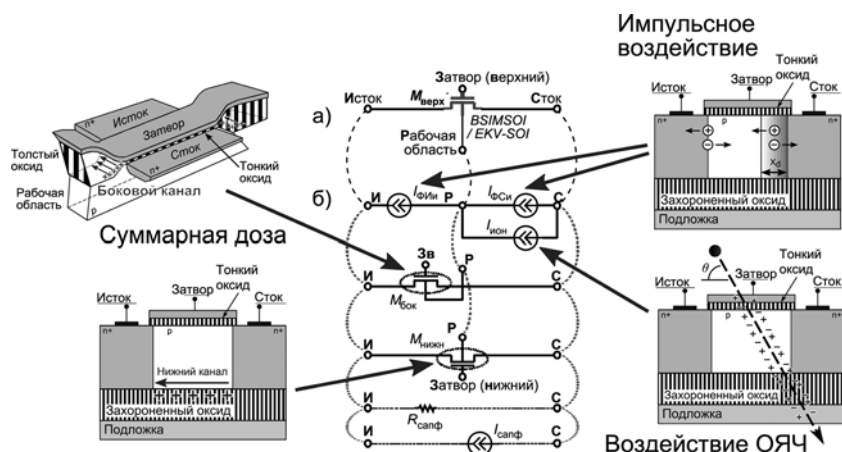


Рис. 6. Эквивалентная схема обобщенной макромодели BSIMSOI-RAD и EKV-RAD для КНИ/КНС МОПТ: а — базовый МОПТ $M_{верх}$ с радиационно-зависимыми параметрами; б — под-схема для учета статических и динамических токов утечки

ведение верхнего МОП-интерфейса. Для учета токов утечки, возникающих при стационарном радиационном воздействии, модель дополняется паразитными нижним $M_{\text{нижн}}$ и боковым $M_{\text{бок}}$ транзисторами; при импульсном воздействии — источниками фототока стокового и истокового p - n -переходов $I_{\text{ФСн}}(t)$ и $I_{\text{ФИи}}(t)$ и сопротивлением сапфира $R_{\text{сапф}}(t)$; при воздействии ОЯЧ — источником тока $I_{\text{ион}}(t)$ (см. рис. 6, б).

Базовый элемент макромодели EKV-RAD — стандартная модель EKV [25] — была изначально разработана для описания МОПТ на объемном кремнии. Эта модель распространена на случай МОПТ с диэлектрической КНИ/КНС-подложкой.

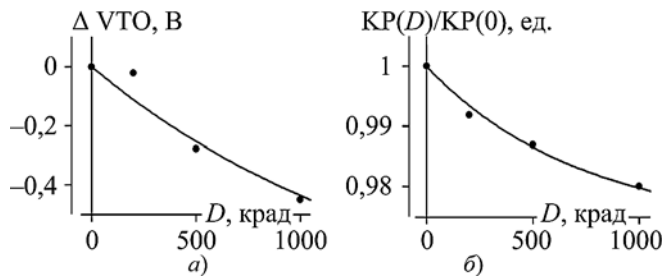


Рис. 7. Типовой вид аппроксимирующих функций для зависимости параметров модели от дозы: а — для пороговых напряжений; б — для подвижности

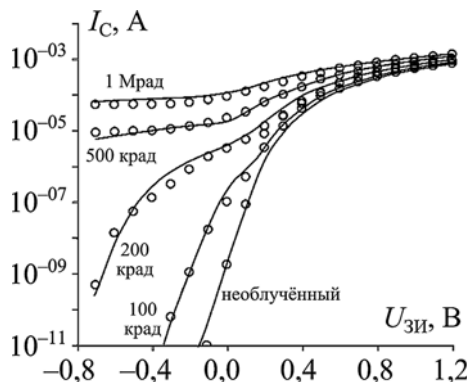


Рис. 8. Измеренные (символы) и смоделированные (линии) сток-затворные ВАХ КНИ МОПТ с размерами затвора $W/L = 8/0,13$ мкм до и после γ -облучения с дозой до 1 Мрад, I_C — ток стока, $U_{\text{зи}}$ — напряжение затвор—исток

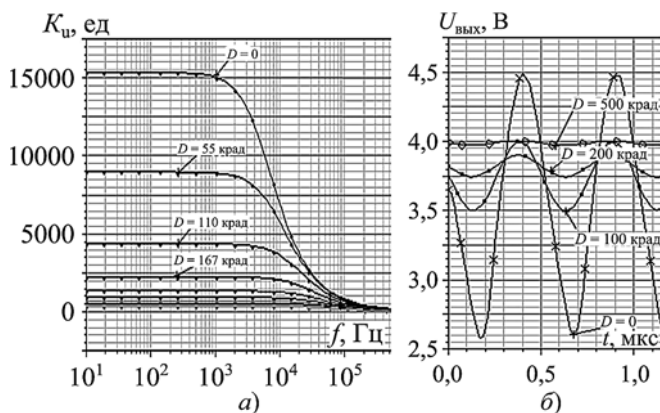


Рис. 9. Результаты моделирования частотных (а) и переходных (б) характеристик операционного усилителя до и после γ -облучения с дозой до 1 Мрад

При воздействии стационарного излучения параметры модели основного верхнего транзистора $M_{\text{верх}}$ и остальных элементов меняются с увеличением дозы излучения. Зависимыми параметрами верхнего и паразитных транзисторов являются пороговое напряжение с коэффициентами (V_{TH0} , V_{TO} и др.), крутизна с коэффициентами (U_0 , U_O , K_P , E_0 , U_A и др.), предпороговый наклон с коэффициентами (CIT , $GAMMA$ и др.). Типичный вид дозовой зависимости параметров приведен на рис. 7. Радиационные изменения вышеперечисленных параметров, в том числе приведенных на рис. 7, описываются выражениями вида

$$a_{1i}(1 - \exp[-a_{2i}D]), i = 1, \dots, N, \quad (4)$$

где a_1 , a_2 — подгоночные коэффициенты, которые составляют набор радиационных параметров макромодели; N — число радиационно-зависимых параметров.

На рис. 8 для примера приведено сравнение результатов измерения и моделирования сток-затворных ВАХ КНИ МОПТ с размерами затвора $W/L = 8/0,13$ мкм. Погрешность совпадения 10...15 % во всем диапазоне изменения дозы и управляющих напряжений.

В качестве примера использования разработанных моделей КНИ/КНС МОПТ в схемотехнике БИС приведены результаты моделирования АЧХ (коэффициент усиления по напряжению K_u) и переходной характеристики выходного напряжения интегрального ОУ с полным размахом входного напряжения ("rail-to-rail"), содержащего 45 КНИ МОП-транзисторов, изготовленного по отечественной технологии, с учетом поглощенной дозы гамма-излучения в диапазоне 0...1 Мрад (рис. 9).

В качестве примера расчета фрагмента цифровой БИС на рис. 10, а приведены результаты моделирования ячейки памяти статического ОЗУ с учетом попадания тяжелого иона в транзистор М1 ячейки в момент времени $t = 2,5$ мкс (этап хранения). Рассчитывались два случая с различной энергией воздействующего иона, выраженной параметром ЛПЭ¹: 1) ЛПЭ иона 18 МэВ · см²/мг (без сбоя, см. рис. 10, б, з); 2) ЛПЭ иона 21 МэВ · см²/мг (со сбоем, рис. 10, в). Сбой ячейки зависит от величины заряда, накопленного в пораженном транзисторе. Результаты моделирования корректно отражают поведение ячейки, что дает возможность оценить ее сбоеустойчивость.

Заключение

Разработана библиотека SPICE-моделей биполярных и МОП-транзисторов, изготовленных по перспективным микроэлектронным технологиям, с учетом воздействия радиационных эффектов раз-

¹ ЛПЭ, линейная передача энергии, — это отношение количества энергии, теряемого ионизирующей частицей вследствие столкновений при пролете сквозь вещество, к длине пройденного пути и плотности вещества.

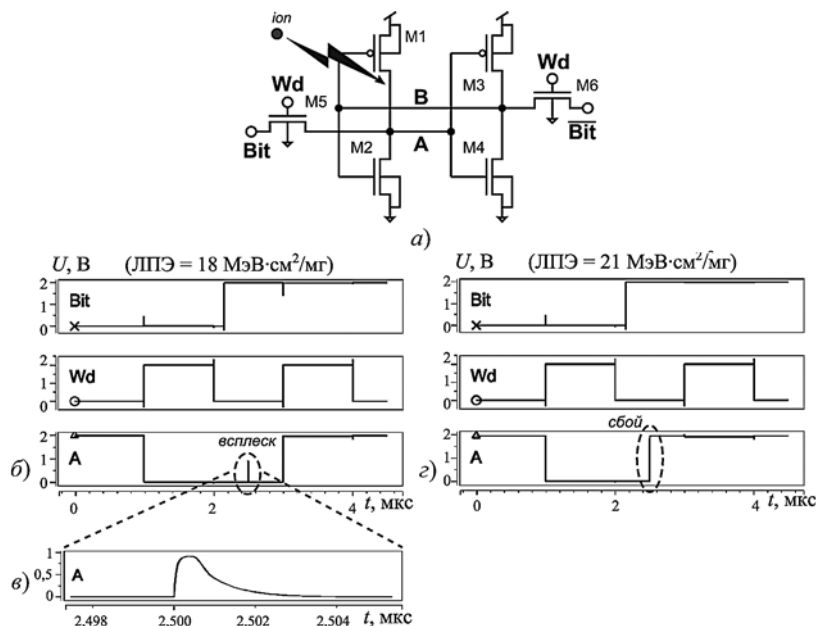


Рис. 10. Результаты моделирования КМОП-ячейки памяти (а) при воздействии ОЯЧ на транзистор МРА:
б — без сбоя; в — скачок напряжения $V(A)$ для случая б; г — со сбоем

личного типа (гамма-лучей, протонов, нейтронов, электронов, одиночных частиц). Библиотека включена в коммерческие версии программы SPICE [1–4] и использовалась в работах по схемотехническому проектированию аналоговых и цифровых ИС и БИС, разрабатываемых отечественной промышленностью, в частности, для аналоговых узлов КНС КМОП ИС с повышенной радиационной стойкостью [22–26], биполярных схем операционных усилителей для приемников и передатчиков систем связи [18], КНИ КМОП БИС, построенных на базе семейства радиационно стойких цифроаналоговых БМК, для микромеханических систем [24] и др.

Таким образом, возможности коммерческих версий программы SPICE расширены в новую область — расчет схем космической и другой специальной электроники.

Статья подготовлена в ходе проведения исследования № 15-01-0165 в рамках Программы "Научный фонд Национального исследовательского университета "Высшая школа экономики" (НИУ ВШЭ)" в 2015 г. и гранта РФФИ № 14-29-09145.

Список литературы

1. Spectre Circuit Simulator, 2013 Cadence Design Systems, Inc.
2. Using SystemVision. Software version 5.7, December 2010, Mentor Graphics Corp.
3. HSPICE, The Gold Standard for Accurate Circuit Simulation, 2010 Synopsys, Inc.
4. SmartSpice, Release Notes, Silvaco Inc. 2014.
5. Van Uffelen M., Geboers S., Leroux P., Berghmans F. SPICE Modeling of a Discrete COTS SiGe HBT for Digital Applications up to MGy Dose Levels // IEEE Trans. on Nuclear Science. 2006. Vol. 53, N. 4. P. 1945–1949.
6. Leroux P., De Cock W., Van Uffelen M., Steyaert M. Modeling, Design, Assessment of 0.4 μm SiGe Bipolar VCSEL Driver IC Under γ -Radiation // IEEE Trans. on Nuclear Science. 2009. Vol. 56, N. 4. P. 1920–1925.

7. Deng Y., Fjeldly T., Ytterdal T., Shur M. S. SPICE modeling of neutron displacement damage and annealing effects in bipolar junction transistors // IEEE Trans. on Electron Devices. 2003. Vol. 50, N. 6. P. 1873–1877.
8. Gorbunov M. S., Danilov I. A., Zebrev G. I., Osipenko P. N. Verilog-A Modeling of Radiation-Induced Mismatch Enhancement // IEEE Trans. on Nuclear Science. 2011. Vol. 58, N. 3. P. 785–792.
9. Alvarado J., Boufouss E., Kilchytska V., Handre D. Compact model for single event transients and total dose effects at high temperatures for partially depleted SOI MOSFETs // Microelectronics Reliability. 2010. Vol. 50. P. 1852–1856.
10. Bu Jianhui, Bi Jinshun, Liu Mengxin, Han Zhengsheng. A total dose radiation model for deep sub-micron PDSOI NMOS // Journal of Semiconductors. 2011. Vol. 32, N. 1. P. 014002-1–014002-3.
11. Petrosyants K., Kozhukhov M. SPICE Model Parameters Extraction Taking into Account the Ionizing Radiation Effects // Proc. of IEEE East-West Design and Test Symposium (EWDTS). 2013. P. 304–307.
12. Petrosyants K. O., Kozhukhov M. V. SiGe HBT Performance Modeling after Proton Radiation Exposure // Proc. of IEEE East-West Design and Test Symposium (EWDTS). 2012. P. 274–277.
13. Petrosyants K. O., Vologdin E. N., Kozhukhov M. V. Si BJT and SiGe HBT Performance Modelling after Neutron Radiation Exposure // Proc. of IEEE East-West Design and Test Symposium (EWDTS). 2011. P. 267–270.
14. Petrosyants K. O., Sambursky L. M., Kharitonov I. A., Yatmanov A. P. SOI/SOS MOSFET compact macromodel taking into account radiation effects // Russian Microelectronics. 2011. Vol. 40, Is. 7. P. 457–462.
15. Petrosyants K. O., Kharitonov I. A., Popov D. A. Coupled TCAD-SPICE simulation of parasitic BJT effect on SOI CMOS SRAM SEU // Proc. of IEEE East-West Design and Test Symposium (EWDTS). 2013. P. 312–315.
16. Label K., Barth J. An Insider's Guide to Designing Spacecraft Systems and Instruments for Operation in the Natural Space Radiation Environment // Proc. of GSFC Systems Engineering Seminar, 2001. P. 562.
17. Данилин Н., Шипилов К. Особенности применения новой электронно-компонентной базы в космических разработках // Вестник электроники. 2012. № 2 (40). С. 42–45.
18. Прокопенко Н. Н., Будяков А. С., Савченко Е. М. Операционные усилители с обобщенной токовой обратной связью // Проблемы разработки перспективных микро- и наноэлектронных систем — 2008. Сборник научных трудов / Под общ. ред. А. Л. Стемпковского. М.: ИППМ РАН, 2008. С. 330–333.
19. Petrosjanc K. O., Kharitonov I. A. VLSI device parameters extraction for radiation hardness modeling with SPICE // Proc. of IEEE International Conference on Microelectronic Test Structures. 1993. P. 9–14.
20. Petrosjanc K. O., Sambursky L. M., Kharitonov I. A. A Compact SOI/SOS MOSFET Macromodel Accounting for Radiation Effects // Proc. of IEEE East-West Design and Test Symposium (EWDTS). 2007. P. 360–364.
21. Petrosjanc K. O., Kharitonov I. A., Orekhov E. V., Sambursky L. M., Yatmanov A. P. Simulation of Radiation Effects in SOI CMOS Circuits with BSIM-SOI-RAD macromodel // Proc. of IEEE East-West Design and Test Symposium (EWDTS). 2009. P. 243–246.
22. Petrosyants K. O., Kharitonov I. A., Sambursky L. M., Bogatyrev V. N., Povarnitcyna Z. M., Drozdenco E. S. Simulation of total dose influence on analog-digital SOI/SOS CMOS circuits with EKV-RAD macromodel // Proc. of IEEE East-West Design and Test Symposium (EWDTS). 2012. P. 60–65.
23. Petrosyants K. O., Kharitonov I. A., Sambursky L. M. Hardware-Software Subsystem for MOSFETs Characteristic Measurement and Parameter Extraction with Account for Radiation Effects // Advanced Materials Research. 2013. Vol. 718–720. P. 750–755.
24. Petrosjanc K. O., Adonin A. S., Kharitonov I. A., Sicheva M. V. SOI device parameter investigation and extraction for VLSI radiation hardness modeling with SPICE // Proc. of IEEE International Conference on Microelectronic Test Structures. 1994. P. 126–129.
25. Enz C. C., Krummenacher P., Vittoz E. A. An analytical MOS transistor model valid in all regions of operation and dedicated to low-voltage and low-current applications // Analog Integrated Circuits and Signal Processing. 1995. Vol. 8, N. 1. P. 83–114.

26. Петросянц К. О., Богатырев В. Н., Гоманилова Н. Б., Поварнищина З. М., Щекин А. А., Харитонов И. А., Самбурский Л. М. Проектирование радиационно-стойкого прецизионного усилителя на базе КНС КМОП технологии // Сб. науч. тр. "Электроника, микро- и нанoeлектроника" / Под ред. В. Я. Стенина. М.: МИФИ, 2013. С. 196—202.

27. Петросянц К. О., Харитонов И. А., Самбурский Л. М., Орехов Е. В., Чёрный А. И., Поварнищина З. М., Богатырев В. Н., Прокофьев Г. В. Исследование характеристик элементной базы аналоговых КНИ КМОП схем, изготовленных по технологии XFAB, с учетом суммарной поглощенной дозы // Сб. науч. тр. "Электроника, микро- и нанoeлектроника" / Под ред. В. Я. Стенина. М.: МИФИ, 2009. С. 57—66.

K. O. Petrosyants, Professor, I. A. Kharitonov, Professor,
M. V. Kozhukhov, Graduate Student, L. M. Sambursky, Associate Professor, e-mail: kpetrosyants@hse.ru

Compact SPICE Models of Bipolar Transistors and MOSFETs for Automation of Radiation-Hardened VLSIs Design

Possibilities of commercial SPICE simulators of integrated circuits are expanded into a new field: design of ICs for space electronics. For this purpose, model libraries of commercial SPICE versions are supplemented with a set of models of bipolar transistors and MOSFETs accounting for radiation influence of various types (gamma rays, protons, neutrons, electrons, single nuclear particles). Examples of bipolar and CMOS ICs design are presented with due account for radiation hardness.

Keywords: bipolar transistors, heterojunction bipolar transistors, MOSFETs, ionizing radiation, SPICE models, computer-aided design, circuit design, radiation hardness

References

1. Spectre Circuit Simulator, 2013 Cadence Design Systems, Inc.
2. Using SystemVision. Software version 5.7, December 2010, Mentor Graphics Corp.
3. HSPICE, The Gold Standard for Accurate Circuit Simulation, 2010 Synopsys, Inc.
4. SmartSpice, Release Notes, Silvaco Inc. 2014.
5. Van Uffelen M., Geboers S., Leroux P., Berghmans F. SPICE Modeling of a Discrete COTS SiGe IIBT for Digital Applications up to MGy Dose Levels, *IEEE Trans. on Nuclear Science*, 2006, vol. 53, no. 4, pp. 1945—1949.
6. Leroux P., De Cock W., Van Uffelen M., Steyaert M. Modeling, Design, Assessment of 0.4 μm SiGe Bipolar VCSEL Driver IC Under γ -Radiation, *IEEE Trans. on Nuclear Science*, 2009, vol. 56, no. 4, pp. 1920—1925.
7. Deng Y., Fjeldly T., Ytterdal T., Shur M. S. SPICE modeling of neutron displacement damage and annealing effects in bipolar junction transistors, *IEEE Trans. on Electron Devices*, 2003, vol. 50, no. 6, pp. 1873—1877.
8. Gorbunov M. S., Danilov I. A., Zebrev G. I., Osipenko P. N. Verilog-A Modeling of Radiation-Induced Mismatch Enhancement, *IEEE Trans. on Nuclear Science*, 2011, vol. 58, no. 3, pp. 785—792.
9. Alvarado J., Boufouss E., Kilchytska V., Flandre D. Compact model for single event transients and total dose effects at high temperatures for partially depleted SOI MOSFETs. *Microelectronics Reliability*, 2010, vol. 50, pp. 1852—1856.
10. Bu Jianhui, Bi Jinshun, Liu Mengxin, Han Zhengsheng. A total dose radiation model for deep submicron PDSOI NMOS, *Journal of Semiconductors*, 2011, vol. 32, no. 1, pp. 014002-1—014002-3.
11. Petrosyants K., Kozhukhov M. SPICE Model Parameters Extraction Taking into Account the Ionizing Radiation Effects, *Proc. of IEEE East-West Design and Test Symposium (EWDTS)*, 2013, pp. 304—307.
12. Petrosyants K. O., Kozhukhov M. V. SiGe HBT Performance Modeling after Proton Radiation Exposure, *Proc. of IEEE East-West Design and Test Symposium (EWDTS)*, 2012, pp. 274.
13. Petrosyants K. O., Vologdin E. N., Kozhukhov M. V. SiBJT and SiGe HBT Performance Modelling after Neutron Radiation Exposure, *Proc. of IEEE East-West Design and Test Symposium (EWDTS)*, 2011, pp. 267—270.
14. Petrosyants K. O., Sambursky L. M., Kharitonov I. A., Yatmanov A. P. SOI/SOS MOSFET compact macromodel taking into account radiation effects, *Russian Microelectronics*, 2011, vol. 40, Iss. 7, pp. 457—462.
15. Petrosyants K. O., Kharitonov I. A., Popov D. A. Coupled TCAD-SPICE simulation of parasitic BJT effect on SOI CMOS SRAM SEU, *Proc. of IEEE East-West Design and Test Symposium (EWDTS)*, 2013, pp. 312—315.
16. Label K., Barth J. An Insider's Guide to Designing Spacecraft Systems and Instruments for Operation in the Natural Space Radiation Environment, *Proc. of GSFC Systems Eng. Seminar*, 2001, pp. 562.
17. Danilin N., Shipilov K. Osobennosti primeneniya novoy elektronno-komponentnoj bazy v kosmicheskikh razrabotkakh, *Vestnik elektroniki*, 2012, no. 2 (40), pp. 42—45.
18. Prokopenko N. N., Budiakov A. S., Savchenko E. M. Operatsionnye usiliteli s obobshhennoj tokovoy obratnoj svyaz'yu, *Problemy razrabotki perspektivnykh mikro- i nanoelektronnykh sistem — 2008*. Sbornik nauchnykh trudov. Pod obshh. red. A. L. Stempkovskogo, Moscow, IPPM RAS, 2008, pp. 330—333.
19. Petrosjanc K. O., Kharitonov I. A. VLSI device parameters extraction for radiation hardness modeling with SPICE, *Proc. of IEEE International Conference on Microelectronic Test Structures*, 1993, pp. 9—14.
20. Petrosjanc K. O., Sambursky L. M., Kharitonov I. A. A Compact SOI/SOS MOSFET Macromodel Accounting for Radiation Effects, *Proc. of IEEE East-West Design and Test Symposium (EWDTS)*, 2007, pp. 360—364.
21. Petrosjanc K. O., Kharitonov I. A., Orekhov E. V., Sambursky L. M., Yatmanov A. P. Simulation of Radiation Effects in SOI CMOS Circuits with BSIMSOI-RAD macromodel, *Proc. of IEEE East-West Design and Test Symposium (EWDTS)*, 2009, pp. 243—246.
22. Petrosyants K. O., Kharitonov I. A., Sambursky L. M., Bogatyrev V. N., Povarnitsyna Z. M., Drozdenko E. S. Simulation of total dose influence on analog-digital SOI/SOS CMOS circuits with EKV-RAD macromodel, *Proc. of IEEE East-West Design and Test Symposium (EWDTS)*, 2012, pp. 60—65.
23. Petrosyants K. O., Kharitonov I. A., Sambursky L. M. Hardware-Software Subsystem for MOSFETs Characteristic Measurement and Parameter Extraction with Account for Radiation Effects, *Advanced Materials Research*, 2013, vol. 718—720, pp. 750—755.
24. Petrosjanc K. O., Adonin A. S., Kharitonov I. A., Sicheva M. V. SOI device parameter investigation and extraction for VLSI radiation hardness modeling with SPICE, *Proc. of IEEE International Conference on Microelectronic Test Structure*, 1994, pp. 126—129.
25. Enz C. C., Krummenacher F., Vittoz E. A. An analytical MOS transistor model valid in all regions of operation and dedicated to low-voltage and low-current applications, *Analog Integrated Circuits and Signal Processing*, 1995, vol. 8, no. 1, pp. 83—114.
26. Petrosyants K. O., Bogatyrev V. N., Gomanilova N. B., Povarnitsyna Z. M., Shchekin A. A., Kharitonov I. A., Sambursky L. M. *Proektirovanie radiacionno-stojkogo precizionnogo usilitelja na baze SOS CMOS tehnologij*. Sbornik nauchnykh trudov "Elektronika, mikro- i nanoelektronika". Pod red. V. Ja. Stenina, Moscow, MPhI, 2013, pp. 196—302.
27. Petrosyants K. O., Kharitonov I. A., Sambursky L. M., Orekhov E. V., Cherny A. I., Povarnitsyna Z. M., Bogatyrev V. N., Prokofiev G. V. *Issledovanie harakteristik jelementnoj bazy analogovykh SOI CMOS shem, izgotovlennykh po tehnologii XFAB, s uchjotom summarnoj pogloszhjonnnoj dozy*. Sbornik nauchnykh trudov. "Elektronika, mikro- i nanoelektronika". Pod red. V. Ja. Stenina, Moscow: MPhI, 2009, pp. 57—66.

ЦИФРОВАЯ ОБРАБОТКА СИГНАЛОВ И ИЗОБРАЖЕНИЙ

DIGITAL PROCESSING OF SIGNALS AND IMAGES

УДК 004.315

Н. И. Червяков, д-р техн. наук, проф., зав. каф., e-mail: k-fmf-primath@stavs.ru,

П. А. Ляхов, канд. физ.-мат. наук, доц. каф., e-mail: ljahov@mail.ru,

Д. И. Калита, аспирант, ассистент кафедры, e-mail: diana.kalita@mail.ru

Северо-Кавказский федеральный университет, г. Ставрополь

Выбор оптимального набора модулей системы остаточных классов для повышения производительности фильтров с конечной импульсной характеристикой

Представлены результаты исследования производительности различных наборов модулей системы остаточных классов для цифровой обработки сигналов на основе фильтров с конечной импульсной характеристикой (КИХ). Показано, что наилучшую скорость фильтрации обеспечивают модули $\{2^n - 1, 2^n, 2^n + 1, 2^{n+1} - 2^{\frac{n+1}{2}} + 1, 2^{n+1} + 2^{\frac{n+1}{2}} + 1\}$ и $\{2^n - 1, 2^n, 2^n + 1, 2^{n+1} - 1\}$. Разница в производительности между наилучшим и наихудшим наборами модулей достигает 93 %. Полученные результаты могут быть использованы для улучшения характеристик устройств цифровой обработки сигналов в системе остаточных классов и применены для обработки изображений, видео, речи и в других приложениях.

Ключевые слова: система остаточных классов, цифровая обработка сигналов, КИХ-фильтр, набор модулей

Введение

Одним из основных способов цифровой обработки сигналов (ЦОС) является фильтрация. В свою очередь, фильтры подразделяются на нерекурсивные — фильтры с конечной импульсной характеристикой (КИХ-фильтры) и рекурсивные — фильтры с бесконечной импульсной характеристикой (БИХ-фильтры) [1]. КИХ-фильтры используют меньше аппаратных затрат, что делает их проектирование проще по сравнению с БИХ-фильтрами. На сегодняшний день цифровая фильтрация с использованием КИХ-фильтров применяется практически везде, где требуется высокоэффективная обработка сигналов, в частности, в спектральном анализе, обработке изображений, обработке звука, обработке видео и во многих других приложениях [2].

Наиболее перспективным способом повышения производительности систем ЦОС является использование системы остаточных классов (СОК) вместо традиционной двоичной арифметики. Основным преимуществом арифметики остатков по сравнению с позиционной системой счисления (ПСС) является возможность выполнения операций сложения, умножения и вычитания параллельно, благодаря чему повышается скорость работы цифровых устройств [3]. Тем не менее, главным ограничением системы остаточных классов является процесс медленного и сложного выполнения таких опера-

ций, как деление, сравнение, определение знака и обнаружение переполнения, а также восстановление числа по его остаткам [4].

Одним из наиболее важных вопросов теории СОК является определение наилучшего набора модулей для конкретного приложения [5]. Выбор набора модулей играет важную роль в выполнении операций СОК, а также может повлиять на приложения, ее использующие [6]. В данной статье приведены результаты исследования производительности различных модулей СОК для ЦОС на основе КИХ-фильтров.

Введение в СОК

В СОК числа представляются в базисе взаимно простых чисел, называемых модулями $\beta = \{p_1, \dots, p_k\}$, $\text{НОД}(p_i, p_j) = 1$, для $i \neq j$. Произведение всех моду-

лей СОК $P = \sum_{i=1}^k p_i$ называется динамическим диа-

пазоном системы. Любое целое число $0 \leq X < P$ может быть единственным образом представлено в СОК в виде вектора $\{x_1, x_2, \dots, x_k\}$, где $x_i = |X|_{p_i} = X \bmod p_i$ [7].

Динамический диапазон СОК обычно делится на две примерно равные части таким образом, чтобы примерно половина диапазона представляла положительные числа, а остальная часть диапазона —

отрицательные. Таким образом, любое целое число, удовлетворяющее одному из двух соотношений

$$-\frac{P-1}{2} \leq X \leq \frac{P-1}{2} \text{ для нечетных } P,$$

$$-\frac{P}{2} \leq X \leq \frac{P}{2} \text{ для четных } P,$$

может быть представлено в СОК.

Операции сложения, вычитания и умножения в СОК определяются формулами

$$A \pm B = (|a_1 \pm b_1|_{p_1}, \dots, |a_k \pm b_k|_{p_k}), \quad (1)$$

$$A \times B = (|a_1 \times b_1|_{p_1}, \dots, |a_k \times b_k|_{p_k}). \quad (2)$$

Равенства (1), (2) показывают параллельную природу СОК, свободную от поразрядных переносов.

Восстановление числа X по остаткам $\{x_1, x_2, \dots, x_k\}$ основано на Китайской теореме об остатках (КТО) [3]

$$X = \left| \sum_{i=1}^k \left| P_i^{-1} \right|_{p_i} x_i P_i \right|_{P}, \quad (3)$$

где $P_i = \frac{P}{p_i}$. Элемент $|P_i^{-1}|_{p_i}$ означает мультипликативный обратный для P_i по модулю p_i .

Существует, однако, и другой способ преобразования числа из СОК в позиционную форму, основанный на применении обобщенной позиционной системы счисления [5]. Число $X < P$ имеет вид $\{x'_1, x'_2, \dots, x'_k\}$, $0 < x'_i \leq p_i$, в обобщенной позиционной системе счисления, если:

$$X = x'_1 + x'_2 p_1 + x'_3 p_1 p_2 + \dots + x'_k \prod_{i=1}^{k-1} p_i, \quad (4)$$

где $x'_i \in [0, p_i)$ — цифры числа X в обобщенной позиционной системе счисления, и

$$\begin{aligned} x'_1 &= x_1 \bmod p_1; \\ x'_2 &= (x_2 - x'_1) c_{12} \bmod p_2; \\ x'_3 &= ((x_3 - x'_1) c_{13} - x'_2) c_{23} \bmod p_3; \\ &\dots \\ x'_k &= (\dots((x_k - x'_1) c_{1k} - x'_2) c_{2k} - \dots \\ &\dots - x'_{k-1}) c_{k-i,k} \bmod p_k. \end{aligned} \quad (5)$$

Константы c_{ij} являются мультипликативными обратными элементами для p_i по модулю p_j для всех $1 \leq i \leq j \leq k$, т. е. $c_{ij} p_i \equiv 1 \bmod p_j$ для $1 \leq i \leq n$, и могут быть вычислены, например, с помощью алгоритма Евклида.

Таким образом, преимущества представления чисел в СОК могут быть описаны следующим образом [8]:

- так как в СОК отсутствует распространение переноса между арифметическими блоками и числа большой размерности представляются в виде

небольших остатков, это приводит к ускорению в обработке данных;

- при представлении данных с использованием СОК числа большой размерности кодируются в набор небольших остатков, соответственно уменьшается сложность арифметических устройств в каждом канале модуля, что облегчает и упрощает работу вычислительной системы;
- СОК является непозиционной системой без отсутствия зависимости между своими арифметическими блоками, следовательно, ошибка в одном канале не распространяется на другие. Таким образом, облегчается процесс обнаружения и исправления ошибок.

Однако такие операции, как деление, сравнение двух чисел, обнаружение знака являются трудоемкими и дорогими в СОК. Для этих проблемных операций были предложены многие решения. Большинство из них состоит в преобразовании остатка в двоичную систему (обратное преобразование). Вместе с тем, выбор правильного набора модулей является еще одним важным вопросом для построения эффективной СОК с достаточным динамическим диапазоном [9].

Одним из самых известных наборов модулей является набор вида $\{2^n - 1, 2^n, 2^n + 1\}$ [10]. Этот набор известен как средство упрощения расчетов, которые необходимы для осуществления операции обратного преобразования. Тем не менее, арифметические схемы, которые используют модуль $(2^n + 1)$ имеют большую задержку среди всех трех каналов. Таким образом, для того чтобы упростить сложность, вызванную данным модулем, были предложены новые наборы модулей: $\{2^{n-1} - 1, 2^n - 1, 2^n\}$ [11] и $\{2^n - 1, 2^n, 2^{n+1} - 1\}$ [12]. Эти три набора имеют $3n$ -битный динамический диапазон. Так как для многих приложений ЦОС необходим больший динамический диапазон, были предложены новые наборы модулей $\{2^n - 1, 2^n, 2^{2n+1} - 1\}$ [13] и $\{2^n - 1, 2^n + 1, 2^{2n} + 1\}$ [14], которые обеспечивают $4n$ -битный диапазон, а также $\{2^n - 1, 2^{2n}, 2^{2n+1} + 1\}$ [15], который дает $5n$ -битный динамический диапазон. Несмотря на то что динамический диапазон стал больше, увеличилась задержка работы арифметических устройств, основанных на этих наборах, в связи с использованием модулей с большими величинами. Чтобы устранить этот недостаток и сохранить большой динамический диапазон, были предложены наборы, имеющие от четырех до пяти модулей, например, $\{2^n - 1, 2^n, 2^n + 1, 2^{n+1} - 1\}$ [16], $\{2^n - 1, 2^n, 2^n + 1, 2^{n+1} + 1\}$ [16], $\{2^n - 1, 2^n, 2^n + 1, 2^{2n} + 1\}$ [17], $\{2^n - 1, 2^n, 2^n + 1, 2^{2n+1} - 1\}$ [18], $\{2^n - 1, 2^n, 2^n + 1, 2^n - 2^{\frac{n+1}{2}} + 1, 2^n + 2^{\frac{n+1}{2}} + 1\}$ [19],

$$\{2^n - 1, 2^n + 1, 2^{2n} - 1, 2^{2n+1} - 3\} [20],$$

$$\{2^n - 1, 2^{2n}, 2^n + 1, 2^{2n+1}\} [18],$$

$$\{2^n - 1, 2^n, 2^n + 1, 2^{n-1} - 1, 2^{n+1} + 1\} [21]$$

и $\{2^n, 2^{\frac{n}{2}} - 1, 2^{\frac{n}{2}} + 1, 2^n + 1, 2^{2n-1} - 1\} [22].$

Каждый из этих наборов имеет свои преимущества и недостатки. Некоторые из них предлагают большой динамический диапазон, в то время как другие имеют большую степень параллелизма. Некоторые из них могут привести к более эффективному обратному преобразованию, в то время как другие — к получению более эффективного остатка арифметических устройств [23]. Как будет видно далее, выбор наиболее эффективного набора модулей играет существенную роль в скорости работы КИХ-фильтра.

Проектирование КИХ-фильтров в СОК

Любой КИХ-фильтр можно задать следующим выражением:

$$y_n = \sum_{k=0}^N b_k x_n - k, \tag{6}$$

где x_n — входная последовательность сигнала; b_k — коэффициенты фильтра; N — порядок фильтра; y_n — последовательность сигнала, полученного на выходе фильтра [24].

Из формулы (6) видно, что для работы КИХ-фильтра используются только операции сложения и умножения, что позволяет при его реализации использовать положительные качества СОК.

Рассмотрим низкочастотный КИХ-фильтр 15-го порядка, АЧХ которого представлена на рис. 1. Проектирование фильтра было осуществлено в MATLAB® с использованием встроенного приложения Filter Design and Analysis. Фильтр, АЧХ которого изображена на рис. 1, построен с использованием алгоритма Паркса—МакКлеллана [25]. Коэффициенты фильтра в формате двойной точности и в целочисленном формате данных приведены в табл. 1.

Для получения целочисленных значений данного фильтра была использована команда

```
num2int(quantizer([12, 11]), z)
```

Таким образом, полученные целочисленные значения коэффициентов фильтра являются 12-битными числами, из которых 11 бит — величина числа, 1 бит — знак. Эти целочисленные значения в дальнейшем могут быть представлены в СОК путем простого вычисления остатков от деления на модули СОК.

Так как исходные коэффициенты фильтра являются числами двойной точности длиной 64 бит, то переход от такого представления к 12-битному влечет за собой ошибку округления. На рис. 2 показана АЧХ этой ошибки. Из рисунка видно, что максимум ошибки примерно на 25 дБ ниже, чем область записания фильтра.

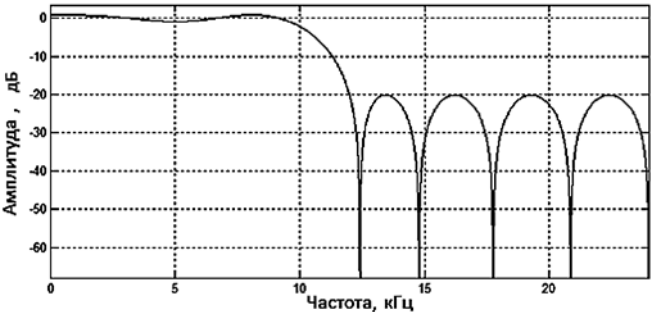


Рис. 1. АЧХ низкочастотного фильтра 15-го порядка

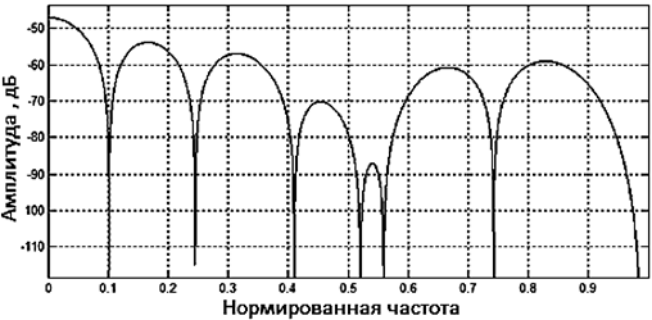


Рис. 2. АЧХ ошибки округления КИХ-фильтра

Таблица 1

Коэффициенты КИХ-фильтра 15-го порядка

Коэффициенты фильтра	Числовой формат представления	
	Двойная точность (64 бит)	Целое число (12 бит)
$b_0 = b_{15}$	−0,04842854521480	−100
$b_1 = b_{14}$	0,031641772569045	64
$b_2 = b_{13}$	0,066305238639482	135
$b_3 = b_{12}$	0,016109250214128	32
$b_4 = b_{11}$	−0,076170627278596	−156
$b_5 = b_{10}$	−0,041780626517982	−86
$b_6 = b_9$	0,183762552952241	376
$b_7 = b_8$	0,417189375297071	854

Для равномерного распределения последствий из-за возникновения ошибок округления между фильтром и сигналом рассмотрим случай, когда в качестве входной последовательности используется сигнал с такой же точностью, что и коэффициенты фильтра, т. е. 12 бит. Тогда диапазон работы КИХ-фильтра равен:

$$\max\{|y(n)|\} = 2^{12} \sum_{k=0}^{15} |b_k| = 4096 \cdot 3606 = 14\,770\,176. \tag{7}$$

Передаточная функция фильтра, описывающая связь между сигналами на входе и выходе фильтра, задается в виде

$$H(z) = b_0 + b_1 z^{-1} + \dots + b_{15} z^{-15} = \sum_{i=0}^{15} b_i z^{-i}. \tag{8}$$

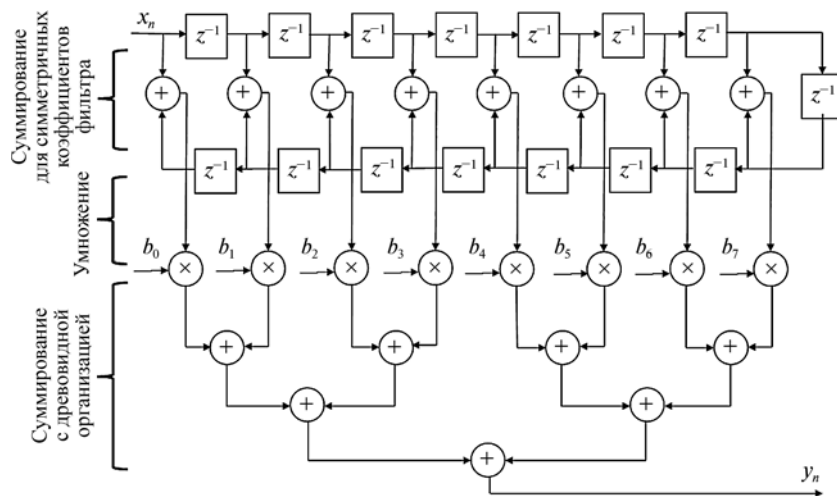


Рис. 3. Схема работы фильтра 15-го порядка

С учетом того, что коэффициенты построенного фильтра симметричны, выражение (8) можно переписать в следующем виде:

$$H(z) = \sum_{k=0}^8 b_k (z^k + z^{15-k}). \quad (9)$$

На рис. 3 представлена схема фильтра, описанного формулой (9), где блок z^{-1} — это блок задержки. При реализации изображенной схемы фильтра в СОК достаточно заменить коэффициенты b_i , $i = 0, \dots, 15$, на их модулярное представление, а блоки сложения и умножения — на блоки сложения по модулю и умножения по модулю соответственно.

Исследование наборов модулей СОК для фильтрации

Для оценки времени работы КИХ-фильтров в СОК проводился подсчет тактов синхронизации вычислительной системы, необходимых для выполнения полного цикла работы фильтра при обработке одного вновь поступившего отсчета входной последовательности. В КИХ-фильтрах в СОК используются только операции сложения и умножения, которые в свою очередь тем или иным способом (в зависимости от конкретного вида модулей СОК) сводится к комбинации базовых логических элементов. В соответствии с методом, предложенным в работе [26], использовались следующие значения числа тактов синхронизации вычислительной системы для базовых элементов:

- время работы двухвходовых элементов "И", "ИЛИ", "И-НЕ", "ИЛИ-НЕ" принимается равным $T_1 = 1$;
- время работы двухвходовых элементов "ИСКЛЮЧАЮЩЕЕ ИЛИ", а также мультиплексора 2:1 принимается равным $T_2 = 2$;
- время работы двоичного полусумматора принимается равным $T_3 = 2$;

- время работы полного двоичного сумматора принимается равным $T_4 = 4$. Как уже отмечалось ранее, выбор набора модулей играет важную роль в улучшении производительности КИХ-фильтров, построенных в СОК. Проведем анализ производительности КИХ-фильтра с использованием наиболее известных из них. В табл. 2 приведено время задержки при модулярном сложении и умножении для каждого рассмотренного набора модулей.

Для каждого из приведенных в табл. 2 наборов модулей было подобрано наименьшее n , обеспечивающее необходимый диапазон, указанный в формуле (7), т. е. все наборы модулей должны быть взаимно простыми и удовлетворять условию $P \geq 14\,770\,176$. В табл. 3 приведены минимальные значения n , удовлетворяющие этому условию.

В наборе модулей $\{2^n - 1, 2^n, 2^n + 1, 2^{n-1} - 1, 2^{n-1} + 1\}$ (см. табл. 2) при наименьшем необходимом $n = 5$ не выполняется требование взаимной простоты. Поэтому этот набор был исключен из исследования. При реализации фильтра 15-го порядка логическая глубина выполнения операций равна 5, при этом используется одно параллельное умножение и 4 параллельных сложения (см. рис. 3). Задержка для модулярного сложения (рис. 4) и умножения получена с учетом данных из табл. 2. В по-

Таблица 2

Время выполнения операций модулярного сложения и умножения для разных наборов модулей СОК

Набор модулей	Задержка, такты	
	Модулярное сложение	Модулярное умножение
$\{2^n - 1, 2^n, 2^n + 1\}$	$8n + 11$	$16n + 12$
$\{2^{n-1} - 1, 2^n - 1, 2^n\}$	$8n$	$16n - 7$
$\{2^n - 1, 2^n, 2^{n+1} - 1\}$	$8n + 8$	$16n + 9$
$\{2^n - 1, 2^n, 2^{2n+1} - 1\}$	$16n + 8$	$32n + 9$
$\{2^n - 1, 2^n + 1, 2^{2n} + 1\}$	$16n + 11$	$32n + 12$
$\{2^n - 1, 2^n, 2^n + 1, 2^{n+1} - 1\}$	$8n + 11$	$16n + 12$
$\{2^n - 1, 2^n, 2^n + 1, 2^{n+1} + 1\}$	$8n + 19$	$16n + 28$
$\{2^n - 1, 2^{2n} - 1, 2^{2n} + 1\}$	$16n + 11$	$32n + 12$
$\{2^n - 1, 2^n, 2^n + 1, 2^{2n} + 1\}$	$16n + 11$	$32n + 12$
$\{2^n - 1, 2^n, 2^n + 1, 2^{2n+1} - 1\}$	$16n + 8$	$32n + 9$
$\{2^n - 1, 2^n, 2^n + 1, 2^{2n} - 2^{\frac{n+1}{2}} + 1, 2^n + 2^{\frac{n+1}{2}} + 1\}$	$8n + 11$	$24n + 13$
$\{2^n - 1, 2^n, 2^n + 1, 2^{n-1} - 1, 2^{n+1} + 1\}$	$8n + 19$	$16n + 28$
$\{2^n, 2^{\frac{n}{2}} - 1, 2^{\frac{n}{2}} + 1, 2^n + 1, 2^{n-1} - 1\}$	$16n - 8$	$32n - 23$
$\{2^n - 1, 2^n + 1, 2^{2n} - 1, 2^{2n+1} - 3\}$	$16n + 11$	$48n + 13$
$\{2^n - 1, 2^{2n}, 2^n + 1, 2^{2n} + 1\}$	$16n + 11$	$28n + 12$

Время (в тактах) выполнения модульных операций для фильтрации в СОК с различными наборами модулей

№ набора модулей	Набор модулей	n	Десятичная запись модулей	Задержка, такты		Итого
				+	×	
1	$\{2^n - 1, 2^n, 2^n + 1\}$	8	{255,256,257}	75	140	440
2	$\{2^{n-1} - 1, 2^n - 1, 2^n\}$	9	{255,511,512}	72	137	425
3	$\{2^n - 1, 2^n, 2^{n+1} - 1\}$	7	{127,128,255}	64	121	377
4	$\{2^n - 1, 2^n, 2^{2n} - 1\}$	6	{63,64,8191}	104	201	617
5	$\{2^n - 1, 2^n + 1, 2^{2n} + 1\}$	6	{63,65,4097}	107	204	632
6	$\{2^n - 1, 2^n, 2^n + 1, 2^{n+1} - 1\}$	6	{63,64,65,127}	59	108	344
7	$\{2^n - 1, 2^n, 2^n + 1, 2^{n+1} + 1\}$	7	{127,128,129,257}	75	140	440
8	$\{2^n - 1, 2^{2n} - 1, 2^{2n} + 1\}$	5	{32,1023,1025}	91	187	551
9	$\{2^n - 1, 2^n, 2^n + 1, 2^{2n} + 1\}$	5	{31,32,33,1025}	91	287	651
10	$\{2^n - 1, 2^n, 2^n + 1, 2^{2n+1} - 1\}$	5	{31,32,33,2047}	88	169	521
11	$\{2^n - 1, 2^n, 2^n + 1, 2^{2n} - 2^{\frac{n+1}{2}} + 1, 2^n + 2^{\frac{n+1}{2}} + 1\}$	5	{31,32,33,25,41}	51	133	337
12	$\{2^n, 2^{\frac{n}{2}} - 1, 2^{\frac{n}{2}} + 1, 2^{2n} + 1, 2^{n-1} - 1\}$	6	{64,7,9,65,2047}	88	169	521
13	$\{2^n - 1, 2^n + 1, 2^{2n} - 1, 2^{2n+1} - 3\}$	4	{15,17,254,509}	75	205	505
14	$\{2^n - 1, 2^{2n}, 2^n + 1, 2^{2n} + 1\}$	4	{15,256,17,257}	75	124	424

следней колонке табл. 3 приведен суммарный результат числа необходимых тактов на выполнение всех модульных операций при фильтрации в СОК. Анализ полученных результатов позволяет сделать вывод, что наиболее эффективными наборами модулей для ЦОС являются $\{2^n - 1, 2^n, 2^n + 1, 2^n - 2^{\frac{n+1}{2}} + 1, 2^n + 2^{\frac{n+1}{2}} + 1\}$ и $\{2^n - 1, 2^n, 2^n + 1, 2^{n+1} - 1\}$. Наименее эффективными наборами модулей, т. е. такими, которые будут приносить наибольшую задержку при выполнении модульных операций, являются наборы $\{2^n - 1, 2^n, 2^n + 1, 2^{2n} + 1\}$ и $\{2^n - 1, 2^n + 1, 2^{2n} + 1\}$.

Среди всех трехмодульных наборов наилучший результат дает набор $\{2^n - 1, 2^n, 2^{n+1} - 1\}$, его задержка составляет 377 тактов, наихудшим является набор $\{2^n - 1, 2^n + 1, 2^{2n} + 1\}$ с задержкой в 632 такта. Наиболее эффективным среди четырехмодульных наборов является набор $\{2^n - 1, 2^n, 2^n + 1, 2^{n+1} - 1\}$ с задержкой, равной 344 такта, а наименее эффективным — набор $\{2^n - 1, 2^n, 2^n + 1, 2^{2n} + 1\}$ с задержкой в 651 такт. Для пятимодульных наборов наименьшую задержку в 337 тактов дает набор $\{2^n - 1, 2^n, 2^n + 1, 2^{2n} - 2^{\frac{n+1}{2}} + 1, 2^n + 2^{\frac{n+1}{2}} + 1\}$, наибольшую в 521 такт — набор $\{2^n - 1, 2^n, 2^n + 1, 2^{2n+1} - 1\}$.

Примем время работы самого быстрого набора модулей $\{2^n - 1, 2^n, 2^n + 1, 2^n - 2^{\frac{n+1}{2}} + 1, 2^n + 2^{\frac{n+1}{2}} + 1\}$, равным 100 %. В табл. 4 показано относительное время работы разных наборов модулей СОК, в сравнении с этим набором модулей. Самый эффектив-

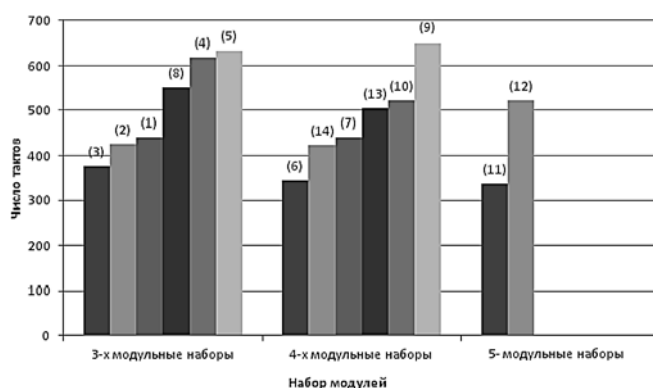


Рис. 4. Модульная задержка наборов модулей. Набор модулей соответствует порядковому номеру в табл. 3

Таблица 4

Сравнение времени фильтрации в СОК для разных наборов модулей

Набор модулей	Итоговая задержка, %
$\{2^n - 1, 2^n, 2^n + 1, 2^{2n} - 2^{\frac{n+1}{2}} + 1, 2^n + 2^{\frac{n+1}{2}} + 1\}$	100
$\{2^n - 1, 2^n, 2^n + 1, 2^{n+1} - 1\}$	102
$\{2^n - 1, 2^n, 2^{n+1} - 1\}$	112
$\{2^{n-1} - 1, 2^n - 1, 2^n\}$	126
$\{2^n - 1, 2^{2n}, 2^n + 1, 2^{2n} + 1\}$	126
$\{2^n - 1, 2^n, 2^n + 1\}$	131
$\{2^n - 1, 2^n, 2^n + 1, 2^{n+1} + 1\}$	131
$\{2^n - 1, 2^n + 1, 2^{2n} - 1, 2^{2n+1} - 3\}$	150
$\{2^n - 1, 2^n, 2^n + 1, 2^{2n+1} - 1\}$	155
$\{2^n, 2^{\frac{n}{2}} - 1, 2^{\frac{n}{2}} + 1, 2^{2n} + 1, 2^{n-1} - 1\}$	155
$\{2^n - 1, 2^{2n} - 1, 2^{2n} + 1\}$	163
$\{2^n - 1, 2^n, 2^{2n+1} - 1\}$	183
$\{2^n - 1, 2^n + 1, 2^{2n} + 1\}$	188
$\{2^n - 1, 2^n, 2^n + 1, 2^{2n} + 1\}$	193

ный набор модулей $\{2^n - 1, 2^n, 2^n + 1, 2^n - 2^{\frac{n+1}{2}} + 1, 2^n + 2^{\frac{n+1}{2}} + 1\}$ на 93 % работает быстрее, чем самый медленный набор $\{2^n - 1, 2^n, 2^n + 1, 2^{2n} + 1\}$, что подтверждает важность задачи выбора наиболее эффективного набора модулей СОК для ЦОС.

Заключение

В статье проведен анализ скорости работы КИХ-фильтров при использовании различных наборов модулей СОК. Наиболее эффективными оказались

наборы $\{2^n - 1, 2^n, 2^n + 1, 2^n - 2^{\frac{n+1}{2}} + 1, 2^n + 2^{\frac{n+1}{2}} + 1\}$ и $\{2^n - 1, 2^n, 2^n + 1, 2^{n+1} - 1\}$. Наихудшими наборами являются наборы $\{2^n - 1, 2^n, 2^n + 1, 2^{2n} + 1\}$ и $\{2^n - 1, 2^n + 1, 2^{2n} + 1\}$, работающие на 93 % и 88 % соответственно медленнее, чем набор модулей $\{2^n - 1, 2^n, 2^n + 1, 2^n - 2^{\frac{n+1}{2}} + 1, 2^n + 2^{\frac{n+1}{2}} + 1\}$.

Дальнейшая работа по исследованию эффективности различных наборов модулей СОК при реализации цифровых фильтров может быть связана с непосредственной реализацией на программируемых логических интегральных схемах и интегральных схемах специального назначения и оценкой параметров скорости работы, энергопотребления и аппаратных затрат.

Интересным направлением дальнейших исследований в области применения КИХ-фильтров в СОК является поиск наиболее эффективных наборов модулей, обеспечивающих высокую скорость выполнения модульных операций для конкретных приложений. Применение оптимального набора модулей в СОК для обработки изображений, видео, речи, спутниковой связи и других приложениях позволит существенно ускорить обработку данных.

Работа выполнена при финансовой поддержке РФФИ, грант № 14-07-31004-мол-а.

Авторы выражают благодарность рецензенту за ценные замечания к рукописи статьи.

Список литературы

1. Tan L., Jiang J. Digital Signal Processing, Second Edition: Fundamentals and Applications // Academic Press. 2013. 876 p.
2. Shahana T. K., James R. K., Jose B. R., Jacob K. P., Sasi S. Performance Analysis of FIR Digital Filter Design: RNS Versus Traditional // ISCIT 2007 International Symposium on Communications and Information Technologies Proceedings. 2007. P. 1–5.
3. Omondi A., Premkumar B. Residue Number Systems: Theory and Implementation // Imperial College Press. 2007. P. 296.
4. Cardarilli G. C., Nannarelli A., Re M. Residue number system for low-power DSP applications // Proc. 41st Asilomar Conf. Signals, Syst., Comput. 2007. P. 1412–1416.
5. Червяков Н. И., Сахнюк П. А., Шапошников А. В., Макоха А. Н. Нейрокомпьютеры в остаточных классах. М.: Радиотехника, 2003. 272 с.

6. Stamenković N. Digital FIR Filter Architecture Based on the Residue Number System // Facta Universitatis, Ser.: Elec. Energ. 2009. Vol. 22, N. 1. P. 125–140.
7. Afsheh A., Mojoodi A. An improved reverse converter for moduli set $(2^n - 1, 2^n, 2^n + 1)$ // ISCIT 2010 International Symposium on Communications and Information Technologies Proceedings. 2010. P. 928–933.
8. Червяков Н. И., Ляхов П. А. Реализация КИХ-фильтров в системе остаточных классов // Нейрокомпьютеры: разработка, применение. 2012. № 5. С. 15–24.
9. Goh V. T., Siddiqi M. U. Multiple Error Detection and Correction Based on Redundant Residue Number Systems // IEEE Transactions on Communications. 2008. Vol. 56, N. 3. P. 325–330.
10. Piestrak S. J. A high-speed realization of a residue to binary number system converter // IEEE Trans. on Circuits and Systems-II: Analog and Digital Signal Processing. 1995. Vol. 42. P. 661–663.
11. Wang W., Swamy M. N., Ahmad M. O., Wang Y. A high-speed residue-to-binary converter for three-moduli $(2^k, 2^k - 1, 2^{k-1} - 1)$ RNS and a scheme for its VLSI implementation // IEEE Trans. on Circuits and Systems-II: Analog and Digital Signal Processing. 2000. Vol. 47. P. 1576–1581.
12. Mohan P. V. A. RNS-to-binary converter for a new three moduli set $\{2^{n+1} - 1, 2^n, 2^n - 1\}$ // IEEE Trans. on Circuits and Systems-II: Express Briefs. 2007. Vol. 54. P. 775–779.
13. Molahosseini A. S., Navi K., Rafsanjani M. K. A new residue to binary converter based on mixed-radix conversion // 3rd International Conference on ICTTA. 2008. P. 1–6.
14. Wang W., Swamy M. N. S., Ahmad O., Wang Y. A study of the residue-to-binary converters for the three-moduli sets // IEEE Trans. on Circuits and Systems-I: Fundamental Theory and Applications. 2003. Vol. 50. P. 235–243.
15. Hariri A., Navi K., Rastegar R. A new high dynamic range moduli set with efficient reverse converter // Computers & Mathematics with Applications Journal. 2008. Vol. 55. P. 660–668.
16. Mohan P. V. A., Premkumar A. B. RNS-to-binary converters for two four-moduli sets $\{2^n - 1, 2^n, 2^n + 1, 2^{n+1} - 1\}$ and $\{2^n - 1, 2^n, 2^n + 1, 2^{n+1} + 1\}$ // IEEE Trans. on Circuits and Systems-I: Regular Papers. 2007. Vol. 54. P. 1245–1254.
17. Cao B., Chang C. H., Srikanthan T. An efficient reverse converter for the 4-moduli set $\{2^n - 1, 2^n, 2^n + 1, 2^{2n} + 1\}$ // IEEE Trans. on Circuits and Systems-I: Fundamental Theory and Applications. 2003. Vol. 50. P. 1296–1303.
18. Molahosseini A. S., Navi K., Dadkhah C., Kavehei O., Timarchi S. Efficient reverse converter designs for the new 4-moduli sets $\{2^n - 1, 2^n, 2^n + 1, 2^{2n+1} - 1\}$ and $\{2^n - 1, 2^n + 1, 2^{2n}, 2^{2n} + 1\}$ based on new CRTs // IEEE Trans. on Circuits and Systems-I: Regular Papers. 2010. Vol. 57. P. 823–835.
19. Hiasat A. A. VLSI implementation of new arithmetic residue to binary decoders // IEEE Trans. on VLSI Systems. 2005. Vol. 13. P. 153–158.
20. Zhang W., Siy P. An efficient design of residue to binary converter for four moduli set $\{2^n - 1, 2^n + 1, 2^{2n} - 2, 2^{2n+1} - 3\}$ based on new CRT-II // Information Sciences Journal. 2008. Vol. 178. P. 264–279.
21. Cao B., Chang C. H., Srikanthan T. A residue-to-binary converter for a new five-moduli set // IEEE Trans. on Circuits and Systems-I: Regular Papers. 2007. Vol. 54. P. 1041–1049.
22. Molahosseini A. S., Dadkhah C., Navi K. A new five moduli set for efficient hardware implementation of the reverse converter // IEICE Electronics Express. 2009. Vol. 6. P. 1006–1012.
23. Antro S. F., Bajard J. C., Sousa L. RNS based Elliptic Curve Point Multiplication for Massive Parallel Architectures // Computer Journal. Oxford University Press. 2011.
24. Стемповский А. Л., Корнилов А. И., Семенов М. Ю. Особенности реализации устройств цифровой обработки сигналов в интегральном исполнении с применением модулярной арифметики // Информационные технологии. 2004. № 2. С. 2–9.
25. Ingle V. K., Proakis J. K. Digital Signal Processing Using MATLAB® // ser. BookWare Companion Series, New York, Madrid: Brooks/Cole Publishing Company. 1999. 218 p.
26. Younes D., Steffan P. A. Comparative Study on Different Moduli Sets in Residue Number System // Dept. of Microelectronics Brno University of Technology Brno, Czech Republic, 2013.

Selection of the Optimal RNS Moduli to Improve FIR Filters Performance

The paper presents the results of a study the performance of different Residue Number System moduli sets for digital signal processing based on the Finite Impulse Response filters. It is shown that the best filtration rate provide by moduli sets $\{2^n - 1, 2^n, 2^n + 1, 2^n - 2^{\frac{n+1}{2}} + 1, 2^n + 2^{\frac{n+1}{2}} + 1\}$ and $(2^n - 1, 2^n, 2^n + 1, 2^{n+1} - 1)$. The difference in performance between the best and the worst moduli sets up to 93 %. The results can be used to improve the performance of the digital signal processing devices used Residue Number System and applied to the processing of images, video, speech and other applications.

Keywords: residue number system, digital signal processing, FIR filter, moduli set

References

1. Tan L., Jiang J. *Digital Signal Processing, Second Edition: Fundamentals and Applications*, Academic Press, 2013. 876 p.
2. Shahana T. K., James R. K., Jose B. R., Jacob K. P., Sasi S. Performance Analysis of FIR Digital Filter Design: RNS Versus Traditional, *ISCIT 2007 International Symposium on Communications and Information Technologies Proceedings*, 2007, pp. 1–5.
3. Omondi A., Premkumar B. Residue Number Systems: *Theory and Implementation*, Imperial College Press, 2007. 296 p.
4. Cardarilli G. C., Nannarelli A., Re M. Residue number system for low-power DSP, applications, *Proc. 41st Asilomar Conf. Signals, Syst., Comput*, 2007, pp. 1412–1416.
5. Chervyakov N. I., Sakhyuk P. A., Shaposhnikov A. V., Makokha A. N. *Neirokomp'yutery v ostatochnykh klassakh* [Neurocomputers in residue numbers], Moscow: Radiotekhnika, 2003. 272 p. (In Russian).
6. Stamenković N. Digital FIR Filter Architecture Based on the Residue Number System, *Facta Universitatis, Ser.: Elec. Energ*, 2009, vol. 22, no. 1, pp. 125–140.
7. Afsheh A., Mojoodi A. An improved reverse converter for moduli set $\{2^n - 1, 2^n, 2^n + 1\}$, *ISCIT 2010 International Symposium on Communications and Information Technologies Proceedings*, 2010, pp. 928–933.
8. Chervyakov N. I., Lyakhov P. A. Realizatsiya KIX-fil'trov v sisteme ostatochnykh klassov [Implementation of FIR filters in Residue Number System], *Neirokomp'yutery: razrabotka, primeneniye*, 2012, no. 5, pp. 15–24. (In Russian).
9. Goh V. T., Siddiqi M. U. Multiple Error Detection and Correction Based on Redundant Residue Number Systems, *IEEE Transactions on Communications*, 2008, vol. 56, no. 3, pp. 325–330.
10. Piestrak S. J. A high-speed realization of a residue to binary number system, converter, *IEEE Trans. on Circuits and Systems-II: Analog and Digital Signal Processing*, 1995, vol. 42, pp. 661–663.
11. Wang W., Swamy M. N., Ahmad M. O., Wang Y. A high-speed residue-to-binary converter for three-moduli $(2^k, 2^k - 1, 2^{k-1} - 1)$ RNS and a scheme for its VLSI implementation, *IEEE Trans. on Circuits and Systems-II: Analog and Digital Signal Processing*, 2000, vol. 47, pp. 1576–1581.
12. Mohan P. V. A. RNS-to-binary converter for a new three moduli set $\{2^{n+1} - 1, 2^n, 2^n - 1\}$, *IEEE Trans. on Circuits and Systems-II: Express Briefs*, 2007, vol. 54, pp. 775–779.
13. Molahosseini A. S., Navi K., Rafsanjani M. K. A new residue to binary converter based on mixed-radix conversion, *3rd International Conference on ICTTA*, 2008, pp. 1–6.
14. Wang W., Swamy M. N. S., Ahmad O., Wang Y. A study of the residue-to-binary converters for the three-moduli sets, *IEEE Trans. on Circuits and Systems-I: Fundamental Theory and Applications*, 2003, vol. 50, pp. 235–243.
15. Hariri A., Navi K., Rastegar R. A new high dynamic range moduli set with efficient reverse converter. *Computers & Mathematics with Applications Journal*, 2008, vol. 55, pp. 660–668.
16. Mohan P. V. A., Premkumar A. B. RNS-to-binary converters for two four-moduli sets $\{2^n - 1, 2^n, 2^n + 1, 2^{n+1} - 1\}$ and $\{2^n - 1, 2^n, 2^n + 1, 2^{n+1} + 1\}$, *IEEE Trans. on Circuits and Systems-I: Regular Papers*, 2007, vol. 54, pp. 1245–1254.
17. Cao B., Chang C. H., Srikanthan T. An efficient reverse converter for the 4-moduli set $\{2^n - 1, 2^n, 2^n + 1, 2^{2n} + 1\}$, *IEEE Trans. on Circuits and Systems-I: Fundamental Theory and Applications*, 2003, vol. 50, pp. 1296–1303.
18. Molahosseini A. S., Navi K., Dadkhah C., Kavehei P., Timarchi S. Efficient reverse converter designs for the new 4-moduli sets $\{2^n - 1, 2^n, 2^n + 1, 2^{2n} + 1\}$ and $\{2^n - 1, 2^n + 1, 2^{2n}, 2^{2n} + 1\}$ based on new CRTs, *IEEE Trans. on Circuits and Systems-I: Regular Papers*, 2010, vol. 57, pp. 823–835.
19. Hiasat A. A. VLSI implementation of new arithmetic residue to binary decoders, *IEEE Trans. on VLSI Systems*, 2005, vol. 13, pp. 153–158.
20. Zhang W., Siy P. An efficient design of residue to binary converter for four moduli set $\{2^n - 1, 2^n + 1, 2^{2n} - 2, 2^{2n} + 1 - 3\}$ based on new CRT-II, *Information Sciences Journal*, 2008, vol. 178, pp. 264–279.
21. Cao B., Chang C. H., Srikanthan T. A residue-to-binary converter for a new five-moduli set, *IEEE Trans. on Circuits and Systems-I: Regular Papers*, 2007, vol. 54, pp. 1041–1049.
22. Molahosseini A. S., Dadkhah C., Navi K. A new five moduli set for efficient hardware implementation of the reverse converter, *IEICE Electronics Express*, 2009, vol. 6, pp. 1006–1012.
23. Antro S. F., Bajard J. C., Sousa L. RNS based Elliptic Curve Point Multiplication for Massive Parallel Architectures, *Computer Journal*, Oxford University Press, 2011.
24. Stempkovskii A. L., Kornilov A. I., Semenov M. Yu. Osobennosti realizatsii ustroystv tsifrovoy obrabotki signalov v integral'nom ispolnenii s primeneniem modulyarnoi arifmetiki [Features of digital signal processing devices realization integrally with the use of modular arithmetic], *Informatsionnye tekhnologii*, 2004, no. 2, pp. 2–9. (In Russian).
25. Ingle V. K., Proakis J. K. *Digital Signal Processing Using MATLAB®*, ser. BookWare Companion Series, New York, Madrid: Brooks/Cole Publishing Company, 1999. 218 p.
26. Younes D., Steffan P. A Comparative Study on Different Moduli Sets in Residue Number System, *Dept. of Microelectronics Brno University of Technology Brno*, Czech Republic, 2013.

УДК 004.056.53

К. А. Щеглов, аспирант, А. Ю. Щеглов, д-р техн. наук, проф. кафедры вычислительной техники
Национальный исследовательский университет информационных технологий, механики и оптики,
Санкт-Петербург, Россия, e-mail: info@npp-itb.spb.ru

Интерпретация и моделирование угрозы атаки на информационную систему. Часть 1. Моделирование угрозы уязвимости и интерпретация угрозы атаки

Введена классификация угроз уязвимостей в качестве простейшего элемента информационной безопасности как объекта моделирования, в рамках которой угрозы разделены на угрозы технологических уязвимостей и уязвимостей реализации, создающих угрозы условных технологических уязвимостей. Показано, что в качестве объекта моделирования угроз уязвимостей для последующего моделирования угрозы атаки следует рассматривать именно угрозы уязвимостей реализации, при этом моделирование может проводиться без каких-либо экспертных оценок — исключительно с использованием существующей статистики в отношении параметров безопасности угроз уязвимостей. Обоснована корректность использования для моделирования угроз атак марковских моделей с дискретными состояниями и непрерывным временем, широко применяемых в теории надежности для моделирования систем с отказами и восстановлениями. Вместе с тем, показана необходимость включения в модель угрозы уязвимости элементов схемы "гибели и размножения", ввиду возможности возникновения в системе одновременно нескольких однотипных уязвимостей, создающих одну и ту же угрозу атаки, чем задача моделирования принципиально отличается от задачи моделирования в теории надежности. Предложена интерпретация угрозы атаки схемой последовательного резервирования угроз уязвимостей, определяющая подход к моделированию угрозы атаки на информационную систему с использованием построенных моделей массового обслуживания для угроз уязвимостей. Приведен пример количественной оценки актуальности угрозы уязвимости и угрозы атаки, создаваемой одной угрозой уязвимости реализации. Этот пример иллюстрирует возможности предложенного подхода к моделированию.

Ключевые слова: информационная система, безопасность, угроза уязвимости, угроза атаки, актуальность угрозы, марковская модель, система массового обслуживания, отказ, восстановление, характеристика безопасности

Введение

Известными методами моделирования угроз безопасности информационных систем в качестве простейшего элемента информационной безопасности рассматривается угроза атаки на информационную систему [1–5]. Моделирование угроз атак реализуется в целях выявления актуальных угроз — угроз, от которых требуется реализация защиты. С подобным подходом к моделированию связан ряд принципиальных недостатков. Прежде всего, это необходимость экспертного задания ключевой характеристики безопасности — вероятности возникновения угрозы атаки. Экспертные оценки используются и при построении модели нарушителя [5], что опять же необходимо для оценки актуальности угрозы атаки, но уже применительно к конкретной информационной системе, для которой проектируется система защиты. При моделировании угрозы безопасности информационной системы

в целом, основанном на использовании в качестве простейшего элемента информационной безопасности угрозы атаки, по понятным причинам — не анализируются составляющие угрозы атаки, создаваемой угрозами уязвимостей, угрозы атак рассматриваются в качестве независимых событий, исходя из чего, используются соответствующие расчетные формулы. Но подобный исходный посыл не корректен в принципе, так как реальные угрозы атак создаются выявляемыми в системе уязвимостями, при этом события возникновения угроз атак, как правило, зависимы по уязвимостям, поскольку многими атаками эксплуатируются одни и те же уязвимости. Например, подавляющая часть угроз атак предполагает внедрение и последующее исполнение на защищаемом компьютере вредоносной программы. Важным является и то, что оперируя при моделировании не простейшим элементом безопасности, каким является угроза уязвимости, а угрозами атак, которые крайне разнородны, невоз-

можно обосновать требования к входным параметрам модели. При проектировании же системы защиты крайне важно то, что в конечном счете системой защиты, если ее рассматривать не как некую абстракцию при моделировании, а пытаться реально спроектировать, защита от угроз атак реализуется нивелированием именно соответствующих угроз уязвимостей, создающих эти угрозы атак.

В качестве простейшего элемента безопасности информационной системы при построении математических моделей угрозы атаки на информационную систему предлагается рассматривать угрозу уязвимости, что позволяет использовать существующую и непрерывно ведущуюся статистику в отношении выявляемых и устраняемых уязвимостей при задании входных параметров разрабатываемых моделей.

1. Модели угрозы уязвимости

Под *уязвимостью* в качестве простейшего элемента безопасности информационной системы будем понимать свойство системы, создающее условие для реализации на нее атаки. В общем случае к угрозам уязвимостей информационной системы могут быть отнесены технологические недостатки ее построения, включая отсутствие требуемых функций защиты (технологическая уязвимость), а также ошибки в используемом прикладном и системном программном обеспечении, позволяющие осуществить обход реализованных функций защиты (уязвимость реализации). Угроза атаки на информационную систему в общем случае создается совокупностью угроз уязвимостей — технологических уязвимостей и уязвимостей угроз реализации, наличие (выявление) которых в системе необходимо для осуществления данной атаки.

Уточним. В любом случае атакой эксплуатируется технологическая уязвимость (уязвимость), связанная с технологическими недостатками построения системы, в части реализации ее защиты. Технологические уязвимости могут быть классифицированы — разделены; на безусловные и условные. *Безусловная технологическая уязвимость* предполагает возможность ее использования атакой без возникновения каких-либо дополнительных условий в системе. Пример безусловной технологической уязвимости — возможность исполнения на компьютере созданного файла под учетной записью интерактивного пользователя. *Условная технологическая уязвимость* возникает в системе лишь при возникновении неких дополнительных условий (без этих условий она отсутствует в системе, а ее угроза непосредственно связана с угрозой выявления соответствующей уязвимости реализации), например, выявленной соответствующей ошибки программирования в системном, либо в прикладном средстве, что уже является угрозой реализации (угрозой реализации соответствующей технологической уязви-

мости). Пример условной технологической уязвимости — возможность исполнения на компьютере созданного файла с системными правами. Естественно, что в штатном режиме функционирования (без возникновения соответствующих условий), это не является технологической уязвимостью — какой-либо ошибкой, либо некорректностью реализации защиты, однако при выявлении соответствующих ошибок программирования в системных средствах (рассмотрим далее), эта возможность уже может рассматриваться в качестве технологической уязвимости.

Отметим, что к условным технологическим уязвимостям (не являющимся уязвимостями без возникновения соответствующих условий) могут быть отнесены и широко используемые на практике штатные возможности современных приложений, позволяющих расширять их с использованием макросов, скриптов и т. д. Угроза данных уязвимостей связана с возможностью надления соответствующего приложения вредоносным кодом.

Таким образом, в общем случае задача реализации защиты информационной системы (задача построения системы защиты) сводится к выявлению и нивелированию угроз безусловных и условных технологических уязвимостей. Нивелирование угроз безусловных технологических уязвимостей в том числе, может решать система защиты посредством их перевода в категорию условных, например, безусловная угроза возможности исполнения на компьютере интерактивным пользователем созданного файла переводится в категорию условных с использованием антивирусного средства защиты — угроза становится реальной при условии, если вредоносная программа, загруженная на компьютер, не детектируется соответствующим антивирусным средством защиты. Естественно, что нивелирование выявленных уязвимостей реализации (в частности, исправление ошибок в программных средствах) никак не связано с задачей системы защиты — это задача разработчика соответствующих программных средств. Отметим, что реализация защиты должна рассматриваться в отношении одной (либо в отношении их необходимых совокупностей) задач защиты информации — обеспечение конфиденциальности (защита от хищения) информации, обеспечение целостности (защита от несанкционированной модификации) и доступности информации, поскольку в общем случае угрозы уязвимостей для них различны.

Под потенциальной угрозой для информационной системы понимаем угрозу, возникновение которой потенциально возможно, под реальной угрозой — реально возникшую угрозу — в системе создаются условия для реализации атаки.

Акцентируем внимание на следующем важном моменте, который должен быть учтен при последующем моделировании. Здесь и далее, говоря об уязвимости реализации и об угрозе уязвимости реализа-

ции, в общем случае мы понимаем некую совокупность однотипных уязвимостей, создающих одни и те же условия для реализации атаки на информационную систему. Например, выявляемые ошибки в системных программных средствах, позволяющие запустить программу с системными правами, мы рассматриваем как одну уязвимость. Вместе с тем для реализации атаки на повышение привилегий можно использовать выявленные уязвимости реализации (ошибки программирования) в различных компонентах системы, работающих в режиме ядра, например, для ОС семейства Windows — это драйвер подсистемы Windows (Win32k.sys), системные драйверы (KM drives) и ядро *ntoskrnl* (NTOS) [6].

С точки зрения последующего моделирования важным является необходимость учета того, что в общем случае в системе одновременно может присутствовать несколько выявленных и не устраненных однотипных уязвимостей реализации.

В отношении выявляемых и исправляемых уязвимостей реализации постоянно ведется соответствующая статистика и аналитическая обработка в целях предоставления пользователям оперативной информации о выявленных уязвимостях и об уровне их критичности. На сегодняшний день наиболее широкое практическое использование нашли следующие способы классификации и количественной оценки актуальности уязвимостей: схема классификации уязвимостей NIPC; шкала анализа уязвимостей SANS; система оценки критичности уязвимостей Microsoft; система оценки уязвимостей по стандарту PCI DSS; системы US—CERT, CVSS и nCircle. Они различаются учитываемыми при классификации уязвимостей параметрами и шкалами оценки уязвимостей.

В качестве примера рассмотрим Общую систему оценки уязвимости (Common Vulnerability Scoring System, CVSS) [7]. Данная система предназначена для классификации уязвимостей по шкале критичности от 0 до 10:

- 0,0—3,9 — низкая степень;
- 4,0—6,9 — средняя степень;
- 7,0—9,9 — высокая степень;
- 10 — критическая степень.

Оценка (отнесение к уровню критичности) уязвимости проводится на основе набора показателей (вектор доступа, сложность доступа, аутентификация, влияние на конфиденциальность, влияние на целостность, влияние на доступность).

Информации в открытых источниках о выявляемых и устраняемых уязвимостях реализации достаточно много. Используя данную статистику, можно определить соответствующие стохастические параметры угрозы уязвимости — интенсивность возникновения (выявления) λ и интенсивность устранения μ , и построить соответствующую математическую модель, позволяющую определять вероятность готовности информационной системы к

безопасной эксплуатации в отношении угрозы уязвимости $P_{0y} = f(\lambda, \mu)$. Данная характеристика угрозы уязвимости может позиционироваться в качестве количественной оценки ее актуальности.

Заметим, что моделирование имеет смысл проводить в отношении угроз уязвимостей реализации, поскольку для угрозы безусловной технологической уязвимости $P_{0y} = 0$ (не являясь безусловными уязвимостями, а порой, создаваемые штатными возможностями программных средств, приводящих при определенных условиях к уязвимостям, на практике подобные уязвимости разработчиками не исправляются).

С учетом сказанного уровень защищенности информационной системы может быть классифицирован следующим образом. Будем говорить, что информационная система *не защищена* при наличии в ней хотя бы одной известной безусловной технологической уязвимости, позволяющей реализовать несанкционированный доступ в целях (либо с совокупностью целей, в зависимости от решаемых задач защиты) нарушения конфиденциальности, целостности, доступности информации. Система имеет *базовый уровень защищенности*, если в ней отсутствуют известные безусловные технологические уязвимости, гипотетически идеально защищена при отсутствии в ней как безусловных, так и потенциально возможных условных технологических уязвимостей. Естественно, что какое-либо проектирование системы защиты для информационной системы имеет смысл применительно к системе базового уровня защищенности, а оптимальность получаемого решения оценивается исходя из защиты от актуальных угроз атак, требующих для реализации использования условных технологических уязвимостей. Защита (дополнительно к базовой) реализуется посредством нивелирования проектируемой системой защиты актуальных угроз условных технологических уязвимостей. В результате проектирования необходимо обеспечить требуемый (актуальный) уровень безопасности защищаемой информационной системы.

Соответственно, моделирование угроз уязвимостей имеет смысл и будет далее проводиться в целях оценки актуальности, именно в отношении угроз уязвимостей реализации.

Замечание. Уязвимости реализации по своей сути разнородны. Некоторые из них при выявлении не создают реальной угрозы до тех пор, пока нарушителем не предпринято соответствующих действий, позволяющих реализовать эту угрозу, например, не создано программного средства, позволяющего осуществить атаку на выявленную уязвимость (эксплойта). Сложность использования уязвимостей соответствующего типа можно учесть, определяя параметр λ только для той части уязвимостей, которые реально эксплуатировались — для которых были разработаны и использовались эксплойты (та-

кая статистика также ведется). Например, в 2013 г., несмотря на множество выявленных, эксплуатировалась лишь одна уязвимость драйвера Win32k.sys.

В отношении угрозы уязвимости (здесь и далее, если нет соответствующего уточнения, говорим об угрозе уязвимости реализации) информационную систему необходимо рассматривать как систему с отказами и восстановлениями характеристики безопасности. Отказом здесь выступает выявление, а восстановлением — устранение выявленной уязвимости.

В теории надежности для моделирования систем с отказами и восстановлениями характеристики надежности (в данном случае ремонта) объектов, как правило, используется аппарат марковских случайных процессов при допущениях о пуассоновском характере потока заявок и о показательном распределении времени обслуживания [8]. Как известно, процесс, протекающий в физической системе, называется марковским (или процессом без последствия), если для каждого момента времени вероятность любого состояния системы в будущем зависит только от состояния системы в настоящий момент и не зависит от того, каким образом система пришла в это состояние. Проанализируем, можно ли использовать (корректно ли использование, а если корректно, то как могут интерпретироваться получаемые результаты) данный аппарат в нашем случае, т. е. для моделирования систем с отказами и восстановлениями, но уже характеристики безопасности.

С этой целью проанализируем, что собой представляют уязвимости, выявление которых в системе создает реальную угрозу атаки. Как отмечали ранее, возникновение уязвимости в информационной системе в общем случае может быть вызвано двумя причинами: отсутствием, либо некорректностью решения соответствующей задачи защиты, либо ошибками реализации средств информационной системы, например, ошибками программирования, которые могут эксплуатироваться нарушителем для обхода защиты. В качестве эксплуатационных параметров уязвимости рассматриваем интенсивность возникновения уязвимости λ , и интенсивность устранения уязвимости μ . Под возникновением уязвимости (здесь и далее) естественно понимаем ее выявление нарушителем.

Предполагая, что система содержит конечное (пусть и очень большое) число не выявленных уязвимостей, можем заключить, что в данном случае процесс не является марковским, поскольку выявление и устранение каждой уязвимости приводит к уменьшению их числа на конечном исходном множестве, т. е. имеем процесс с последствием, при этом входной поток не будет являться пуассоновским, поскольку в этих предположениях $\lambda \neq \text{const}$. Однако оценим, как будут на практике изменяться параметры уязвимости в процессе эксплуатации

информационной системы. Очевидно, что в общем случае интенсивность возникновения уязвимости λ по прошествии некоторого времени будет снижаться, поскольку в первую очередь нарушитель будет выявлять наиболее простые недочеты реализации защиты и ошибки в программном обеспечении (увеличение сложности выявления уязвимости естественно приведет к снижению интенсивности λ). В отношении же параметра μ можем сказать, что он никак не связан со сложностью выявления уязвимости, определяется исключительно типом уязвимости (например, ошибки в системных драйверах и в приложениях требуют различной трудоемкости исправления), т. е. для каждого типа уязвимости можем принять $\mu = \text{const}$.

Теперь допустим, что мы спроектировали систему защиты, применив формальную экстраполяцию (прогнозная экстраполяция здесь мало применима, ввиду высокой интенсивности переходов на новые версии программных средств в современных информационных системах) с использованием марковской модели. Тем самым при моделировании мы предположили, что поток без последствия, т. е. интенсивности возникновения уязвимости λ , и устранения уязвимости μ будут неизменными в процессе последующей эксплуатации защищенной информационной системы. Очевидно, что с учетом сказанного ранее (а именно, что значение λ будет только уменьшаться, а μ останется неизменным в процессе последующей эксплуатации системы), используя подобную модель мы найдем граничные (при худших для системы условиях) значения требуемых характеристик, учет которых гарантирует, что "хуже не будет". На самом же деле, определение значений именно таких характеристик при проектировании системы защиты и требуется (не можем же мы проектировать систему защиты, оперируя заниженными, с учетом их уменьшения в процессе эксплуатации системы, значениями параметров уязвимости). Вот если бы последствие приводило к увеличению λ в процессе эксплуатации информационной системы, тогда, другое дело, подобное последствие при моделировании необходимо было бы учитывать.

Из сказанного можем сделать крайне важный вывод о том, что при моделировании угрозы уязвимости, угрозы атаки и угрозы безопасности информационной системы в целом, поскольку все угрозы (и угрозы атаки и угрозы безопасности информационной системы в целом) в конечном счете создаются угрозами уязвимостей, можно использовать марковские модели, позволяющие в данном случае определять граничные значения характеристик безопасности, которые и необходимо использовать при проектировании системы защиты. Это существенно упрощает рассматриваемую задачу моделирования.

Теперь в двух словах о прогнозировании. Используя рассмотренный подход к моделированию с использованием марковских моделей, характеристики угроз уязвимости мы оцениваем за некоторый прошедший интервал времени, в то время как система защиты проектируется для использования в будущем.

В методическом плане основным инструментом любого прогноза является схема экстраполяции [9]. Сущность экстраполяции заключается в изучении сложившихся в прошлом и настоящем устойчивых тенденций развития объекта прогноза и в переносе их на будущее.

Различают формальную и прогнозную экстраполяцию. Формальная экстраполяция базируется на предположении о сохранении в будущем прошлых и настоящих тенденций развития объекта прогноза; при прогнозной экстраполяции фактическое развитие увязывается с гипотезами о динамике исследуемого процесса с учетом изменений влияния различных факторов в перспективе.

При моделировании, задавая параметры угроз безопасности, мы пользуемся формальной экстраполяцией, исходя из прогноза о том, что в будущем, по причинам, рассмотренным выше, они хуже не станут.

С учетом того, что вероятностью одномоментного появления в системе нескольких однотипных уязвимостей (не одновременного присутствия, о чем говорили ранее) можем пренебречь, процесс возникновения и устранения в системе угрозы уязвимости может быть описан схемой "гибели и размножения" [10]. Тогда для случая одного обслуживаемого прибора искомая характеристика безопасности (стационарный коэффициент готовности, в данном случае, готовности к безопасной эксплуатации в отношении угрозы уязвимости) определяется следующим образом:

$$P_{0y} = 1 - \rho, \text{ где } \rho = \lambda/\mu,$$

а вероятность наличия в системе одновременно R неустранимых уязвимостей $P_{Ry} = \rho^R(1 - \rho)$.

В качестве обслуживаемого прибора в нашем случае выступает коллектив разработчиков, устраняющих выявленную в системе уязвимость с интенсивностью μ . На практике одновременно может устраняться несколько уязвимостей, т. е. в общем случае следует рассматривать схему "гибели и размножения" с C обслуживаемыми приборами. Для такой модели искомая характеристика определяется следующим образом:

$$P_{0y} = \left(1 + \rho + \frac{\rho^2}{2!} + \dots + \frac{\rho^C}{C!}\right)^{-1},$$

а вероятность наличия в системе одновременно R неустранимых уязвимостей $P_{Ry} = \frac{\rho^R}{C!} P_{0y}$.

Можно предположить, что при условии $\rho = \frac{\lambda}{\mu} \ll 1$

значение вероятности $P_{R > 1y}$ мало и им можно пренебречь, что далее упростит построение модели. Возможность упрощения модели имеет смысл исследовать, ввиду того, что нам далее потребуется построить модели угрозы атаки и модели угрозы безопасности информационной системы в целом, в которых угроза уязвимости выступает простейшим элементом безопасности. Оценим данное условие, для чего рассмотрим изменение на интересующих нас интервалах значений характеристик P_{Ry} от изменения значений характеристики ρ для одноканальной (табл. 1) и двухканальной ($C = 2$) (табл. 2) систем.

Проанализировав результаты, представленные в табл. 1 и в табл. 2, можем сделать следующие выводы. Если $\rho \leq 0,2$ при моделировании угрозы уязвимости можно использовать одноканальную схему "гибели и размножения", если $\rho > 0,2$ необходимо использовать двухканальную схему.

Замечание. Условие $\rho > 0,9$ для угрозы уязвимости реализации не анализируется, поскольку при выполнении данного условия можно принять $P_{0y} = 1$, отнеся соответствующую условную технологическую уязвимость к безусловным, а информационную систему — к незащищенным.

К слову сказать, приведенные в табл. 2 результаты исследования позволяют оценить современное положение дел в области информационной безопасности. Так, при средней продолжительности устранения уязвимостей, составляющей 30 дней (некоторые уязвимости в современных ОС устраняются месяцами), выявление всего лишь шести уязвимостей в год обеспечивает значение коэффициента готовности, близкое к 0,5.

Таблица 1

Характеристики одноканальной системы

P_{Ry}	ρ				
	0,1	0,2	0,3	0,4	0,5
P_{0y}	0,90	0,80	0,70	0,60	0,50
P_{1y}	0,09	0,16	0,21	0,24	0,25
$P_{R \geq 2y}$	0,01	0,04	0,09	0,16	0,25

Таблица 2

Характеристики двухканальной системы

P_{Ry}	ρ						
	0,3	0,4	0,5	0,6	0,7	0,8	0,9
P_{0y}	0,74	0,68	0,60	0,56	0,51	0,47	0,43
P_{1y}	0,23	0,27	0,32	0,34	0,36	0,38	0,39
P_{2y}	0,03	0,05	0,08	0,10	0,13	0,15	0,18
$P_{R \geq 3y}$	0	0	0	0	0	0	0

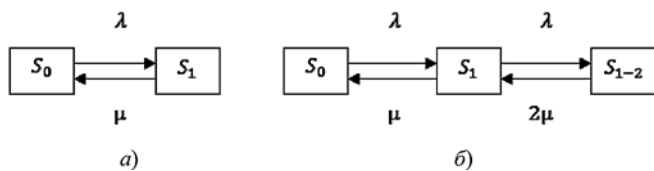


Рис. 1. Графы системы состояний случайного процесса для угрозы уязвимости:
а — при $\rho \leq 0,2$; б — при $\rho > 0,2$

Графы состояний случайного процесса выявления и устранения уязвимостей (марковского процесса с дискретными состояниями и непрерывным временем), которые нами далее будут использованы, представлены на рис. 1, где S_0 — исходное состояние системы, S_1 — в системе выявлена и не устранена одна из уязвимостей, S_{1-2} — в системе выявлены и не устранены две уязвимости.

2. Интерпретация угрозы атаки на информационную систему

В общем случае реализация атаки предполагает использование нескольких присутствующих в системе уязвимостей, причем, как правило, в определенной последовательности. В работе [1] угрозу атаки на информационную систему было предложено представлять соответствующим орграфом, проиллюстрированным на рис. 2, а, где через P_{0yr} , $r = 1, \dots, R$, обозначается вероятность отсутствия в системе r -й уязвимости (информационная система готова к безопасной эксплуатации в отношении угрозы r -й уязвимости) — одной из R угроз уязвимостей, последовательно (дуги графа определяют последовательность использования выявленных уязвимостей при реализации атаки) используемых атакой на информационную систему.

При подобном представлении угроза атаки на информационную систему может интерпретироваться схемой параллельного резервирования угроз уязвимостей (в общем случае безусловных и условных технологических уязвимостей и уязвимостей реализации), резервируемыми и резервирующими элементами которой являются угрозы уязвимости

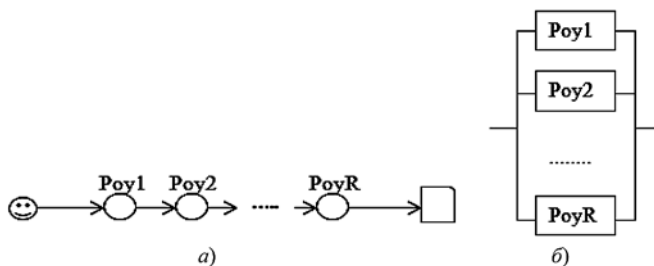


Рис. 2. Орграф угрозы атаки и ее интерпретация схемой параллельного резервирования угроз уязвимостей:
а — орграф угрозы атаки; б — схема параллельного резервирования

(рис. 2, б), поскольку каждая угроза уязвимости, присутствующая в системе с вероятностью P_{0yr} , может рассматриваться в качестве резервирующего элемента (с вероятностью P_{0yr} предотвращает атаку).

Замечание. Для угроз технологических уязвимостей как для безусловных, так и для условных имеем $P_{0yr} = 1$, для угроз уязвимостей реализации значения характеристики P_{0yr} необходимо рассчитывать с использованием представленных выше моделей.

Исключив из орграфа угрозы атаки вершины угроз технологических уязвимостей, для которых $P_{0yr} = 1$, получим орграф угроз уязвимостей реализации и можем определить характеристику — вероятность того, что информационная система готова к безопасной эксплуатации в отношении угрозы атаки P_{0a} (очевидно, что события выявления и устранения уязвимостей реализации, эксплуатируемых атакой, можно рассматривать в качестве независимых), в предположении, что все R угроз уязвимостей, оставшихся на приведенном орграфе, — это угрозы уязвимостей реализации [1]:

$$P_{0a} = 1 - \prod_{r=1}^R (1 - P_{0yr}).$$

Данная характеристика угрозы атаки может позиционироваться в качестве количественной оценки ее актуальности [1] для реализации защиты.

Данная интерпретация позволяет представлять систему защиты информационной системы (СЗИ) в виде отдельной вершины (отдельных вершин) на орграфе угрозы атаки с параметрами $\lambda_{СЗИ}$ и $\mu_{СЗИ}$ уже собственно системы защиты (это параметры угроз уязвимостей системы защиты, куда включены и безусловные, и условные угрозы технологических уязвимостей). При этом можно сделать вывод о том, что угрозы технологических уязвимостей, создающие угрозу атаки, с точки зрения их нивелирования системой защиты, эквивалентны [1], поскольку при нивелировании любой из них угрозы системы защиты включаются в схему параллельного резерва (см. рис. 1, б) одинаково, с параметрами безопасности системы защиты: $\lambda_{СЗИ}$ и $\mu_{СЗИ}$.

Если обозначить вероятность того, что система защиты, используемая для нивелирования одной из угроз технологических уязвимостей, создающих угрозу атаки, готова к безопасной эксплуатации, через $P_{0СЗИ}$, то вероятность того, что защищенная в отношении угрозы уязвимости информационная система будет готова к безопасной эксплуатации ($P_{0узис}$) при использовании системы защиты, нивелирующей эту уязвимость, может быть определена следующим образом:

$$P_{0узис} = 1 - (1 - P_{0y})(1 - P_{0СЗИ}).$$

Вероятность того, что защищенная информационная система готова к безопасной эксплуатации ($P_{0аизс}$) в отношении угрозы атаки, при использовании сис-

темы защиты, нивелирующей одну из угроз технологических уязвимостей, создающих угрозу атаки,

$$\text{равна } P_{0\text{АЗИС}} = 1 - (1 - P_{0\text{СЗИ}})^R \prod_{r=1}^R (1 - P_{0\text{УР}}).$$

В двух словах остановимся на вопросах нивелирования угрозы технологической уязвимости системой защиты. Например, возможность, в том числе и с системными правами, исполнения в системе создаваемых в процессе работы интерактивными пользователями файлов, можно рассматривать в качестве технологической уязвимости системы. Сегодня на практике широко распространены решения, не предполагающие нивелирования (устранения) угроз технологических уязвимостей, предполагающие перевод угроз безусловных технологических уязвимостей в группу угроз условных уязвимостей за счет создания системой защиты дополнительных условий возможности их реализации. К подобным решениям можно, например, отнести антивирусные средства защиты, основанные на анализе контролируемых событий на соответствие неким эталонным множествам (например, сигнатурный и/или поведенческий анализ). Поскольку данные эталонные множества априори не могут быть полными, не может быть полностью (здесь не рассматриваются угрозы уязвимости реализации) и нивелирована угроза безусловной технологической уязвимости запуска создаваемых исполняемых файлов. В результате этого значение параметра $\lambda_{\text{СЗИ}}$ подобных систем защиты растет по мере увеличения интенсивности создаваемых вредоносных программ (функционально зависит от интенсивности создания новых вредоносных программ). По самым оптимистичным прогнозам современными антивирусными средствами защиты выявляется — детектируется, до 75 % новых вредоносных программ (например, в отношении новых технологий детектирования, опирающихся на возможности KSN- (технология Kaspersky Security Network, основанная на создании антивирусного облака), утверждается, что их применение позволило увеличить с 60 до 75 % долю угроз, обнаруживаемых эвристическими методами без обновления классических антивирусных баз, т. е. 25 % угроз средством защиты не детектируются, а интенсивность создания новых вредоносных программ определяется десятками [11], если уже не сотнями миллионов в год. С учетом сказанного несложно рассчитать $\lambda_{\text{СЗИ}}$ подобных современных систем защиты. Задавая различные расчетные значения $\mu_{\text{СЗИ}}$ (а это, по крайней мере, единицы, а то и десятки дней, ведь мы говорим о тех вредоносных программах, которые не выявляются антивирусным средством — их еще нужно каким-то образом обнаружить), можно легко определить сколько обслуживающих приборов (в данном случае одновременно работающих над различными сигнатурами вирусных аналитиков С) потребуется

антивирусной компании для выполнения требования к стационарности системы: $\frac{\lambda_{\text{СЗИ}}}{C \mu_{\text{СЗИ}}} < 1$. К слову

сказать, из этого анализа можем сделать вывод о том, что основными характеристиками эффективности антивирусной системы защиты является не интенсивность выявления сигнатуры вируса $\mu_{\text{СЗИ}}$, а средняя длина очереди заявок на обслуживание (выявленных вредоносных программ) и среднее время пребывания заявки на обслуживание в очереди. Отдельно стоит задача оценки выявления вредоносной программы, не детектируемой антивирусным средством защиты. Отметим, что для моделирования подобных систем марковские процессы уже не применимы, поскольку интенсивность создания новых вредоносных программ (что и не удивительно при такой эффективности защиты) растет из года в год, причем крайне стремительно [11].

Оценим потенциальные возможности реализации эффективной защиты с использованием систем защиты, нивелирующих угрозы технологических уязвимостей, основанных на реализации разграничительной политики доступа к ресурсам [12], в частности, эффективное решение защиты от вредоносных программ, призванное нивелировать угрозу соответствующей уязвимости, рассмотрено в работе [13]. С этой целью получим и проанализируем значения характеристики $P_{0\text{У}}$, средства защиты $P_{0\text{УСЗИ}}$ при различных значениях $\lambda_{\text{СЗИ}}$ и $\mu_{\text{СЗИ}}$ (поскольку нас интересует условие $\rho \leq 0,2$, то для расчетов используем модель, приведенную на рис. 1, а). Например, $P_{0\text{УСЗИ}} = 0,99$ достигается при выявлении в средстве защиты только одной уязвимости в год при продолжительности ее устранения 3,65 дня. При выявлении двух уязвимостей в год при той же продолжительности их устранения, уже имеем $P_{0\text{УСЗИ}} = 0,98$, этот же результат получим при выявлении в средстве защиты одной уязвимости в год при продолжительности ее устранения, составляющей 7,3 дня.

Отметим, что соответствующие требования к характеристикам системы защиты могут задаваться (и на практике задаются) в задании на их техническое сопровождение, их выполнение при этом легко проконтролировать. К слову сказать, из практического опыта можем заключить, что, естественно, без каких-либо расчетов, интуитивно на практике задаются требования к значению характеристики системы защиты $P_{0\text{УСЗИ}}$ не ниже 0,9 (если это, конечно, не системы антивирусной защиты).

Из проведенного исследования можно сделать вывод о том, что потенциально высокое значение (0,9 и выше) коэффициента готовности системы защиты к безопасной эксплуатации систем защиты, решающих задачу нивелирования угроз технологических уязвимостей, достижимо.

Акцентируем внимание на следующем принципиальном моменте. Методы контроля и разграничения прав доступа, используемые для реализации разграничительной политики доступа к ресурсам, составляющие основу практически любой современной системы защиты информации (не говорим здесь о методах, основанных на анализе контролируемых событий на соответствие соответствующим эталонным множествам), служат на практике для разделения между пользователями компьютерных ресурсов с учетом решаемых ими задач в информационной системе.

В данной работе сформулирована принципиально иная постановка задачи защиты, для решения которой могут и, как мы выше показали, должны применяться методы контроля и разграничения прав доступа, — задача нивелирования безусловных и условных технологических уязвимостей, решаемая в целях защиты от актуальных угроз атак для повышения уровня безопасности информационной системы. Естественно, что данная задача защиты, иная собственно в своей постановке, требует и принципиально иных решений, в том числе технических, для реализации разграничительной политики доступа к ресурсам, создаваемой уже совершенно с иными целями. Требования к методам контроля и разграничения прав доступа, используемым для решения данной задачи защиты, реализация которых позволяет построить безопасную систему, собственно методы и их практическая реализация на примере апробированных технических решений, разграничительные политики доступа, используемые для защиты от наиболее актуальных современных угроз атак, рассмотрены в работе [12].

3. Пример количественной оценки актуальности угрозы уязвимости и угрозы атаки

В качестве примера рассмотрим угрозу атаки на повышение привилегий. Для исследования она интересна тем, что экспертная оценка ее актуальности при проектировании системы защиты совсем не очевидна. Подобная угроза атаки предполагает внедрение на компьютер вредоносной программы, что можно рассматривать как безусловную технологическую уязвимость, использование выявленной уязвимости реализации (выявленной программной ошибки) в компоненте (программе) ядра ОС, запущенного с системными правами, для исполнения внедренного на компьютер в процессе работы исполняемого вредоносного файла, что можно рассматривать уже в качестве условной технологической уязвимости системы (возможно при выявлении соответствующей ошибки в системном средстве), с системными правами. Получаем оргграф угрозы атаки, предполагающей последовательное использование нарушителем данных трех уязвимостей (оргграф угрозы атаки содержит в своем составе три взвешенных вершины). Если возможность

загрузки на компьютер исполняемого файла и возможность исполнения созданного в процессе работы системы файла, как отмечали, можно отнести к уязвимостям технологического характера (они всегда присутствуют в незащищенной системе) — для них $P_{0yr} = 1$, то интересующую нас угрозу уязвимости реализации — уязвимости, возникающей в результате выявления программных ошибок в компонентах (программах) ядра ОС, можно промоделировать с использованием соответствующей статистики.

Заметим, что чтобы дать экспертную оценку актуальности угрозы данной атаки, можно, например, воспользоваться исследованиями, представленными в работах [14, 15], результаты которых приведены на рис. 3 и 4 (см. четвертую сторону обложки).

Как дать экспертную (хотя бы качественную, не говоря уже о количественной) оценку актуальности угрозы рассматриваемой атаки, исходя из подобной статистики, а уж тем более, как ее обосновать? Насколько адекватной будет подобная оценка?

Построим математическую модель исследуемой угрозы уязвимости реализации, для чего обратимся к соответствующей статистике, приведенной в работе [16]. Выявленные и исправленные за 2014 г. уязвимости уровня ядра ОС Windows сведены в табл. 3.

Как видим из табл. 3, уязвимостей, позволяющих осуществить атаку на повышение привилегий (Elevation of Privilege) за 2014 г. было выявлено достаточно много. Однако уязвимости уровня ядра ОС сложны в практическом использовании для реализации соответствующей атаки, так как требуют разработки соответствующего эксплойта. Уязвимости, для которых были созданы и использованы эксплойты, в табл. 3 подчеркнуты. Что же касается угрозы атак на повышение привилегий, то за 2014 г. были созданы эксплойты к следующим трем уязвимостям: CVE-2014-6324, CVE-2014-4113, CVE-2014-0318 (примем для расчетов значение интенсивности выявления угрозы интересующей нас уязвимости — 3 в год).

Исходя из того, что среднее время устранения выявленных уязвимостей в ОС Windows составляет около месяца [17] (примем значение интенсивности устранения выявленной уязвимости — 12 в год), можем оценить загрузку соответствующей системы массового обслуживания, $\rho = 0,25$. С учетом сказанного ранее получаем количественную оценку актуальности угрозы рассматриваемой уязвимости $P_{0y} = 0,75$, используя которую, можно утверждать, что в любой момент времени вероятность того, что в системе угроза уязвимости реализации, позволяющая реализовать атаку на повышение привилегий, реальна (уязвимость присутствует), составляет 25 %. Заметим, что таким же значением будет характеризоваться и уровень актуальности угрозы атаки на повышение привилегий, поскольку для

Исправленные уязвимости для различных компонентов Windows

Компонент	Обновление	Тип	Уязвимость
Windows UMC (VBScript, Direct2D, MSXML, DirectShow, SAMR, File Handling/ kernel32.dll, Shell handler/ shell32.dll, Remote Desktop, Journal, On-Screen Keyboard, Media center/mcplayer. dll, Installer, Task Scheduler, OLE, Message Queuing, Schannel, Kerberos, Audio Service, IIS, IME (Japanese), GDI+/gdi32.dll, RPC/rpcrt4.dll, Graphics/ windowscodecs.dll	MS14-011, MS14-007, MS14-005, MS14-013, MS14-016, MS14-027, MS14-030, MS14-033, MS14-038, MS14-039, MS14-041, MS14-043, MS14-049, MS14-054, MS14-060, MS14-062, MS14-064, MS14-066, MS14-067, MS14-068, MS14-071, MS14-074, MS14-076, MS14-078, MS14-036, MS14-047, MS14-084, MS14-085	Remote Code Execution(11), Information Disclosure(3), Security Feature Bypass(4), Elevation of Privilege(9), Tampering(1)	CVE-2014-0271, CVE-2014-0263, CVE-2014-0266 , CVE-2014-0301, CVE-2014-0317, CVE-2014-0315, CVE-2014-1807 , CVE-2014-1816, CVE-2014-0296, CVE-2014-1824, CVE-2014-2781, CVE-2014-2780, CVE-2014-4060, CVE-2014-1814, CVE-2014-4074, CVE-2014-4114, CVE-2014-4971, CVE-2014-6332, CVE-2014-6352 , CVE-2014-6321, CVE-2014-4118, CVE-2014-6324 , CVE-2014-6322, CVE-2014-6318, CVE-2014-4078, CVE-2014-4077, CVE-2014-1818, CVE-2014-0316, CVE-2014-6363, CVE-2014-6355
Win32k	MS14-003, MS14-015, MS14-045, MS14-058, MS14-079	Elevation of Privilege(4), Denial of Service(1)	CVE-2014-0262, CVE-2014-0300, CVE-2014-0323, CVE-2014-0318, CVE-2014-1819, CVE-2014-4113 , CVE-2014-4148 , CVE-2014-6317
KM drivers (ndproxy.sys, tcpip.sys, afd.sys, fastfat.sys)	MS14-002, MS14-006, MS14-031, MS14-040, MS14-045, MS14-063, MS14-070	Elevation of Privilege(5), Denial of Service(2)	CVE-2013-5065 , CVE-2014-0254, CVE-2014-1811, CVE-2014-1767, CVE-2014-4064, CVE-2014-4115, CVE-2014-4076
.NET Framework	MS14-009, MS14-026, MS14-046, MS14-053, MS14-057, MS14-072	Elevation of Privilege(3), Security Feature Bypass(1), Denial of Service(1), Remote Code Execution(1)	CVE-2014-0253, CVE-2014-0257, CVE-2014-0295 (ASLR Bypass) , CVE-2014-1806, CVE-2014-4062 (ASLR Bypass), CVE-2014-4072, CVE-2014-4073, CVE-2014-4121, CVE-2014-4122 (ASLR Bypass), CVE-2014-4149

остальных угроз уязвимостей, создающих рассматриваемую угрозу атаки, имеем $P_{0yr} = 1$.

В порядке замечания отметим, что, как видим, по уровню актуальности данная угроза уязвимости сопоставима с актуальностью угрозы условной технологической уязвимости запуска на защищаемом антивирусной системой защиты компьютере (переводит соответствующую угрозу безусловной технологической уязвимости в категорию условных) вредоносной программы.

Для практической оценки актуальности уязвимости более наглядно и на практике целесообразно использование иной характеристики безопасности — среднего времени наработки на отказ безопасности (рассматриваем систему с отказами и восстановлением характеристики безопасности), или среднего интервала времени между выявлениями (средняя продолжительность устранения уязвимости составляет $1/\mu$) в информационной системе уязвимости, $T_{0y} = 1/\lambda$.

Используя данную характеристику безопасности, можем заключить, что в среднем рассматриваемая угроза уязвимости, а соответственно, и угроза атаки на повышение привилегий, будет реальной каждые четыре месяца, причем нарушитель сможет ею воспользоваться в течение месяца.

Исходя из сказанного, применительно к рассматриваемому примеру можно утверждать, что без использования каких-либо дополнительных средств защиты, нивелирующих соответствующую угрозу технологической уязвимости (либо угрозу установки на компьютер исполняемого файла, либо угрозу его исполнения с системными правами), при условии соответствующей заинтересованности нарушителя в осуществлении несанкционированного доступа к обрабатываемой в системе информации (в предположении, что он реализует первую же реальную угрозу данной уязвимости), в среднем, соответствующая успешная атака будет осуществлена в течение 4—5 месяцев эксплуатации системы. А с учетом того, что речь идет о возможности запуска в системе произвольной вредоносной программы с системными правами, то эта атака может быть осуществлена нарушителем практически с любой целью.

Замечание. Решение по реализации эффективной защиты от подобных атак приведено в работе [6].

Вот она реальная количественная оценка актуальности угрозы уязвимости, получение которой не потребовало проведения экспертных оценок. Имея подобную оценку уже несложно сделать объективный вывод о том, стоит ли нивелировать по-

добную угрозу технологической уязвимости системной защиты, актуальна ли она.

Аналогичное исследование можно провести в отношении угроз уязвимостей как в системных средствах, так и в приложениях. Например, для браузера Internet Explorer за 2014 г. выявлены десятки уязвимостей, подавляющая часть из которых позволяет удаленно выполнить вредоносный код (Remote Code Execution), для семи из них были созданы соответствующие эксплойты [16].

Решение по реализации эффективной защиты от подобных атак приведено в работе [18].

Отметим, что при построении современных систем защиты информации, одной из наиболее актуальных современных задач, что наглядно иллюстрирует проведенное в работе исследование, является задача нивелирования актуальных угроз технологических уязвимостей как в системных средствах, так и в приложениях, в том числе связанных с их программной реализацией. Естественно, что для решения новой в своей постановке задачи защиты информации необходимы и новые технические решения. Так, например, подходы к защите, описанные в работах [6, 18], основаны на реализации запатентованного авторами апробированного технического решения [19].

Заключение

В данной работе исследовались вопросы моделирования и оценки актуальности угроз уязвимостей. При моделировании же угрозы атаки практический интерес представляет не только построение модели с отказами и восстановлениями характеристики безопасности, причем с учетом того, что угроза атаки создается не одной, а в общем случае некоторой совокупностью разнородных угроз уязвимостей, т. е. модели, позволяющей получать количественные оценки характеристик возникновения и устранения в системе реальной угрозы атаки, но и построение модели с фатальным отказом, предполагающей уже непосредственно реализацию нарушителем реальной угрозы атаки на конкретную информационную систему (реализацию несанкционированного доступа), для которой проектируется система защиты. С этой целью уже необходимо оценить (опять же без применения экспертных оценок в модели нарушителя) готовность реализации атаки определенной сложности нарушителем на конкретную информационную систему, характеризующую меру его заинтересованно-

сти в осуществлении подобной атаки, что в том числе предполагает количественную оценку сложности реализации атаки. Разработанные модели угрозы атаки и нарушителя будут рассмотрены во второй части статьи.

Список литературы

1. **Щеглов К. А., Щеглов А. Ю.** Математические модели эксплуатационной информационной безопасности // Вопросы защиты информации. 2014. Вып. 106, № 3. С. 52—65.
2. **Росенко А. П.** Внутренние угрозы безопасности конфиденциальной информации: Методология и теоретическое исследование. М.: Красанд, 2010.
3. **Корт С. С.** Теоретические основы защиты информации: учеб. пособие. М.: Гелиос АРВ, 2004.
4. **Корченко А. Г.** Построение систем защиты информации на нечетких множествах. Теория и практические решения. Киев: МК-Пресс, 2006.
5. **Белов Е. Б., Лось В. П., Мещеряков Р. В., Шелупанов А. А.** Основы информационной безопасности. М.: Горячая линия — Телеком, 2006.
6. **Щеглов К. А., Щеглов А. Ю.** Защита от атак на повышение привилегий // Вестник компьютерных и информационных технологий. 2015. № 1. С. 36—42.
7. **Полное** руководство по общему стандарту оценки уязвимостей версии 2. Часть первая. Группы метрик [Электронный ресурс]. URL: <http://www.securitylab.ru/analytics/355336.php>, свободный (10.01.2015).
8. **Половко А. М., Гуров С. В.** Основы теории надежности. СПб.: БХВ-Петербург. 2006. 704 с.
9. **Саркисян В. А., Каспин В. И., Лисичкин В. А., Минаев Э. С., Пасенчик Г. С.** Теория прогнозирования и принятия решений. М.: Высшая школа. 1977.
10. **Саати Т.** Элементы теории массового обслуживания и ее приложения. М.: Советское радио, 1965.
11. **Эксперты:** трояны по-прежнему остаются самым популярным вредоносным ПО [Электронный ресурс]. URL: http://www.itsec.ru/newstext.php?news_id=100297, свободный (10.01.2015).
12. **Щеглов А. Ю.** Модели, методы и средства контроля доступа к ресурсам вычислительных систем: учеб. пособие. СПб.: Изд-во университета ИТМО, 2014. 95 с.
13. **Щеглов К. А., Щеглов А. Ю.** Система защиты от запуска вредоносных программ // Вестник компьютерных и информационных технологий. 2013. № 5. С. 38—43.
14. **Отчет** по уязвимостям 20.02—26.02 [Электронный ресурс]. URL: <http://www.securitylab.ru/vulnerability/reports/420676.php>, свободный (10.01.2015).
15. **Отчет** по уязвимостям за второй квартал 2008 года [Электронный ресурс]. URL: <http://www.securitylab.ru/analytics/358113.php>, свободный (10.01.2015).
16. **Windows Exploitation** 2014. [Электронный ресурс] // URL: <http://www.welivesecurity.com/wp-content/uploads/2015/01/Windows-Exploitation-in-2014.pdf>, свободный (10.01.2015).
17. **"Критические дни":** Linux, Mac OS X, Solaris and Windows. [Электронный ресурс]. URL: <http://www.securitylab.ru/opinion/297876.php>, свободный (10.01.2015).
18. **Щеглов К. А., Щеглов А. Ю.** Технология изолированной обработки данных критичными приложениями // Вопросы защиты информации. 2015. Вып. 108. № 1. С. 15—22.
19. **Щеглов А. Ю., Щеглов К. А.** Система контроля доступа к файлам на основе их автоматической разметки. Патент на изобретение № 2524566. Приоритет от 18.03.2013, опубл. 27.07.2014, Бюл. № 21.

Informational System Attack Threat Modeling and Interpretation. Part 1. Vulnerability Threat Modeling and Attack Threat Interpretation

We introduce vulnerability threats classification as a basic informational security element (modeling object) in borders of which threats are divided to technological and implementation ones (which create threats of technological vulnerabilities). We show that implementation vulnerability threats must be researched as vulnerability threats modeling object (for attack modeling). While this modeling can be done without any expert assessments, i.e. just by solely using existing statistics about security parameters of vulnerabilities threats. We substantiate Markov models with discrete states and continuous time (widely used in reliability theory for modeling of systems failures and recoveries) usage correctness to model attack threats. However, we show the necessity of inclusion into vulnerability threat model of "birth and death" scheme elements (because of same time similar vulnerability possibility which create the same threat of attack). This produces fundamentally differences from the modeling problems of reliability theory. We suggest vulnerability threat serial reservation scheme as an attack threat interpretation (which defines informational system attack threat modeling approach with use of built queuing models). We show an example of vulnerability and attack threats quantitative relevance assessment (produced by single implementation vulnerability threat) illustrating possibilities of the proposed modeling approach.

Keywords: informational system, security, vulnerability threat, attack threat, threat relevance, Markov model, queuing system, failure, recovery, security characteristics

References

1. Shcheglov K. A., Shcheglov A. Ju. Matematicheskie modeli jekspluatacionnoj informacionnoj bezopasnosti, *Voprosy zashhity informacii*, 2014, Issue 106, no. 3, pp. 52–65.
2. Rosenko A. P. *Vnutrennie ugrozy bezopasnosti konfidencial'noj informacii: Metodologija i teoreticheskoe issledovanie*, Moscow: Krasand, 2010.
3. Kort S. S. *Teoreticheskie osnovy zashhity informacii: Uchebnoe posobie*, Moscow, Gelios ARV, 2004.
4. Korchenko A. G. Postroenie sistem zashhity informacii na nechetkih mnozhestvah. Teorija i prakticheskie reshenija, Kiev, MK-Press, 2006.
5. Belov E. B., Los V. P., Meshheriakov R. V., Shelupanov A. A., *Osnovy informacionnoj bezopasnosti*, Moscow, Gorjachaja linija — Telekom, 2006.
6. Shcheglov K. A., Shcheglov A. Ju. Zashhita ot atak na povyshenie privilegij, *Vestnik komp'juternyh i informacionnyh tehnologij*, 2015, no. 1, pp. 36–42.
7. *Polnoe rukovodstvo po obshhemu standartu ocenki uязvimostej versii 2*. Chast' pervaja. Gruppy metrik [Elektronnyj resurs]. URL: <http://www.securitylab.ru/analytics/355336.php>. svobodnyj (10.01.2015).
8. Polovko A. M., Gurov S. V. *Osnovy teorii nadezhnosti*. — SPb.: BHV-Peterburg. — 2006.
9. Sarkisjan V. A., Kaspin V. I., Lisichkin V. A., Minaev Je. S., Pasenchnik G. S. *Teorija prognozirovaniya i prinjatija reshenij*, Moscow, Vysshaja shkola, 1977.
10. Saati T. *Jelementy teorii massovogo obsluzhivaniya i ee prilozhenija*, Moscow, Sovetskoe radio, 1965.
11. Jeksperty: trojany po-prezhnemu ostajutsja samym popularnym vredonosnym PO [Elektrorihij resurs]. Rezhim dostupa: http://www.itsec.ru/newstext.php?news_id=100297, svobodnyj (10.01.2015).
12. Shcheglov A. Ju. *Modeli, metody i sredstva kontrolja dostupa k resursam vychislitel'nyh sistem*, Uchebnoe posobie, Sankt-Peterburg: Universitet ITMO, 2014.
13. Shcheglov K. A., Shcheglov A. Ju. Sistema zashhity ot zapuska vredonosnyh programm, *Vestnik komp'juternyh i informacionnyh tehnologij*, 2013, no. 5, pp. 38–43.
14. *Otchet po uязvimostjam* 20.02–26.02 [Elektronnyj resurs], URL: <http://www.securitylab.ru/vulnerability/reports/420676.php>, svobodnyj (10.01.2015).
15. *Otchet po uязvimostjam za vtoroj kvartal 2008 goda* [Elektronnyj resurs], URL: <http://www.securitylab.ru/analytics/358113.php>, svobodnyj (10.01.2015).
16. *Windows Exploitation* in 2014. [Elektronnyj resurs], URL: <http://www.welivesecurity.com/wp-content/uploads/2015/01/Windows-Exploitation-in-2014.pdf>, svobodnyj (10.01.2015).
17. *"Kriticheskie dni": Linux, Mac OS X, Solaris and Windows*, [Elektronnyj resurs], URL: <http://www.securitylab.ru/opinion/297876.php>, svobodnyj (10.01.2015).
18. Shcheglov K. A., Shcheglov A. Ju. Tehnologija izolirovannoj obrabotki dannyh kritichnymi prilozhenijami, *Voprosy zashhity informacii*, 2015, Issue 108, no. 1, pp. 15–22.
19. Shcheglov A. Ju., Shcheglov K. A. Sistema kontrolja dostupa k fajlam na osnove ih avtomaticheskoy razmetki. *Patent na izobretenie* № 2524566. Prioritet ot 18.03.2013, opubl. 27.07.2014, Bjul. № 21.

ЖУРНАЛ В ЖУРНАЛЕ



НЕЙРОСЕТЕВЫЕ ТЕХНОЛОГИИ

№ 12

ДЕКАБРЬ

2015

Главный редактор:

ГАЛУШКИН А. И.

Редакционная коллегия:

АВЕДЬЯН Э. Д.
БАЗИЯН Б. Х.
БЕНЕВОЛЕНСКИЙ С. Б.
БОРИСОВ В. В.
ГОРБАЧЕНКО В. И.
ЖДАНОВ А. А.
ЗЕФИРОВ Н. С.
ЗОЗУЛЯ Ю. И.
КРИЖИЖАНОВСКИЙ Б. В.
КУДРЯВЦЕВ В. Б.
КУЛИК С. Д.
КУРАВСКИЙ Л. С.
РЕДЬКО В. Г.
РУДИНСКИЙ А. В.
СИМОРОВ С. Н.
ФЕДУЛОВ А. С.
ЧЕРВЯКОВ Н. И.

**Иностранные
члены редколлегии:**

БОЯНОВ К.
ВЕЛИЧКОВСКИЙ Б. М.
ГРАБАРЧУК В.
РУТКОВСКИЙ Л.

Редакция:

БЕЗМЕНОВА М. Ю.
ГРИГОРИН-РЯБОВА Е. В.
ЛЫСЕНКО А. В.
ЧУГУНОВА А. В.

Галушкин А. И.

Нейрочипы и нейроморфные ЭВМ: проблемы моделирования 942

Горбатков С. А., Касимова Л. И.

Нейросетевой итерационный метод построения динамических моделей вероятности риска банкротства при неполных данных 949

Нейрочипы и нейроморфные ЭВМ: проблемы моделирования

Представлен комплекс моделей, необходимых для разработки современных и перспективных нейрочипов и нейроморфных ЭВМ на их основе. В состав комплекса моделей входят модели, связанные с разработкой нейросетевых алгоритмов решения задач, технологические модели, связанные с построением базовых элементов схмотехники, а также схмотехнические модели принципиальных схем, реализующих нейрочипы и их функциональные элементы.

Ключевые слова: нейронные сети, настройка коэффициентов, элементы нейрочипов, схмотехнические модели

Введение

Сложность современных и перспективных нейрочипов и нейроморфных ЭВМ делает необходимым разработку комплекса моделей для проверки их основных свойств и функционирования. Сочетание в современных и перспективных нейрочипах и нейроморфных ЭВМ различных технологий (мемристорной; аналоговой КМОП; цифровой КМОП; оптической, а также различных технологий изготовления нейроплат, нейроблоков, нейростоек и супернейрокомпьютеров) делает необходимым разработку и использование комплекса моделей, систем проектирования и библиотек элементов, а именно моделей:

- входного сигнала нейронных сетей;
- функционалов первичной оптимизации;
- нейронных сетей;
- процессов настройки коэффициентов нейронных сетей в нейроморфных ЭВМ;
- процессов настройки коэффициентов нейронной сети с переменной структурой;
- процессов настройки коэффициентов нейронных сетей в нейроморфных ЭВМ с ограничениями на настраиваемые коэффициенты;
- типовых входных сигналов нейронных сетей;
- планов экспериментов и визуализации результатов для проверки качества работы нейронных сетей;
- решения формализуемых и неформализуемых задач с применением больших нейронных сетей для перспективных нейроморфных ЭВМ;
- распараллеливания нейросетевых алгоритмов в соответствии с реальной структурой физических моделей перспективных нейроморфных ЭВМ;
- технологических;
- архитектуры нейроморфных ЭВМ;
- схмотехнических для нейроморфных ЭВМ;
- нейрофизиологических, для исследования на нейроморфных ЭВМ.

В статье рассмотрены эти модели.

1. Модели входного сигнала нейронных сетей

Основная аксиоматика нейронных сетей основана на вероятностной модели внешнего мира.

Внешний мир представлен в виде набора случайных сигналов. Практический опыт с применением классических методов математической статистики показал их основные ограничения — априорную необходимость формального описания функций распределения входных многомерных случайных сигналов. В реальных задачах начала — середины 60-х годов прошлого века была реальная необходимость в отказе от подобного, характерного для методов математической статистики, представления функций распределения случайных сигналов в виде известных функций. Считалось, что функция распределения неизвестна и может быть очень сложной, многомодальной.

Канонически входной сигнал нейронных сетей может быть представлен как входной сигнал в виде непосредственно случайного процесса и как указание учителя нейронной сети, формирующего отношение каждого входного образа (элемента случайного процесса) к тому или иному классу. Указания учителя могут быть двоичными и вещественными, одномерными и многомерными.

Постановка задачи обработки таких сигналов подобна задаче математической статистики. Рассматривались в основном задачи классификации (распознавания) и задачи кластеризации (распознавание с учителем, имеющим нулевую квалификацию).

С развитием теории нейронных сетей и ее применений при решении ряда практических проблем в 60-е годы прошлого века появились следующие модификации данных задач:

- обучение с учителем, имеющим конечную квалификацию, когда учитель не может с единичной вероятностью (полной уверенностью) указать принадлежность образов к тому или иному классу (например, в задачах медицинской диагностики);
- режим "вредительства", когда учитель заведомо ложно или приблизительно ложно сообщает нейронной сети информацию о принадлежности входных образов к тому или иному классу.

Количественной характеристикой входного сигнала нейронной сети является также априорная вероятность появления образов того или иного класса. Например, при решении задачи оптического рас-

познавания букв печатного текста необходимо учитывать, что априори известна вероятность появления каждой буквы в тексте. Эта вероятность может быть получена обработкой текста достаточно большого объема.

2. Модели функционалов первичной оптимизации в системах обработки информации с применением нейросетевых технологий

Функционал первичной оптимизации определяет основную цель, достигаемую при работе нейронной сети. Вероятностная модель внешнего мира и рассмотренные выше модели входного сигнала делают необходимым рассмотрение критериев первичной оптимизации, связанных с понятием средней функции риска, известной из классической математической статистики.

Простейшим таким критерием является критерий минимума средней функции риска, который при равных априорных вероятностях появления классов и при диагональной матрице потерь [1] преобразуется в более простой — критерий минимума среднеквадратичной ошибки.

На практике формируются различные модели функционала первичной оптимизации:

- функционал, связанный с критерием минимума средней функции риска;
- функционал для критерия минимума средней функции риска при заданном значении условной функции риска для каждого из классов;
- функционал для критерия минимума средней функции риска при равных значениях условной функции риска для различных классов;
- другие возможные на практике функционалы.

Рассмотренные модели функционала первичной оптимизации формируются, в свою очередь, своеобразно для указанных выше моделей входного сигнала в режимах классификации, кластеризации, обучения с учителем, имеющего конечную квалификацию, вредительства и других, имеющих место на практике.

При построении моделей функционала первичной оптимизации необходимо в них в обязательном порядке учитывать свойства матрицы потерь. Как правило, она по умолчанию принимается симметричной. При этом при решении задачи распознавания двух классов образов потери при принятии образов первого класса за образы второго класса считаются равными потерям при принятии образов второго класса за образы первого. Такая постановка является некорректной во многих практических задачах, например, в системах обнаружения мин с помощью геолокатора, задачах медицинской диагностики и др. Именно это требует учета недиагональности матрицы потерь при построении модели функционала первичной оптимизации в системах обработки информации с применением нейросетевых технологий и на последующих этапах построения моделей для нейроморфных ЭВМ.

3. Модели нейронных сетей

Буквально через несколько лет после выхода основополагающей книги Ф. Розенблатта "Принципы нейродинамики" [5] в России сформировалась научная школа в области теории нейронных сетей [1—4]. Именно в России, еще в 60-е годы прошлого столетия рассматривались в качестве канонических многослойные нейронные сети:

- с полными последовательными связями (настраиваемые связи от каждого предыдущего слоя нейронов к последующему);
- с полными обратными связями (с выхода последующих слоев нейронов на входы предыдущих);
- с полными перекрестными связями (с выхода предыдущих на входы всех последующих, включая полные последовательные).

Одним из важных моментов построения моделей нейронных сетей является выбор типа функций активации первого, последующих и, что особенно важно, последнего слоя нейронов.

На протяжении последних 50 лет основной идеей формирования структуры нейронной сети, решающей ту или иную задачу, была идея формирования структуры, адекватной решаемой задаче, с возможностью изменения некоторых параметров структуры в целях оптимизации выбора варианта с точки зрения качества решения задачи.

На протяжении последних 50 лет для моделирования нейронных сетей использовались наиболее производительные ЭВМ и при соответствующем росте требований к размерам и быстродействию нейронных сетей всегда не хватало производительности моделирующих ЭВМ и памяти для хранения массивов перестраиваемых коэффициентов. Последний такой эксперимент был проведен с суперЭВМ IBM BlueGene.

Вполне естественным в этом плане выглядит формируемая в последние годы идея так называемого глубокого (глубинного) обучения (*deep learning*), в которой делается попытка введения в структуру нейронных сетей ограниченных связей, специфических для данной решаемой задачи, с естественным увеличением числа слоев нейронной сети. Причем основное внимание на первом этапе развития глубокого обучения уделяется задаче обработки изображений.

По сути, эта идея не нова и родилась вовсе не в 2006—2009 гг. Еще в своей классической монографии [5] Фрэнк Розенблатт предложил для сокращения числа настраиваемых связей ретины размером $N \times N$ с первым слоем из H_1 нейронов использовать случайным образом формируемые $A(H_1)$ связи (причем $A(H_1) \ll N$) каждого нейрона первого слоя с элементами ретины размером $N \times N$. Тогда число настраиваемых коэффициентов сокращается с $N \times N \times H_1$ до $N \times A(H_1) \times H_1$.

В 60-е годы прошлого столетия это была достаточно эффективная методика уменьшения числа настраиваемых связей при решении задач обработки,

в частности распознавания, изображений. В работе [6] известного ученого — профессора университета Беркли — отмечено, что идеи глубокого обучения во многом повторяют работы 60-х и 80-х годов прошлого века в области многослойных нейронных сетей.

На вопрос корреспондента: "Из Вашей статьи я сделал вывод, будто Вы считаете, что в темах глубинного обучения, больших данных, машинного зрения и им подобных, присутствует огромное количество дезинформации" Майкл Джордан ответил следующее: "В частности тематика глубинного обучения — это, в значительной степени, просто переработанная информация о нейронных сетях с 1980-х годов. А в те годы они повторяли то, что было известно в 1960-е годы, такое чувство, что каждые 20 лет проходит волна, затрагивающая одни и те же темы. В волне, которая идет сейчас, главной идеей является сверточная нейронная сеть, о которой уже говорили 20 лет назад.

Одна из проблем, связанных с обеими предыдущими волнами, и которая упорно продолжит присутствовать в нынешней волне, заключается в том, что люди продолжают считать, будто секрет как-то связан с неврологией. Они считают, что для глубинного обучения нужно понимание того, как мозг обрабатывает информацию, учится, принимает решения или справляется с огромным количеством информации. Это откровенная ложь".

4. Модели процессов настройки коэффициентов нейронных сетей в нейроморфных ЭВМ

С самого начала работ в 60-е годы прошлого столетия при построении моделей процессов настройки коэффициентов нейронной сети использовались результаты работ советской научной школы под руководством д-ра техн. наук, профессора В. В. Солодовникова в области так называемых аналитических самонастраивающихся систем [7]. В этих системах градиент функционала оптимизации процесса функционирования вычисляется не введением искусственных поисковых колебаний и детектированием выходных сигналов, а путем специальной обработки промежуточных (внутренних) и выходных сигналов. При этом в работах [1, 3, 4] нейронная сеть рассматривается как частный случай объекта управления в аналитической самонастраивающейся системе. Данный объект является многомерным, многосвязным и нелинейным. Именно в этом была отмечена основная специфика нейронных сетей как объектов управления, определившая в том числе и специфику построения моделей процессов настройки коэффициентов нейронных сетей на нейроморфных ЭВМ.

Первым этапом построения моделей процессов настройки коэффициентов нейронных сетей является формирование функционала *вторичной оптимизации* с помощью внутренних и выходных сигналов нейронных сетей той или иной структуры. Вид

функционала вторичной оптимизации в значительной степени зависит от решаемой задачи. В задачах аппроксимации и экстраполяции функций и их модификациях (распознавания образов), как правило, применяется простейший вид функционала вторичной оптимизации — среднеквадратическая ошибка, вычисляемая на некотором интервале функционирования системы.

Вторым этапом построения моделей процессов настройки коэффициентов нейронных сетей является построение моделей (разработка алгоритмов) поиска экстремума функционала вторичной оптимизации в нейронных сетях. Причем основная проблема заключается в том, что данный функционал является многомерным (размерность — число настраиваемых коэффициентов нейронной сети) и многоэкстремальным с достаточно большим числом локальных экстремумов, в том числе и глобального.

В зависимости от решаемой задачи здесь возможны различные варианты постановки задачи:

- найти глобальный экстремум функционала вторичной оптимизации;
- найти все локальные и глобальные экстремумы функционала вторичной оптимизации;
- найти любой локальный экстремум функционала вторичной оптимизации, как это имеет место при решении больших систем линейных неравенств [8].

Третьим этапом построения моделей процессов настройки коэффициентов нейронных сетей является выбор коэффициента усиления в алгоритме поиска экстремума. В большинстве случаев он выбирается эмпирически при анализе процесса сходимости процедуры обучения. Иногда прибегают к некоторым модификациям, выбирая этот коэффициент уменьшающимся во времени в процессе настройки, что зачастую приводит к резкому увеличению времени настройки. К подобным модификациям относится известный метод стохастической аппроксимации.

Четвертым этапом построения модели процесса настройки коэффициентов нейронной сети является выбор размера памяти фильтра обработки потока ошибок в контуре настройки. Как правило, эта память по умолчанию выбирается равной единице, что не всегда является оптимальным с точки зрения результирующего времени настройки, а, во-вторых, справедливо только для случаев стационарных входных сигналов (образов) на входе (сигналов с априори постоянной во времени сложной функцией распределения). Отсутствие или малая память фильтра сигнала ошибки может привести к внесению дополнительных шумов в контур настройки коэффициентов нейронной сети.

Пятым этапом построения моделей процессов настройки коэффициентов нейронной сети является выбор начальных значений настраиваемых коэффициентов. Как правило, во многих работах выбор начальных условий проводится случайным

образом. Это осуществляется по умолчанию либо вследствие слабой профессиональной подготовки разработчика и приводит к резкому увеличению как времени настройки, так и требований к вычислительным ресурсам.

Этот один из основных недостатков моделей нейронных сетей с фиксированной до этапа настройки структурой можно ликвидировать, выбирая начальные условия настройки для каждой задачи. Еще в работе [1] были представлены такие подходы к решению задач классификации и кластеризации, а в [4] — дополнительно для задач нейроуправления. Качественный, адекватный решаемой задаче, выбор начальных условий настройки коэффициентов нейронных сетей является важным условием резкого уменьшения времени настройки для нахождения необходимых локальных и глобальных экстремумов функционала вторичной оптимизации.

5. Модели процессов настройки коэффициентов нейронной сети с переменной структурой

Отказавшись на первом этапе построения моделей и алгоритмов настройки нейронных сетей от информации о параметрах фиксированной структуры, возможно для настройки коэффициентов использовать модели с переменной структурой, предложенные в работе [2] и успешно применяемые при решении практических задач в течение многих лет. Модели настройки нейронных сетей с переменной структурой реализуют последовательную итерационную процедуру наращивания числа нейронов первого слоя сети до достижения необходимого качества решения задачи. Последующие за первым слоем нейронной сети реализуют функцию в пространстве выходных сигналов нейронов первого слоя. Модель с переменной структурой может быть использована также для выбора начальных условий при дальнейшей подстройке коэффициентов нейронной сети с фиксированной структурой.

6. Модели процессов настройки коэффициентов нейронных сетей в нейроморфных ЭВМ с ограничениями на настраиваемые коэффициенты

Вполне естественным является желание учета в математической модели управляемого объекта ограничений физической модели в целях их максимального соответствия.

Именно с этой целью в 60-е годы прошлого века, в рамках развиваемой в СССР научной школы нейроморфных ЭВМ, были разработаны модели настройки нейронных сетей с учетом реальных ограничений на настраиваемые коэффициенты, присутствовавшие в физических реализациях нейроморфных ЭВМ. Были разработаны [1] модели настройки коэффициентов нейронных сетей для нескольких видов ограничений с возможностью расширения методики на возможные перспективные ограничения. Основными, применяемыми на практике, моделями

настройки были модели для ограничений на коэффициенты типа насыщения, так как в реальных разработанных физических моделях настраиваемые коэффициенты реализовывались на потенциометрах [9].

В перспективных нейроморфных ЭВМ с применением мемристоров [10] необходимо разработать модели настройки коэффициентов нейронных сетей с учетом реальных нелинейных передаточных функций мемристоров как элементов физической реализации настраиваемых коэффициентов. При этом нейроморфный компьютер будет многоуровневой аналогово-цифровой ЭВМ с распределением нейросетевого алгоритма решения задачи по различным уровням, включая аналоговый, реализуемый на мемристорах. Именно поэтому разработка моделей мемристоров и мемристорных матриц (см. п. 11) представляет особый интерес.

7. Модели типовых входных сигналов нейронных сетей

Как и в классической теории управления, типовые входные сигналы необходимы как для проверки качества функционирования разработанной системы, так и для регулярного тестирования (проверки правильности функционирования) в процессе эксплуатации. Для линейных одномерных систем такими типовыми входными сигналами являются дельта-функция, единичная ступенька, линейный сигнал, сигналы большего порядка, которыми проверяется астатизм системы и проводится проверка работы в процессе функционирования. Для линейных многомерных систем и нелинейных объектов, как правило, типовые входные сигналы формируются специфическим образом для каждой решаемой задачи. То же самое делается для нейронных сетей, которые являются многомерными многосвязными нелинейными динамическими системами. В большинстве работ по нейросетевым алгоритмам решения задач реализуется некоторое, достаточно субъективное, разделение обучающей выборки на обучающую и контрольную, по результатам работы на которых делаются выводы о качестве работы системы.

Стремление к повышению объективности сравнения работы таких сложных систем, которыми являются нейронные сети, привело еще в 60—70-е годы прошлого века к формированию типовых тестовых входных сигналов, специфических для данной решаемой задачи [1]. Для задач классификации и кластеризации это были многомерные (N) случайные выборки с изменяемым числом мод (M) и среднеквадратичным отклонением (δ) для каждой моды. Подобные типовые сигналы, в принципе, позволяют достаточно объективно сравнивать разрабатываемые алгоритмы классификации и кластеризации, варьируя параметры в пространстве $[M, \delta]$. Были сделаны попытки [1, 4] сформировать типовые входные сигналы для задач экстраполяции функций и нейроуправления сложными нелинейными динамическими системами.

Иногда в реальных практических задачах, а также в случае применения методов разделения выборки на обучающую и контрольную выборка бывает недостаточно большой, а иногда весьма малой, чтобы нейронная сеть путем обработки восприняла ее статистические свойства. В этом случае применяется процедура рандомизации конечной выборки, когда элементы конечной выборки считаются математическим ожиданием случайных выборок, искусственно генерируемых добавлением к ним случайной выборки с нормальным распределением и некоторым среднеквадратичным отклонением δ_k . При этом предметом исследования разработанных нейросетевых алгоритмов решения задач является исследование зависимости качества решения от величины δ_k .

8. Модели планов экспериментов и визуализации результатов для проверки качества работы нейронных сетей

План эксперимента по проверке качества работы нейронных сетей на типовых и реальных входных сигналах разрабатывается так, чтобы максимально покрыть исследуемое пространство параметров входного сигнала и структуры нейронной сети. Диапазон изменения этих параметров выбирается с учетом того, что сложность решаемой задачи в процессе настройки косвенно отображается в сложности настроенной нейронной сети.

Построение модели визуализации результатов настройки нейронных сетей является важным результатом работы. В настоящее время основным методом построения моделей визуализации результатов настройки нейронных сетей на типовых и реальных входных сигналах является представление зависимости от времени в процессе настройки значения функционала вторичной оптимизации и, в частности, среднеквадратичной ошибки решения задачи.

Главная трудность решения задачи визуализации результатов настройки нейронных сетей заключается в многомерности пространства входных сигналов и указаний учителя. Для демонстрации эффективности моделей и алгоритмов настройки иногда, начиная с 60-х годов прошлого века, применялись 2D-модели, а в последнее время и 3D-модели, но они являются весьма косвенным, начальным аргументом эффективности. Основным же аргументом эффективности остается отмеченная выше зависимость в процессе настройки функционала вторичной оптимизации.

9. Модели решения формализуемых и неформализуемых задач с применением больших нейронных сетей для перспективных нейроморфных ЭВМ

Раздел разработки моделей решения формализуемых задач является прямым развитием работ, начатых во всем мире в конце 80-х годов прошлого

века, впервые интегрированных в коллективной монографии [8], которые посвящены решению классических математических задач повышенной сложности в нейросетевом логическом базисе. Если формализуемые задачи требуют перехода в нейросетевой логический базис при достижении и превышении определенного уровня сложности, то для неформализуемых задач нейросетевые технологии являются основным, если не единственным, методом решения.

Основная задача здесь должна заключаться в возможности использования программно-аппаратных эмуляторов с большим и очень большим числом нейронов и связей и в оптимизации нейросетевых алгоритмов решения формализуемых и неформализованных задач под многоуровневую архитектуру перспективных аналого-цифровых нейроморфных ЭВМ с применением мемристоров.

10. Модели распараллеливания нейросетевых алгоритмов в соответствии с реальной структурой физических моделей перспективных нейроморфных ЭВМ

Проблема распараллеливания нейросетевых алгоритмов стояла в течение последних десятилетий развития вычислительной техники. Она была актуальной для малопроцессорных ЭВМ, многопроцессорных, в частности транспьютерных [11], систем, для суперЭВМ на базе графических процессоров. Каждый раз с изменением архитектуры ЭВМ модели и методы распараллеливания нейросетевых алгоритмов менялись.

Важно отметить, что распараллеливание на логические элементы физической модели необходимо проводить не только для настраиваемой нейронной сети, но и для алгоритма, реализующего контур настройки коэффициентов.

В случае нейрокомпьютеров с применением мемристоров модель распараллеливания нейросетевых алгоритмов будет специфической вследствие особенностей архитектуры, содержащей несколько слоев обработки информации:

- внешний слой аналоговой обработки системой на базе мемристоров;
- слой обработки на базе аналоговых КМОП-схем, стыкованный с мемристормым блоком;
- слой ЦАП и АЦП;
- слой цифровой обработки на базе цифровых КМОП-схем (ПЛИС, GPU и др.);
- слой узловых ЭВМ, соединяемых в некоторый кластер;
- хост ЭВМ.

11. Технологические модели

В состав технологических моделей, необходимых для построения перспективных нейроморфных ЭВМ должны входить, в первую очередь, модели мемристоров и мемристормых матриц большой

размерности, адекватные архитектуре разрабатываемых нейрочипов с применением мемристоров. Данные модели должны разрабатываться в тесном контакте с технологиями на базе ПЭВМ, ПЛИС и GPU и использоваться для отработки технологических процессов, а в будущем — для разработки моделей и алгоритмов настройки нейронных сетей в нейроморфных ЭВМ.

В состав технологических моделей должны быть включены модели анализа и выбора технологий изготовления мемристоров с точки зрения производительности будущих вычислительных систем с применением мемристоров и, непосредственно, методики оценки производительности совокупной вычислительной системы.

Ряд технологических моделей должен быть уточнен разработчиками, работающими с технологическими процессами изготовления мемристоров и мемристормных матриц.

Необходимым разделом технологических моделей должны явиться модели памяти и коммутационных систем на базе мемристоров как намного более массовых изделий в производстве в будущем и, по сути дела, обеспечивающих жизнеспособность мемристорного направления развития нейроморфных ЭВМ.

12. Модели архитектуры нейроморфных ЭВМ

До этапа разработки схемотехники нейроморфных ЭВМ с применением мемристоров необходимо пройти этап разработки и моделирования архитектуры нейроморфных ЭВМ.

Нижним уровнем подобных работ является создание модели архитектуры перспективных СБИС-нейрочипов с применением мемристоров. Основными принципами построения подобных СБИС по сравнению с классическими микропроцессорами являются следующие:

- реализация нейросетевого логического базиса на программно-аппаратном уровне обработки информации, т. е. переход от последовательных к высокопараллельным архитектурам обработки информации [9];
- переход в нейросетевой программно-аппаратной реализации от представления сигналов в виде уровней токов и напряжений к представлению сигналов в виде частоты последовательности узких импульсов, что приводит к резкому (в десятки и сотни тысяч раз) понижению энергопотребления в реальных разработках даже без применения мемристоров (цифровых и аналоговых). Этот важный шаг в развитии вычислительной техники не должен сейчас приводить к спекуляциям типа "мы создаем модели мозга", так как принцип функционирования мозга и технологии его реализации принципиально отличны от того, что предлагается создать в нейроморфных ЭВМ, в том числе с применением мемристоров [6];

- переход от чисто цифровой к аналого-цифровой программно-аппаратной реализации нейроморфных ЭВМ вследствие технологических особенностей мемристоров и мемристормных матриц, что приводит к резкому повышению технической производительности при соответствующем контроле точности обработки. Этот переход идет сейчас, на этапе создания специализированных нейроморфных ЭВМ без применения мемристоров, для того чтобы сформировать достаточно широкую сферу применения [12].

Архитектура перспективных СБИС-нейрочипов с применением мемристоров должна включать в себя кроме элементов слоев обработки, перечисленных выше в п. 10, также и коммутационные среду и интерфейс, специализированные на реализацию нейросетевых алгоритмов.

При разработке моделей нейроплат также следует учитывать необходимость размещения на плате нейрочипов, элементов памяти, специализированной для платы коммутационной среды и интерфейсов.

По аналогичным принципам необходимо строить модели нейроблоков путем соединения нескольких моделей нейроплат, модели нейростоек — путем соединения нескольких моделей нейроблоков.

Необходимо отметить, что в настоящее время активно ведутся разработки ряда специализированных нейроморфных систем с реализацией указанных выше трех принципов. В некоторых разработках реализуются только первые два принципа. Однако во всех случаях это приводит к достаточно большому эффекту. Именно поэтому при разработке модели архитектуры специализированных нейроморфных ЭВМ с соблюдением всех трех принципов реализации необходимо создание следующих моделей:

- архитектуры перспективного аналогового кардиостимулятора с резким понижением энергопотребления на базе нейрочипа с частотно-импульсным представлением обрабатываемых сигналов [12];
- архитектуры перспективного кохлеаимпланта;
- модели архитектуры перспективного зрительного импланта;
- архитектуры перспективного интерфейса "мозг-компьютер";
- архитектуры перспективной видеокамеры с применением мемристоров [13].

Наверняка перечень специализированных нейроморфных ЭВМ будет в дальнейшем расширяться в целях формирования конкретных разработок и подготовки рынка будущих систем с применением мемристоров.

13. Схемотехнические модели нейроморфных ЭВМ

Все представленные схемотехнические модели нейроморфных ЭВМ можно условно разделить на две части:

- схемотехнические модели сегодняшнего дня, которые можно использовать при создании

нейроморфных ЭВМ до появления мемристормых систем в серийном производстве;

- схмотехнические модели нейроморфных ЭВМ с применением мемристоров.

Основная задача разработчиков схмотехнических моделей нейроморфных ЭВМ заключается в том, чтобы разработать схмотехнические модели сегодняшнего дня, которые в будущем будут максимально использованы при разработке схмотехнических моделей нейроморфных ЭВМ с применением мемристоров.

В состав схмотехнических моделей нейроморфных ЭВМ должны входить следующие модели:

- интеграции технологий изготовления мемристормых матриц большого размера в КМОП-схемы;
- аналогового фрагмента схмотехники перспективного СБИС-нейрочипа с применением мемристоров, включая ЦАП и АЦП;
- цифрового фрагмента перспективного СБИС-нейрочипа с применением мемристоров;
- схмотехники СБИС-нейрочипа с применением мемристоров с использованием библиотеки конкретного производства;
- схмотехники нейроплат, нейроблоков, нейростоек.

В состав схмотехнических моделей кроме вышеперечисленных должны входить схмотехнические модели специализированных нейроморфных ЭВМ:

- кардиостимулятора;
- кохлеаимпланта;
- интерфейса "мозг-компьютер";
- зрительного импланта;
- перспективной видеокамеры с применением мемристоров.

14. Нейрофизиологические модели для исследования на нейроморфных ЭВМ

Нейроморфные ЭВМ являются наиболее адекватным средством построения нейрофизиологических моделей различных разделов мозга. Однако исследование нейрофизиологических моделей разделов мозга не являются основной задачей, определяющей необходимость разработки нейроморфных ЭВМ, так как не являются задачей массового применения. Основными задачами здесь является широкий круг практических задач обработки сигналов, изображений, управления динамическими объектами, в первую очередь роботами наземного, воздушного, морского, подводного базирования. Не умаляя значимости создания и исследования нейрофизиологических моделей на базе будущих нейроморфных ЭВМ, необходимо проводить эти работы для различных разделов мозга. При этом разработчики нейроморфных ЭВМ будут участвовать в этой работе в качестве реализаторов, подобно тому, как это делалось в течение последних десятилетий, в том числе на последних вариантах суперЭВМ: транспьютер-

ных и транспьютероподобных, на базе графических процессоров и классических кластерных. Каждый раз оказывалось, что реально достигнутая производительность суперЭВМ крайне далека от требуемой для построения нейрофизиологических моделей в реальном времени. В этом плане разрабатываемые и перспективные нейроморфные ЭВМ в ближайшие годы могут, по нашему мнению, только слегка раздвинуть границы исследований нейрофизиологических моделей разделов мозга.

Заключение

Данная статья посвящена одной из многих трудных проблем, стоящих перед разработчиками нейроморфных ЭВМ — разработке систем моделей для современных и перспективных нейрочипов и нейроморфных ЭВМ. Автор ставил своей целью показать необходимость органического единства разработок моделей различного уровня и опасается, что текущий уровень организации науки и промышленных разработок может привести к раздробленности рассматриваемого раздела работ. Представленная структура разработки системы моделей является основой для будущей многолетней работы в области создания нейроморфных ЭВМ. Основой этой работы должен быть научно-технический задел советской и российской науки в области нейросетевых технологий, о приоритете которой было указано в предисловиях к монографиям [3, 4] ведущих мировых ученых Лотфи Заде, Роберта Хехт-Нильсена и Амары.

Список литературы

1. **Галушкин А. И.** Синтез многослойных систем распознавания образов. М.: Энергия, 1974.
2. **Галушкин А. И.** Многослойные системы распознавания образов. М.: Изд. МИЭМ, 1968.
3. **Galushkin A. I.** Neural networks theory. Springer, 2007.
4. **Галушкин А. И.** Нейронные сети: основы теории. М.: Горячая линия — Телеком, 2010.
5. **Розенблатт Ф.** Принципы нейродинамики. М.: Мир, 1963.
6. **Джордан М.** Искусственный интеллект, большие данные и дезинформация технологий. ICloud.ru
7. **Аналитические** самонастраивающиеся системы / Под ред. В. В. Солодовникова. М.: Машиностроение, 1965.
8. **Нейроматематика.** Сер. "Нейрокомпьютеры и их применение". Кн. 6. М.: ИПРЖР, 2002.
9. **Галушкин А. И.** Нейрокомпьютеры. Сер. "Нейрокомпьютеры и их применение". Кн. 3. М.: ИПРЖР, 2000.
10. **Галушкин А. И.** На пути к нейрокомпьютерам с использованием мемристоров // Информационные технологии. 2014. № 4. Приложение. 32 с.
11. **Галушкин А. И.** Транспьютерные системы — начало становления в России ЭВМ с массовым параллелизмом // Нейрокомпьютеры. 2005. № 3.
12. **Qing Sun and all.** Implementation Study of an Analog Spiking Neural Network for Assisting Cardiac Delay Prediction in Cardiac Resynchronization Therapy Device // IEEE Trans. on Neural Networks. June 2011. Vol. 22, N. 6.
13. **Новый** чип ускорит обработку видео в 1000 раз. Системы видеонаблюдения. Продолжение эволюции. М., 4 декабря 2013 г. URL: <http://www.cnews.ru>, 16.08.2013.

Neurochips and Neuromorphic Computers: a Modelling Challenges

The complex of models needed for the development of current and future neurochips and neuromorphic computers based on them is presented. The complex of models includes models related to the development of neural network algorithms for solving problems, process models, associated with the construction of the basic elements of the circuitry, and circuit design model concepts realizing neurochips and functional elements.

Due to complexity of current and future neurochips and neuromorphic computers, it is necessary, before their creation, to develop their complex models for testing their basic features and functions. The combination in modern and future neurochips and neuromorphic computers such different technologies as memristors, analog CMOS, digital CMOS, optical, various techniques of development of neuroboards, neroblocks, neuroracks and superneurocomputers, makes it necessary to develop and use such complex models, CAD systems and elements libraries.

This article describes these models and give an explanation of the objective necessity of the implementation such software models on personal computers and GPU-supercomputers.

Keywords: neural networks, weights adjustment, neurochips, circuits design models

References

1. Galushkin A. I. *Sintez mnogosloynnykh sistem raspoznavaniya obrazov*. Moscow: Energiya, 1974.
2. Galushkin A. I. *Mnogosloynnyye sistemy raspoznavaniya obrazov*. Moscow: Izd. MIEM, 1968.
3. Galushkin A. I. *Neural networks theory*. Springer, 2007.
4. Galushkin A. I. *Neyronnyye seti: osnovy teorii*. Moscow: Goryachaya liniya — Telekom, 2010.
5. Rozenblatt F. *Printsipy neyrodinamiki*. Moscow: Mir, 1963.
6. Dzhordan M. *Iskusstvennyy intellekt, bol'shiye dannyye i dezin-formatsiya tekhnologiy*. ICloud.ru
7. *Analiticheskiye samonastroyayushchiyesya sistemy*. Pod red. V. V. Solodovnikova. Moscow: Mashinostroyeniye, 1965.
8. *Neyromatematika*. Ser. "Neyrokompyutery i ikh primeneniye". Kn. 6. Moscow: IPRZHR, 2002.
9. Galushkin A. I. *Neyrokompyutery*. Ser. "Neyrokompyutery i ikh primeneniye". Kn. 3. Moscow: IPRZHR, 2000.
10. Galushkin A. I. Na puti k neyrokompyuteram s ispol'zovaniyem memristorov, *Informatsionnyye tekhnologii*, 2014, no. 4, Prilozheniye, 32 p.
11. Galushkin A. I. Transpyuternyye sistemy — nachalo stanovleniya v Rossii EVMs massovym paralelizmom, *Neyrokompyutery*, 2005, no. 3.
12. Qing Sun and all. Implementation Study of an Analog Spiking Neural Network for Assisting Cardiac Delay Prediction in Cardiac Resynchronization Therapy Device, *IEEE Trans. on Neural Networks*. June 2011, vol. 22, no. 6.
13. *Novyy chip uskorit obrabotku video v 1000 raz*. Sistemy videonablyudeniya. Prodolzheniye evolyutsii. Moscow, 4 dekabrya 2013 g. URL: <http://www.cnews.ru>, 16.08.2013.

УДК 338.27

С. А. Горбатков, доктор техн. наук, профессор, e-mail: sgorbatkov@mail.ru,
Финансовый университет при Правительстве РФ, Уфимский филиал, г. Уфа,
Л. И. Касимова, аспирант, клиентский менеджер, e-mail: liankakasimva@yandex.ru,
ОАО "Сбербанк России", Стерлитамакское отд., г. Стерлитамак

Нейросетевой итерационный метод построения динамических моделей вероятности риска банкротства при неполных данных

Предложен нейросетевой итерационный метод оценки вероятности развивающегося во времени риска банкротства экономических объектов применительно к сложным условиям моделирования (неполноты данных, их сильной зашумленности с неизвестным законом распределения шумов). Сделаны оценки сходимости метода для модели, полученной на реальных данных, на основе байесовского подхода к регуляризации модели.

Ключевые слова: диагностирование и прогнозирование банкротств, нейросеть, динамическая модель, логистическая трансформация вероятности

Введение

Математические модели оценки вероятности риска банкротств в настоящее время актуальны в связи со многими приложениями в реальном и финансовом секторах экономики. К ним относятся задачи оценки банкротств корпораций; оценки финансово-экономического положения партнеров в сделках; оценки инвестиционной привлекательности предприятий; обеспечения экономической безопасности предприятий путем постоянного мониторинга их финансового состояния; оценки банком кредитоспособности заемщиков; при выработке управленческих решений по налоговому регулированию (управлению) на местном уровне и во многих других экономических приложениях.

Методы моделирования оценки риска банкротств развивались, начиная с пионерских работ Альтмана (1968 г.) [1] и его последователей, где использовались различные версии линейных методов множественного дискриминантного анализа (MDA). Эти методы оказались практически неприменимыми для российских предприятий вследствие различия в законодательстве (инвестиции на Западе более защищены законодательством, чем в России), различия в стандартах бухгалтерской отчетности, отсутствия глубокой адаптации моделей банкротств для различных стран и отраслей экономики.

В связи с указанными причинами в России были предприняты многочисленные попытки совершенствования MDA — линейных моделей, а также появившихся в 1980 г. логистических моделей банкротств [2]. Бурное развитие в России в последнее 20-летие интеллектуальных информационных технологий, в частности нейросетевых [3], открыло принципиально новые возможности в следующих аспектах:

- учет сложной многофакторной нелинейной связи моделируемого показателя Y с вектором экономических показателей (факторов) анализируемого объекта и окружающей среды:

$$Y = \varphi(\mathbf{x}), \mathbf{x} = (x_1, x_2, \dots, x_j, \dots, x_n), \quad (1)$$

где $\varphi(\cdot)$ — нелинейная функция;

- создание моделей банкротств в условиях сильного зашумления данных, отягченного их дефицитом.

Однако до настоящего времени не исследованы нейросетевые прогнозные динамические модели оценки вероятности риска банкротств, которые представляют наибольший практический интерес. Знание динамики развивающегося во времени процесса банкротства позволяет в каждый момент времени определить стадию банкротства и применить упреждающие управляющие воздействия.

На пути создания динамических моделей риска банкротства предприятий возникают труднопреодолимые препятствия в виде неполноты данных. Суть проблемы заключается в следующем. Пусть имеются данные наблюдения вектора $\mathbf{x}$ в несколь-

ких временных срезах $\{t_k\}$, $k = 1, 2, \dots, N$, т. е. известны значения всех факторов во временных срезах $\{x_{jk}\}$, $j = \overline{1, n}$, $k = \overline{1, N}$. Пусть обучающая выборка D содержит наблюдения для нескольких предприятий $\langle y_{gk}, \mathbf{x}_{gk} \rangle$, $g = \overline{1, G}$, где $\langle, \rangle$ — знак кортежа, g — номер предприятия. При этом в последнем временном срезе ($k = N$), как правило, известны векторы значения факторов $\{\mathbf{x}_{gN}\}$, а также значения эндогенной (выходной) переменной $\{y_{gN}\}$, (метки "банкрот-небанкрот"). Это позволяет оценить вероятность риска банкротства в последнем временном срезе по формуле логистической трансформации риска банкротства:

$$P_{gN} = 1 / [1 + \exp(-\hat{y}_g(\mathbf{x}_{gN}))], \quad (2)$$

где $\hat{y}$ — расчетное значение случайной величины Y в выражении (1).

При этом показатель экспоненты $\hat{y}(\mathbf{x}_{gN})$ в нашем исследовании является нелинейной функцией от вектора факторов $\mathbf{x}_{gN}$ и аппроксимируется нейросетевой моделью [4].

Однако в других временных срезах $\{t_k\}$, $k \neq N$ известны только векторы факторов $\{\mathbf{x}_{gk}\}$, а значения эндогенной переменной $\{y_{gk}\}$ и, соответственно, вероятности $\{P_{gt_k}\}$ неизвестны, поскольку часть предприятий имеет промежуточную (неизвестную) стадию банкротства, которую по статистической модели вида (2), связывающей $\mathbf{x}$ и P , определить не представляется возможным. Причиной этого является деформация связи (2) с течением времени во временных срезах $\{t_k\}$.

В итоге столбцы (либо строки) данных во временных срезах оказываются неполными и обучение нейросетевой динамической модели оценки риска банкротства невозможно.

В данной работе предлагается гибридный нейросетевой итерационный метод, суть которого основана на следующей идее. Будем считать, что нелинейная функциональная связь между $\hat{y}$ и вектором $\mathbf{x}$ в последнем временном срезе $t = N$ приближенно сохраняется ("наследуется") и в других временных срезах $\{t_k\}$, $k \neq N$. При таком допущении эндогенная переменная y_{gk} , $k = 1, 2, \dots, N - 1$ может быть восстановлена поочередно в каждом временном срезе подстановкой вектора $\mathbf{x}_{gk}$ в обученную статическую нейросетевую модель вида (2). После этого данные становятся полными (комплектными), несущими в себе динамическую информацию через функции $\mathbf{x}_g(t_k)$ и $y_g(t_k)$, где временный аргумент t_k , $k = 1, 2, \dots, N - 1$ является дискретным.

Естественно, допущение о сохранении нелинейной зависимости $\hat{y}(\mathbf{x}_{gN})$ вида (1) во всех временных срезах вызывает определенную погрешность в оценке динамики риска банкротства, поэтому для уменьшения этой погрешности вводится специальный итерационный процесс исправления функции $\hat{y}(\mathbf{x}_g(t))$, в итоге которого мы получим

динамическую модель оценки риска банкротства с непрерывным временем $t \in [0; T]$, что и является целью исследования в работе.

1. Постановка задачи

Пусть известны неполные данные $D' \subset D$, где термин "неполноты" понимается в смысле отсутствия наблюдаемых значений эндогенной переменной $\{y_{kg}\}$ в части временных срезов для некоторых предприятий. Здесь D — гипотетически полные данные.

Требуется построить логистическую динамическую нейросетевую модель оценки вероятности риска банкротства вида

$$P(t) = 1/[1 + \exp(-\hat{y}(\mathbf{x}(t), t))], \quad (3)$$

где показатель экспоненты $\hat{y}(\mathbf{x}(t), t)$ восстанавливается по неполным данным D' с помощью нейросетевого отображения вида

$$\hat{y}(\mathbf{x}, t) = F(\mathbf{x}, \mathbf{W}, t_k). \quad (4)$$

Здесь оператор нейросетевого отображения

$$F: \mathbf{x} \in R^{(n+1)} \rightarrow \hat{y} \in R^{(1)}, \quad (5)$$

где $R^{(n+1)}$, $R^{(1)}$ — пространства действительных чисел размерности $(n+1)$ и 1 соответственно; $\mathbf{W}$ — матрица параметров (синаптических весов) нейро-

сети; t — непрерывно изменяющееся время; t_k — дискретное время.

Поскольку требуемая нейросетевая модель (3) имеет четко выраженную прикладную направленность, то постановку задачи приблизим к реальным сложным условиям моделирования.

1. Помимо оговоренной неполноты данных и возникающей при этом неопределенности в оценке динамической ситуации риска банкротства, будем считать, что данные $D' \subset D$ сильно зашумлены (вплоть до сознательного искажения части данных, что может иметь место в бухгалтерской отчетности).

2. Априорный вид закона распределения шумов считается неизвестным.

3. Число наблюдений в выделенном кластере предприятий данного сектора экономики должно быть не менее двух-трех сотен.

В литературе такие сложные условия моделирования относят к классу задач с триадой "НЕ-факторов" (неполнота, неопределенность, неточность данных).

2. Описание алгоритма предлагаемого гибридного метода

Алгоритм метода (рис. 1) содержит два итерационных цикла:

- цикл внутренних итераций с индексом k , $k = 1, 2, \dots, N-1$, целью которых является восстано-

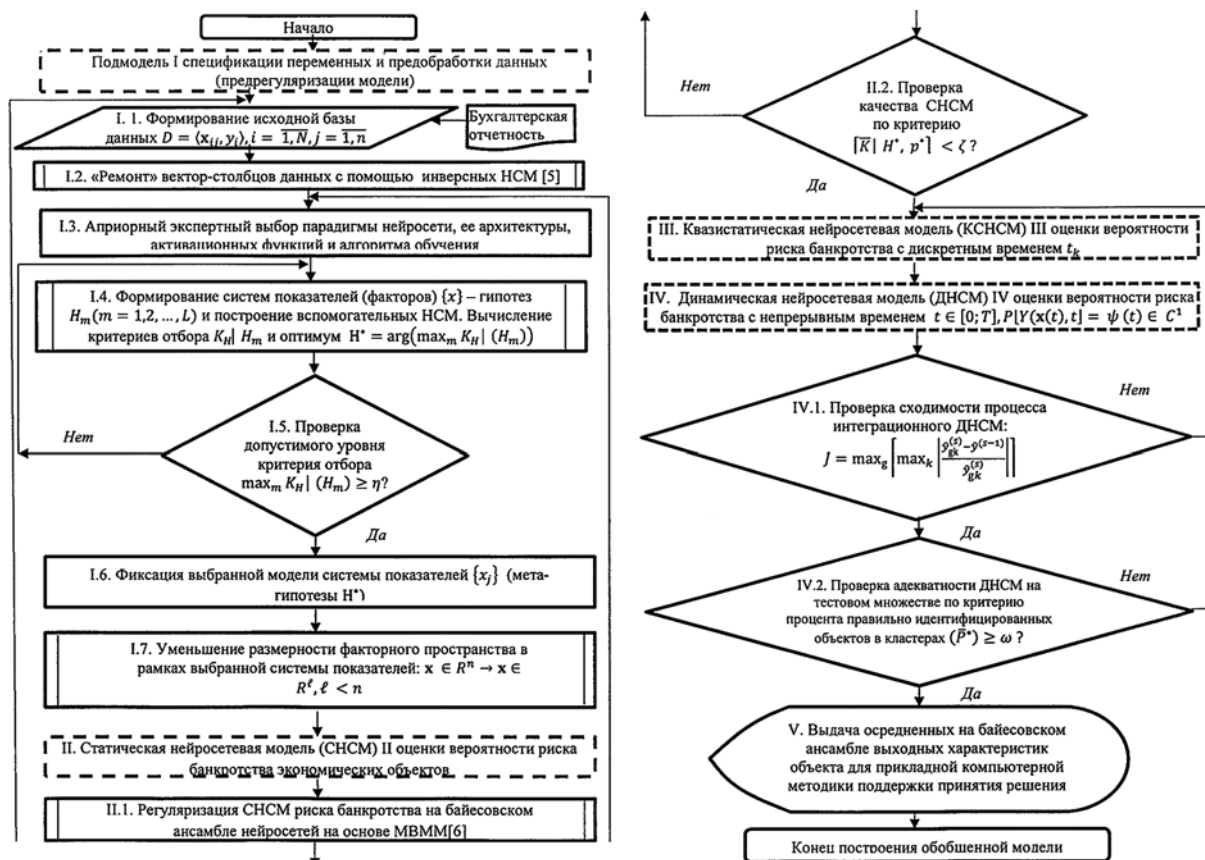


Рис. 1. Логическая схема алгоритма динамического метода оценки риска банкротства предприятий

ление эндогенной переменной $\{y_{gt_k}\}$ во временных срезах $\{t_k\}$;

- цикл внешних итераций с индексом $s = 1, 2, \dots$, целью которых является введение в нейросетевую модель вида (3) информации о времени и, соответственно, получение нейросетевой динамической модели с непрерывным временем.

Построение промежуточной квазистатической нейросетевой модели оценки риска банкротства (КСНСМ). Опишем алгоритм предлагаемого итерационного метода по шагам.

Шаг 1 ($k = 1; s = 1$). Используя данные последнего временного среза ($t = N$), где известны вектор факторов $\mathbf{x}_{gN}$ и метки "банкрот" ($P_{gN} = 1$), "небанкрот" ($P_{gN} = 0$) для всех объектов $g \in \overline{1, G}$ и соответственно значения аргумента экспоненты $\hat{y}_{gN}$, строим статическую нейросетевую модель (СНСМ) (2), например, типа многослойного персептрона (MLP) с алгоритмом обучения типа "обратного распространения ошибки" (BP) [4] (далее кратко MLP-BP).

На первом шаге внешних итераций ($s = 1$) важно восстановить наиболее точно функциональную связь $Y(x)$, скрытую в данных, в сложных условиях триады "НЕ-факторов", оговоренных во введении к статье. От качества СНСМ на первом шаге будет на последующих шагах зависеть качество восстановления динамических закономерностей $P(\hat{y}, t_k)$ во временных срезах $t_k, k = 1, 2, \dots, N - 1$. Поэтому применим для построения КСНСМ модифицированный метод вложенных математических моделей (МВММ), подробно описанный в работах [5–7], который обеспечивает регуляризацию СНСМ на основе байесовского подхода.

Поясним кратко суть процедуры регуляризации, основанной на парадигме сужения класса искомого решения обратной задачи восстановления зависимостей, скрытых в данных. В широко известной теории А. Н. Тихонова [8] решения обратных некорректных задач указанное "сужение" выполняется путем введения в алгоритм специального регуляризирующего (стабилизирующего) функционала А. Н. Тихонова, для построения которого требуется использование некоторой априорной информации об искомом решении обратной задачи. Чаще всего это информация о свойствах гладкости искомого решения. В данной статье используется та же парадигма "сужения", но другой подход (концепция) к реализации этой парадигмы, основанной на теории Байеса [5, 6]. Идея состоит в том, что оператор нейросетевого отображения (5) реализуется не на одной нейросети, а на байесовском ансамбле нейросетей, различающихся архитектурой, параметрами промежуточного слоя и видом активационных функций. После обучения ансамбля проводится фильтрация априорных гипотез — ней-

росетевых моделей по критерию качества объяснения данных, например:

$$P_q = (N_q^*/N) \geq \omega, \omega \in [0; 1]. \quad (6)$$

Здесь N_q^* — число точек тестового множества, где ошибка обобщения сети не превосходит заданного уровня ξ :

$$N_q^* = \{i_q^*\}: \delta_{q,i} = \left| \frac{y_i - \hat{y}_{qi}}{y_i} \right| \leq \xi; \\ q = \overline{1, Q}; i = \overline{1, N}. \quad (7)$$

В выражениях (6), (7) введены следующие обозначения: N — число точек тестового множества, которое должно быть одинаковым для всех нейросетей, входящих в байесовский ансамбль; q — номер нейросети в ансамбле; Q — число сетей в ансамбле; P_q — вероятный критерий приемлемого уровня качества объяснения данных, задаваемый параметрами ω и ξ , которые взаимосвязаны через прогностические свойства обученных нейросетей, т. е. $\omega = \rho(\xi)$, $\rho(\cdot)$ — некоторая непрерывная функция; i — номер наблюдения в данных; звездочка сверху означает, что эта величина прошла операцию фильтрации в алгоритме (6), (7).

Процедуры предобработки данных. Процедуры I.1–I.7 (см. рис. 1), входящие в МВММ, по сути, являются процедурами предобработки (предрегуляризации) исходных "сырых" данных, сформированных в блоке I.1 по стандартным данным бухгалтерской отчетности. Эти процедуры, кроме I.7, совпадают с аналогичными процедурами схемы КВММ из работы [6, рис. 3.5]. Общей концепцией предобработки (предрегуляризации) данных в блоках I.1–I.7 является последовательное (сверху-вниз) сужение пространства переменных $[X \times Y]$ и, соответственно, повышение качества (однородности и информативности) данных.

Сделаем некоторые комментарии к процедурам I.4 и I.7, которые детально описаны в работах [5, 6]. Формирование систем показателей факторов в процедуре I.4 — это по существу формализованный оптимальный выбор по минимуму критерия K_{H_m} взвешенной суммы ошибок I и II рода идентификации объектов "банкрот—небанкрот". $\{H_m\}$ — это m -я априорная гипотеза (на уровне метагипотез) о подходящей традиционно сложившейся системе показателей (факторов) СНСМ банкротств согласно квазيبайесовскому методу регуляризации нейросетевых моделей [6]. Критерий K_m отбора (фильтрации) указанных метагипотез имеет следующий вид:

$$K_{H_m} = N_{H_m}^* r_1 - N_{H_m}^1 r_2 - N_{H_m}^2 r_3, \quad (8)$$

где $N_{H_m}^*$ — число верно идентифицированных предприятий для данной зафиксированной байесовской метагипотезы H_m ; $N_{H_m}^1, N_{H_m}^2$ — число ошибок пер-

вого и второго рода соответственно при идентификации; r_1, r_2, r_3 — удельные веса показателей в критерии (8), характеризующие их предпочтения, т. е. коэффициенты Фишберна [9]

$$r_i = \frac{2(n-i+1)}{(n+1)n}, \quad (9)$$

где n — число ранжируемых показателей, входящих в критерий (8). Формулу (9) можно использовать в случае [10], если множество показателей $N_{H_m}^*$; $N_{H_m}^1$, $N_{H_m}^2$ упорядочено в порядке убывания их предпочтения по значимости: $N_{H_m}^* > N_{H_m}^1 > N_{H_m}^2$.

В блоке I.7 применяется нейросетевой аналог известного метода "включения" факторов в регрессионную модель [11]. Для выбранной в блоке I.4 системы факторов H^* (мы использовали 16 показателей А. О. Недосекина [9, 10], разбитых на четыре группы: "рентабельность", "ликвидность", "деловая активность", "финансовая устойчивость"). Вначале выбрано по одному фактору из каждой группы и построены вспомогательные НСМ вида (2), (5). Затем на каждом шаге итераций добавлялось по одному новому фактору, выбранному случайно. После полного перебора факторов по индексу $p = \overline{1, 16}$ фиксируется тот оптимальный набор p^* , который соответствует при тестировании модели условию

$$p^* = \arg\left(\max_p K_{H_{mp}} | H^* \right). \quad (10)$$

Нам удалось сократить размерность факторного пространства с 16 до 6 без заметного снижения критерия (10), т. е. процента правильной идентификации объектов по банкротству.

В блоке III.1 осуществляется регуляризация статической нейросетевой модели (НСМ) оценки риска банкротства на байесовском ансамбле нейросетей на основе МВММ, подробно описанного в работах [5, 6]. Качество регуляризированной модели определяется критерием (10) в блоке III.2:

$$[\bar{K} | H^*, p^*] \geq \zeta,$$

где $\bar{K}$ — усредненное на байесовском отфильтрованном ансамбле значение критерия (8).

В итоге шага 1 в блоке III.2 для фиксированного времени $t_k = T$ в последнем временном срезе мы получаем **статическую** логистическую нейросетевую модель (СЛНСМ) оценки риска банкротства вида (2), (11):

$$\hat{y}(\mathbf{x}) = F(\mathbf{x}, \mathbf{W}), \quad (11)$$

где $F(\cdot)$ — оператор нейросетевого отображения [12].

Квазистатический нейросетевой метод (КСНСМ) оценки риска банкротства с дискретным временем. Термин "квазистатический" здесь понимается в том смысле, что время в нейросетевой модели банкротства "почти учитывается" в

дискретной форме $\{t_k\}$, $k = \overline{1, N}$, но не более того. То есть квазистатическая модель НСМ не позволяет ни прогнозировать вероятность риска банкротства в промежутках времени между временными срезами, ни экстраполировать ее в будущее.

Шаг 2 ($k = 1, 2, \dots, N-1$; $s = 1$). Далее в блоке IV будем восстанавливать эндогенную переменную $\{y_{gk}\}$ в каждом временном срезе t_k во внутреннем итерационном цикле ($k = 1, 2, \dots, N-1$), используя статическую нейросетевую модель (2), (11), полученную на шаге $s = 1$ для каждого g -го предприятия, т. е. подставляя в правую часть (11) факторы $\mathbf{x}_{gk}(t_k)$ для каждого k -го временного среза.

Итогом этой процедуры является множество значений вероятностей риска банкротства во временных срезах:

$$P_g[\hat{y}(\mathbf{x}(t_k))] = 1/[1 + \exp(-\hat{y}(\mathbf{x}(t_k)))];$$

$$k = 1, 2, \dots, N-1; g = \overline{1, G}; \quad (12)$$

$$\{P_g(t_k)\}_{k=1}^N = P_{gt_1}, P_{gt_2}, \dots, P_{gt_N}. \quad (13)$$

Замечание 1. Модель (4), (5) (12), (13) — это уже квазистатическая нейросетевая модель оценки риска банкротства с логистической трансформацией вероятности (ЛКСНСМ) и дискретным временем $\{t_k\}$. Здесь уже можно приближенно оценивать стадии развивающегося процесса банкротства по вероятностям во временных срезах, $g = \overline{1, G}$.

На языке функционального анализа модель (11), (12) представляет собой композицию двух нелинейных операторов:

$$\mathbf{x} \in R^n \xrightarrow{\beta} P(t_k) \in M^N; \beta = A \circ F; k = \overline{1, N-1}, \quad (14)$$

где F — оператор нейросетевого отображения вида (11); A — оператор логистической трансформации вероятности вида (12); M^N — конечномерное пространство дискретных функций размерности N .

Динамический нейросетевой метод (ДНСМ) оценки вероятности риска банкротства с непрерывным временем.

Шаг 3 ($k = 1, 2, \dots, N-1$; $s = 2$). На шаге 3 в блоке V алгоритма мы переходим к внешним итерациям введения в динамическую модель непрерывного времени $t \in [0; T]$. Для этого в каждой НСМ в состав вектора факторов $\mathbf{x}$ введем в качестве независимого нового фактора — время t :

$$\mathbf{x} = (x_1, x_2, \dots, x_j, \dots, x_n, x_{n+1} \equiv t). \quad (15)$$

Используя восстановленные данные, т. е. кортежи

$$\langle \hat{y}_{gk}, \mathbf{x}_{gk}, t_k \rangle, k = 1, 2, \dots, N-1, N, \quad (16)$$

обучаем заново и тестируем нейросети:

$$\hat{y}(\mathbf{x}(t), t) = (F(\mathbf{x}(t), t, \mathbf{W}), \quad (17)$$

где в нейросетевой модели время t входит явно и является непрерывным. Значит модель (3), (17) можно в полной мере считать динамической, по-

Структура байесовского ансамбля нейросетей для статической модели

№ сети	Число скрытых слоев	Оптимальное число нейронов в скрытых слоях	Вид активационной функции в скрытых слоях
HCM1	1	7	Гиперболический тангенс
HCM2	1	6	Сигмоид
HCM3	2	14, 9	Гиперболический тангенс в обоих слоях
HCM4	2	14, 11	Гиперболический тангенс в первом слое, сигмоид — во втором слое
HCM5	2	8, 13	Сигмоид в первом слое, гиперболический тангенс во втором
HCM6	2	13, 12	Сигмоид в обоих слоях
HCM7	3	14, 13, 14	Гиперболический тангенс в первом и третьем слоях, сигмоид во втором слое
HCM8	3	8, 14, 12	Гиперболический тангенс во всех слоях
HCM9	3	12, 14, 10	Гиперболический тангенс во всех слоях
HCM10	3	14, 13, 15	Гиперболический тангенс во всех слоях

Таблица 2

Результаты тестирования нейросетевых статических моделей, обучавшихся на данных за последний год ($t = 5$)

№ сети	Доля верно распознанных предприятий, %	Всего ошибок идентификации (банкрот—небанкрот)	Число ошибок 2-го рода	Число ошибок 1-го рода	Критерий K_{H_k}
HCM1	65,85	14	5	9	10,33
HCM2	80,49	8	5	3	14,33
HCM3	80,49	8	3	5	14,66
HCM4	78,05	9	5	4	13,66
HCM5	78,05	9	5	4	13,66
HCM6	85,37	6	3	3	16,00
HCM7	75,61	10	6	4	12,83
HCM8	82,93	7	3	4	15,33
HCM9	80,49	8	3	5	14,66
HCM10	78,05	9	3	6	14,00
Среднее на отфильтрованном ансамбле	85,37	6	3	3	16,00

скольку при обучении НСМ учитывается, что зависимость (1) "плывет" во времени. Причем можно получить информацию о вероятности риска банкротства, как во временных срезах, так и в любой точке между ними и, главное, экстраполировать ее на заданный горизонт прогноза. Следовательно, появляется возможность уточнения восстанавливаемых значений эндогенных переменных в corteжах:

$$y_{gk}^{(2)} = F(\mathbf{x}(t), t, \mathbf{W}^{(1)}), k = 1, 2, \dots, N - 1, \quad (18)$$

где синаптические веса $\mathbf{W}^{(1)}$ берутся из предыдущего шага внешних итераций ($s = 1$).

Шаг 4 ($k = 1, 2, \dots, N - 1$; $s = 3$). Заново обучаем ДНСМ с учетом новых восстановленных данных (18), т. е. реализуем шаги 2, 3 и т. д.

Критерий сходимости итерационного процесса по индексу s восстановления эндогенных переменных $\hat{y}_{gk}$ и уточнения ДНСМ выбираем в виде

$$J = \max_g \left[\max_k \left| \frac{\hat{y}_{gk}^{(s)} - \hat{y}_{gk}^{(s-1)}}{\hat{y}_{gk}^{(s)}} \right| \right] \leq \alpha;$$

$$g = \overline{1, G}; k = \overline{1, N - 1}; s = 3, 4, \dots, \quad (19)$$

Затем критерий (6) усредняется на отфильтрованном ансамбле ДНСМ, вычисляется $\bar{P}^*$ и проверяется условие адекватности модели в блоке V.2 на рис. 1.

Замечание 2. Вычислительный процесс построения ДНСМ является довольно трудоемким, поскольку восстанавливать приходится эндогенные переменные $\{y_{gk}^{(s)}\}$ на шагах 3, 4, ... для всех предприятий, входящих в данные, и во всех временных срезах. Это естественная плата за переход от квазистатической нейросетевой модели к динамической для данного сектора экономики и данного региона (страны). Зато последующее использование на практике ДНСМ довольно просто — оно сводится к постановке на вход обученной и верифицированной нейросети, единой для всего сектора модели, вектора $\mathbf{x}(t)$, факторов для анализируемого g -го объекта в интересующие моменты времени t .

Замечание 3. Основная ценность модели банкротства, получаемой с помощью ДНСМ, — это возможность получения качественного прогноза банкротства с информацией о стадиях его развития.

4. Количественные оценки построения динамической модели банкротств

Использовались реальные данные для строительных организаций из работы [9].

Этап 1 (Итерация 0). Исследовалась статическая нейросетевая модель (11) для фиксированного временного среза $t_k = t_5$. Был обучен и протестирован ансамбль из 10 сетей на выборке из 136 наблюдений. Результаты тестирования и структура сетей представлены в табл. 1 и 2.

После фильтрации обученным сетям, т. е. моделям (11), были предъявлены данные за предыдущие 4 года. Получены осредненные оценки величины Y для всех временных срезов $t = 1, 2, 3, 4$ согласно (12). Была сформирована общая выборка, состоящая из 680 наблюдений ($136 \cdot 5 = 680$).

Этап 2 (Внешняя итерация $s = 1$). В общую выборку был добавлен фактор времени (относительное время). Данные были разбиты на обучающее и тестовое множества следующим образом: 610 на-

Результаты тестирования байесовского ансамбля динамических нейросетевых моделей для первого шага внешних итераций ($s = 1$)

№ сети	Доля верно распознанных предприятий, %	Всего ошибок идентификации	Число ошибок 2-го рода	Число ошибок 1-го рода	Критерий K_{H_k}	Средняя на отфильтрованном ансамбле ошибка обобщения δ по (7), %
HCM1	84,29	11	5	6	26,83	163
HCM2	85,71	10	1	9	28,16	173
HCM3	84,29	1	6	5	26,66	170
HCM4	88,57	8	4	4	29	172
HCM5	85,71	10	1	9	28,16	177
HCM6	71,43	20	0	20	21,66	171
HCM7	87,14	9	4	5	28,33	179
HCM8	80,0	14	3	11	25,16	191
HCM9	81,43	13	6	7	25,33	174
HCM10	85,71	10	3	7	27,83	178
Среднее на отфильтрованном ансамбле	87,14	9	3	6	28,5	

блюдений в обучающем множестве и 70 в тестовом. На этих данных был переобучен ансамбль нейросетей (оптимальное число нейронов в скрытых слоях подбирали заново). Результаты тестирования полученной модели (17) показаны в табл. 3.

Критерий фильтрации $K_{H_k} > 26$.

Оставшимся после фильтрации семи сетям (№ 1, 2, 3, 4, 5, 7, 10) были предъявлены данные и получены значения $\hat{y}$.

Дополнительно для каждой нейросети была рассчитана средняя относительная погрешность аппроксимации обучения по формуле

$$\bar{\delta} = \sum_{i=1}^{N_{learn}} \left| \frac{y_i - \hat{y}_i}{y_i} \cdot 100 \% \right| / N_{learn} \quad (20)$$

где N_{learn} — число наблюдений в обучающем множестве.

Этап 3 (Итерация $s = 2$). В данных значениях Y (показатель экспоненты) были заменены на значения $\hat{y}$, полученные на предыдущей итерации. Цикл обучения повторялся заново. Результаты показаны в табл. 4.

Результат. Практически процесс внешних итераций сошелся за два шага, все сети (кроме HCM 10) показали один и тот же результат. При этом ошибки идентификации наблюдаются в одних и тех же точках тестового множества. Этот результат можно объяснить тем, что при переходе к следующей итерации нейросетевое отображение "сжимает" невязку для $\hat{y}$, т. е. отображение (4) является сжимающим [8]. Дополнительное "сжатие" реализует также логистическая функция (3).

Относительная погрешность также уменьшилась примерно в 2,5 раза.

Полученная динамическая модель с непрерывным временем позволяет достаточно точно отслеживать характерную динамику вероятности банкротства и своевременно принимать соответствующие управляющие воздействия. Для иллюстрации на рис. 2 приведены графики динамики вероятно-

Таблица 4

Результаты тестирования динамического байесовского ансамбля нейросетей, для второго шага внешних итераций ($s = 2$)

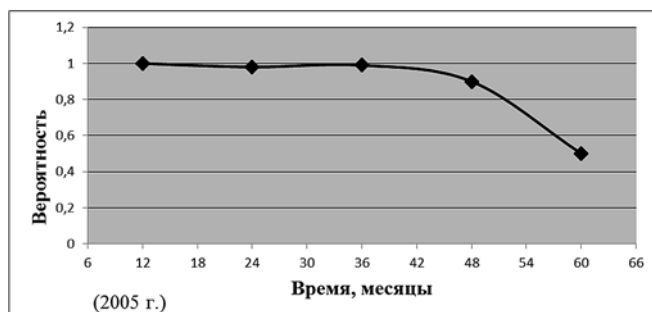
№ сети	Доля верно распознанных предприятий, %	Всего ошибок	Число ошибок 2-го рода	Число ошибок 1-го рода	K_{H_k}	δ , %
HCM1	85,71	10	1	9	26,16	63,37
HCM2	85,71	10	1	9	28,16	65,98
HCM3	85,71	10	1	9	28,16	78,92
HCM4	85,71	10	1	9	28,16	69,61
HCM5	85,71	10	1	9	28,16	66,38
HCM7	85,71	10	1	9	28,16	70,84
HCM10	87,14	9	1	8	28,83	70,03
Среднее на отфильтрованном ансамбле	85,91	10	1	9	28	

сти риска банкротств трех предприятий из выборки 136 предприятий строительных организаций [9].

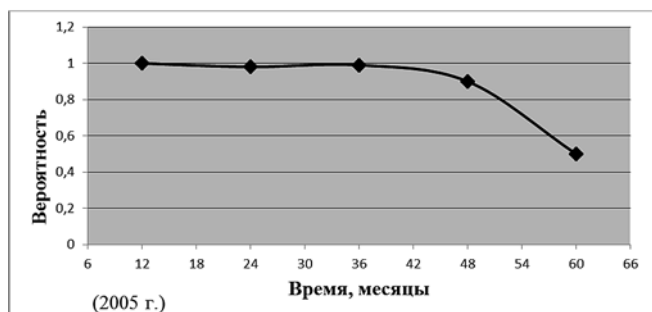
Рис. 2, *a–в* соответствуют объектам из трех различных кластеров, где кластерообразующие признаки x входят в качестве аргумента в показатель экспоненты $\hat{y}(x, t)$ в логистическую функцию (3). Видно, что различие в признаках x трансформируется в динамических моделях в различие динамики риска банкротства. На рис. 2, *a* видно, что предприятие, близкое к банкротству в период с 2005 по 2008 г., благодаря принятым антикризисным мерам в 2009 г. улучшило свое финансово-экономическое состояние, и вероятность риска банкротства снизилась до 0,42.

Из рис. 2, *б* следует, что предпринятые антикризисные меры оказались недостаточно эффективными: вероятность риска банкротства оказалась к 2010 г. на высоком уровне ($P = 0,8$).

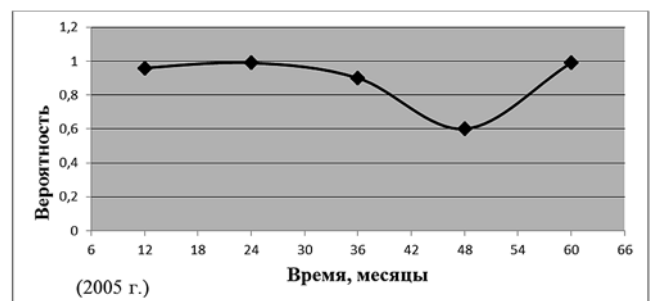
На рис. 2, *в* видно, что принимаемые антикризисные меры в период с 2008 г. по 2009 г. оказались



а)



б)



в)

Рис. 2. Динамика вероятности риска банкротств

недостаточными, и к 2010 г. предприятие обанкротилось.

Таким образом, вычислительные эксперименты подтвердили эффективность основной идеи разработки гибридного нейросетевого метода построения динамической модели банкротств с непрерывным временем, т. е. идеи восстановления динамической зависимости вероятности риска банкротств предприятий по неполным данным.

Заключение

Разработан динамический нейросетевой метод (ДНСМ) оценки банкротств, который может быть использован в широком спектре динамических прикладных задач оценки вероятности риска наступления нежелательных событий в экономических объектах применительно к условиям моделирования, приближенным к реальным (наличие триады "НЕ-факторов" — неполнота, неопределенность, неточность данных).

Достоинством предложенного метода, по мнению авторов, является принципиальная возможность оценивания на его основе интегрального показателя финансово-экономического состояния экономического объекта — вероятности риска банкротства — для любого момента времени (как между временными "срезами" в данных, так и для заданного горизонта прогноза).

Следовательно, появляется возможность текущего мониторинга стадий развивающегося кризиса объекта и принятия соответствующих упреждающих управленческих воздействий. Примером может служить реструктуризация кредиторской задолженности корпорации-заемщика.

Предложенный метод достаточно подробно проанализирован на реальных данных с точки зрения устойчивости (регуляризации) и адекватности получаемой нейросетевой модели.

По сути, имеются три ограничения для области применения ДНСМ.

1. В последнем временном срезе $t_N = T$ должны быть известны метки "банкрот—небанкрот" для эндогенной переменной ($P = 1$ либо $P = 0$).

2. Выборка должна содержать по крайней мере 200—300 наблюдений.

3. Среди наблюдений должны быть как предприятия "банкроты", так и предприятия "небанкроты" (примерно одинаковое число в последнем временном срезе $t_N = T$).

Традиционно используемые ограничения на вид закона распределения шумов в данных в нашей работе отсутствуют.

В качестве перспектив развития предложенного ДНСМ хотелось бы отметить разработку алгоритмов введения качественных факторов в вектор экзогенных переменных x в модели (17).

Список литературы

- Altman E. I. Financial ratios, discriminant analysis and the prediction of corporate bankruptcy // Journal of Finance. 1968. Vol. 23, N. 9. P. 589—609.
- Ohlson J. A. Financial Ratios and the Probabilistic Prediction of Bankruptcy // Journal of Accounting Research. 1980. N. 18 (1). P. 109—113.
- Галушкин А. И. Нейрокомпьютеры и их применение на рубеже тысячелетий в Китае. В 2-х томах. Том 1. М.: Горячая линия — Телеком, 2004. 367 с., Том 2. М.: Горячая линия — Телеком, 2004. 464 с.
- Галушкин А. И. Теория нейронных сетей. Кн. 1: учеб. пособие для вузов. М.: ИПРЖ, 2000. 416 с.
- Горбатков С. А., Фархиева С. А., Белолищев И. И. Обобщенная нейросетевая модель риска банкротства предприятия // XVI Всероссийская научно-техническая конференция "Информатика — 2014" 20: Сб. науч. трудов в 3-х частях. Ч. 3. М.: Изд-во НИЯУ МИФИ, 2013. С. 210—220.
- Горбатков С. А., Полупанов Д. В., Макеева Е. Ю., Бирюков А. Н. Методологические основы разработки нейросетевых моделей экономических объектов в условиях неопределенности / Под ред. С. А. Горбаткова. М.: ИД "Экономическая газета", 2012. 494 с.
- Горбатков С. А., Белолищев И. И., Фархиева С. А. Приближенный метод байесовской регуляризации и двухступенчатая оценка адекватности гибридной нейросетевой модели на-

логового контроля // Научная сессия НИЯУ МИФИ-2011: XIII Всероссийская научно-техническая конференция "Нейроинформатика-2011", г. Москва, 24–28 января 2011 г.: Сб. научных трудов. М.: Изд-во МИФИ, 2011. С. 144–154.

8. **Тихонов А. Н., Арсенин В. Я.** Методы решения некорректных задач. М.: Наука. Физматлит, 1984. 288 с.

9. **Горбатков С. А., Белоплицев И. И., Макеева Е. Ю.** Выбор системы экономических показателей для диагностики и прогнозирования банкротств на основе байесовского подхода и интеллектуальных информационных технологий // Вестник Финансового университета при Правительстве РФ. 2013. № 4 (76). С. 50–62.

10. **Недосекин А. О.** Комплексная оценка риска банкротства корпорации на основе нечетких описаний. URL: <http://sedok.narod.ru/sc-group.html>.

11. **Горбатков С. А., Полупанов Д. В., Фархиева С. А., Коротнева М. В.** Эконометрика: учеб. пособие / Под. ред. С. А. Горбаткова. Уфа: Изд-во РИЦ БашГУ, 2012. 204 с.

12. **Хайкин С.** Нейронные сети: полный курс: учебник, 2-е изд.; пер. с англ. М.: Вильямс, 2006. 1104 с.

S. A. Gorbakov, Professor of Mathematics and Computer Science

of the Ufa branch of the Financial University under the Government of the Russian Federation,

Doctor of Technical Sciences, e-mail: sgorbakov@mail.ru,

L. I. Kasimova, Client Manager of "Sberbank of Russia. Sterlitamak branch", e-mail: liankakasimva@yandex.ru

Neural Network Iterative Method for Constructing Dynamic Models of the Probability of Bankruptcy Risk at Incomplete Data

The goal of this article is to develop the original iterative method of constructing dynamic model estimates the probability of the risk of bankruptcy for a wide range of facilities in the financial and real sectors of the economy. The models of this class are had the great theoretical interest due to their poor knowledge and interest for applications. These include the problem of estimating the developing stages in time the corporations bankruptcy; evaluation of the financial and economic situation of the partners in the transaction; evaluation of investment attractiveness of corporations; ensuring the economic security corporations through constant monitoring of their financial condition; decision support for the restructuring of credit debt of legal entities; making decisions on taxation, and in many other applications.

A key objective of the article is to restore the hidden dynamic patterns of development of the process of bankruptcy of an economic object in data with using an ensemble of neural networks. The data are incomplete: in general, the label "bankrupt — not bankrupt" contained not in all time cuts, but only in the final cut. The data sets are incomplete and very noisy with unknown noise distribution law. The main result of this paper is an original dynamic iterative neural network method (DINNM) of constructing predictive models of this class. To ensure the sustainability DINNM in such difficult modeling conditions develops a Bayesian approach to regularization neural network model. On the basis of real data for construction companies was conducted a series of computational experiments to substantiate the adequacy obtained dynamic model. The iterative process of restoring dynamic dependencies in DINNM converges in a few iterations with the discrepancy of a few percent.

Keywords: diagnostics and prediction of bankruptcy, neural network, dynamic logistic model, Bayesian regularization

References

1. **Altman E. I.** Financial ratios, discriminant analysis and the prediction of corporate bankruptcy, *Journal of Finance*, 1968, vol. 23, no. 9, pp. 589–609.

2. **Ohlson J. A.** Financial Ratios and the Probabilistic Prediction of Bankruptcy, *Journal of Accounting Research*, 1980, no. 18 (1), pp. 109–113.

3. **Galushkin A. I.** *Neyrocomputeri i ich primeneniye na rubege tisyachetij v Kitae*. V 2-h tomah. Vol. 1. Moscow, Gorjachaja linija-Telekom, 2004, 367 p., Vol. 2. Moscow, Gorjachaja linija-Telekom, 2004, 464 p.

4. **Galushkin A. I.** *Teoria neyronnykh setej*, Kn. 1, uchebn. posobie, Moscow, IPRY, 2000, 416 p.

5. **Gorbakov S. A., Farhieva S. A., Beloplichev I. I.** Obobshhennaya neyrosetevaya model' riska bankrotstva predpriyatija // XVI Vserossiyskaya nauchno-tehnicheskaya konferenciya "Neiroinformatika-2014" 20: Sbornik nauchnykh trudov v 3-h chastyah. Ch. 3. — М.: NIJaU MIFI. 2013, s. 210–220.

6. **Gorbakov S. A., Polupanov D. C., Makeeva E. Yu., A. Birjukov A. N.** *Metodologicheskiye osnovy razrabotki neyrosetevykh modeley jekonomiceskikh ob'ektov v usloviyah neopredelennosti*, pod red. S. A. Gor-

batkova, Moscow, Izdatel'skij dom "Jekonomicheskaja gazeta", 2012. 494 p.

7. **Gorbakov S. A., Beloplichev I. I., Farhieva S. A.** Priblizhennyj metod bajesovskoy reguljarizacii i ddvuhstupenchataja ocenka adekvatnosti gibridnoj neyrosetevoj modeli nalogovogo kontrolja // Nauchnaja sessija NIJaU MIFI-2011: XIII Vserossiyskaja nauchno-tehnicheskaja konferenciya "Neiroinformatika-2011": Sb. nauchnykh trudov, Moscow, MIFI, 2011, pp. 144–154.

8. **Tihonov A. N., Arsenin V. Ja.** *Metody reshenija nekorrektnykh zadach*. Moscow, Nauka, Fizmatlit, 1986, 288 p.

9. **Gorbakov S. A., Beloplichev I. I., Makeeva E. Ju.** Vybor sistemy jekonomicheskikh pokazatelej dlja diagnostiki i prognosirovaniya bankrotstv na osnove bajesovskogo podhoda i intellektual'nykh informacionnykh tehnologij, *Vestnik Finansovogo Universiteta pri Pravitel'stve RF*, 2013, № 4 (76), pp. 50–62.

10. **Nedosekin A. O.** *Kompleksnaja ocenka riska bankrotstva korporacij na osnove nechetkih opesaniy*, URL: <http://sedok.narod.ru/sc-group.html>.

11. **Gorbakov S. A., Polupanov D. V., Farhieva S. A., Korotneva M. V.** *Jekonometrika*: Ucheb. posobie, Ed. S. A. Gorbakov, Ufa, Izd-vo RIC BashGU, 2012, 204 p.

12. **Haykin S.** *Nejronnye seti*: polnyj kurs: Uchebnik, 2-e izd., Per. s angl., Moscow, Vil'jams, 2006, 1104 p.

Указатель статей, опубликованных в журнале "Информационные технологии" в 2015 г.

ОБЩИЕ ВОПРОСЫ

Цветков В. Я. Информационная неопределенность и информационная определенность в науках об информации . . . № 1

ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

Борисовский П. А., Еремеев А. В. Построение расписаний многопродуктового производства с использованием целочисленного линейного программирования и эволюционных вычислений № 6

Воробьев В. В., Паршикова Е. А. Применение мультимножеств для оценки ситуации мобильным агентом № 6

Дергачев К. В., Капулин Д. В., Казанцев М. А. Автоматизированная система управления складом уровня цеха для проектного производства № 11

Дыкин В. С., Мусатов В. Ю., Варезников А. С., Большаков А. А., Сысоев В. В. Применение генетического алгоритма для определения архитектуры нейронной сети для прибора вида "электронный нос" № 12

Захлебный И. В., Фомичев В. А. Разработка метода семантического поиска специалистов в корпоративной базе данных по естественно-языковым запросам № 5

Имамвердиев Я. Н., Сухостат Л. В. Разработка робастного метода извлечения речевых признаков на основе эмпирического вейвлет-преобразования № 1

Катруца А. М., Стрижов В. В. Проблема мультиколлинеарности при выборе признаков в регрессионных задачах № 1

Катуева Я. В., Назаров Д. А. Методы параметрического синтеза на основе сеточного представления области работоспособности № 9

Кулагин В. П. Тензорные методы исследования структур сетей Петри № 2

Кулагин В. П., Цветков В. Я., Лапчинская М. П. Идентификация морских объектов при сильном волнении водной поверхности № 5

Левин В. И. Анализ поведения неточно заданных функций с помощью интервально-дифференциального исчисления № 3

Левин В. И. Интервально-дифференциальные уравнения и моделирование, систем в условиях неопределенности . . № 11

Лелеков С. Г., Лях А. М. Нечеткий вывод в адаптивных таксономических экспертных системах № 12

Лучкова С. В., Перемитина Т. О., Ященко И. Г. Использование нечеткого моделирования для повышения репрезентативности информации на примере анализа характеристик нефти . . № 5

Мочалов И. А., Хрисат М. С., Шихаб Еддин М. Я. Нечеткие дифференциальные уравнения в задачах управления. Часть I . . . № 3

Мочалов И. А., Хрисат М. С., Шихаб Еддин М. Я. Нечеткие дифференциальные уравнения в задачах управления. Часть II . . № 4

Мочалов И. А., Хрисат М. С., Шихаб Еддин М. Я. Нечеткие уравнения в частных производных в задачах управления . . . № 8

Родионов А. Н. Периодические данные: идентификация, моделирование и представление в базах данных № 9

Сирота А. А., Цуриков А. В. Модели и алгоритмы классификации фрагментов текста и их применение для создания контентно-зависимых цифровых водяных знаков № 2

Сирота А. А., Цуриков А. В. Модели и алгоритмы классификации фрагментов текста и их применение для создания контентно-зависимых цифровых водяных знаков № 2

Стемпковский А. Л., Тельпухов Д. В., Соловьев Р. А., Мячиков М. В. Повышение отказоустойчивости логических схем с использованием нестандартных мажоритарных элементов № 10

Фомичев В. А., Разоренов А. А. Значение теории K-представлений для исследований по автоматическому выявлению семантических ролей № 6

Харламов А. А., Ермоленко Т. В. Нейросетевая среда (нейроморфная ассоциативная память) для преодоления информационной сложности. Поиск смысла в слабоструктурированной массивах информации. Часть I. Структурная обработка информации в коре № 11

Харламов А. А., Ермоленко Т. В. Нейросетевая среда (нейроморфная ассоциативная память) для преодоления информационной сложности. Поиск смысла в слабоструктурированной массивах информации. Часть II. Семантическая обработка информации в гиппокампе. Модель мира № 12

ных массивах информации. Часть II. Обработка информации в гиппокампе. Модель мира № 12

МОДЕЛИРОВАНИЕ И ОПТИМИЗАЦИЯ

Бойко А. А. Способ стратифицированного аналитического описания процесса функционирования информационно-технологических средств № 1

Бронштейн Е. М., Копылов И. Е. Оптимизационная задача транспортной логистики с простыми маршрутами передвижения № 9

Булычев Г. Г. Моделирование динамического разрушения полых тел продуктами детонации № 6

Вдовин А. Н., Четырбоцкий А. Н., Четырбоцкий В. А. Компьютерное моделирование динамики роста рыб (на примере южного одноперого терпуга *Pleurogrammus azonus*). Часть I . . . № 2

Вдовин А. Н., Четырбоцкий А. Н., Четырбоцкий В. А. Компьютерное моделирование динамики роста рыб (на примере южного одноперого терпуга *Pleurogrammus azonus*). Часть II № 3

Зарубин В. С., Кувыркин Г. Н., Савельева И. Ю. Иерархическая система математических моделей процесса теплопроводности в композиционном материале № 3

Казakov П. В. Использование дифференциальной эволюции при определении множества Парето генетическими алгоритмами многокритериальной оптимизации № 2

Каширина И. Л., Львович Я. Е., Сорокин С. О. Модели и численные методы оптимизации формирования эффективной сетевой системы с кластерной структурой № 9

Комоско Л. Ф., Бацын М. В. Эффективная раскраска графа с помощью битовых операций № 7

Кухаренко Б. Г., Солнцева М. О. Анализ результатов кластеризации многомерных траекторий посредством моделей линейных динамических систем № 2

Кухаренко Б. Г., Солнцева-Чалей М. О. Применение моделей нелинейных динамических систем для анализа результатов кластеризации многомерных траекторий № 5

Кухаренко Б. Г., Солнцева-Чалей М. О. Спектральный метод с использованием полярной кривизны для анализа результатов кластеризации многомерных траекторий № 12

Мишин И. В. Численное моделирование однолинейных систем обслуживания № 5

Моисеев А. А. Моделирование химической стабильности жидкости методом теории подобия № 9

Моисеев А. А. Модель продаж стандартными партиями . . № 12

Орехов Э. Ю. Способ генерации примеров для тестирования эвристических алгоритмов решения целочисленных задач . . № 7

Струченков В. И. Комбинированные методы решения вариационных задач с заданной структурой экстремали и ограничениями № 1

Чеканин В. А., Чеканин А. В. Повышение эффективности конструирования ортогональной упаковки объектов № 1

Шелудько А. С., Ширяев В. И. Алгоритм гарантированного оценивания параметра одномерного хаотического отображения № 1

БЕЗОПАСНОСТЬ ИНФОРМАЦИИ

Барабанов А. В., Марков А. С., Цирлов В. Л. Оценка соответствия средств защиты информации "Общим критериям" № 4

Глушенко С. А., Долженко А. И. Система поддержки принятия решений нечеткого моделирования рисков информационной безопасности организации № 1

Имамвердиев Я. Н. Нечеткая когнитивная модель стратегического управления информационной безопасностью электронного правительства № 6

Капулин Д. В., Дрозд О. В. Устройство аппаратного шифрования производственных данных № 8

Маркин Д. О., Комашинский В. В., Баранов И. Ю. Модель управления профилем защиты мобильного устройства при доступе к услугам с разным уровнем конфиденциальности № 8

Перегудов М. А., Бойко А. А. Оценка защищенности сети пакетной радиосвязи от имитации абонентских терминалов на уровне процедуры случайного множественного доступа к среде типа S-ALOHA № 7

Щеглов К. А., Щеглов А. Ю. Возможности методов резервирования для повышения уровня интегрированной информационно-эксплуатационной безопасности современных информационных систем № 7

Щеглов К. А., Щеглов А. Ю. Интерпретация и моделирование угрозы атаки на информационную систему. Часть I. Моделирование угрозы уязвимости и интерпретация угрозы атаки № 12

Щеглов К. А., Щеглов А. Ю. Технология защиты данных, обрабатываемых в распределенных информационных системах. № 6

ПРИКЛАДНЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

- Жусов Д. Л., Козленко А. В., Шульга Р. В.** Алгоритм анализа глубины гипертекстовых переходов официальных сайтов федеральных органов исполнительной власти Российской Федерации. № 1
- Зак Ю. А.** Вероятностные динамические модели анализа работы сборочных конвейерных линий. № 4
- Зак Ю. А.** Об одной задаче построения допустимых и оптимальных маршрутов доставки грузов. № 5
- Панкратьев П. С., Шакиров В. А.** Двухуровневый многокритериальный анализ пунктов строительства гидроэлектростанции. № 5
- Ракитин В. В., Русаков С. Г.** Генераторные схемы на мемристорных элементах. № 10
- Яшина А. Г., Прозоров Д. Е.** Взвешенная косинусная мера векторной модели информационного поиска речевых документов. № 9

ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ И СЕТИ

- Аряшев С. И., Бобков С. Г., Зубковский П. С., Ивасюк Е. В.** Реализация блока компенсированного суммирования для повышения точности аппаратных вычислений. № 8
- Асратян Р. Э., Лебедев В. Н., Орлов В. Л.** Организация защищенных каналов взаимодействия на основе применения протокола HTTPS в прокси-серверах. № 9
- Богатырев В. А., Богатырев А. В.** Оптимизация резервированного распределения запросов в кластерных системах реального времени. № 7
- Коваленко В. Б., Дордопуло А. И., Гудков В. А., Гуленок А. А., Сластен Л. М.** Использование софт-архитектур при решении задач цифровой обработки сигналов на реконфигурируемых вычислительных системах. № 2
- Оцоков Ш. А.** Ускорение высокоточных вычислений за счет распараллеливания операции округления в комплексе систем счисления. № 5
- Рзаев Р. Р., Гоюшов А. И., Алмасов А. Ш.** Оценка качества услуг телекоммуникационной сети связи на основе метода нечеткого моделирования. № 3
- Саак А. Э.** Диспетчеризация массивов заявок суммарной ресурсной меры, равной квадрату целого числа. № 9
- Федюнин Р. Н.** Оценка пространственной сложности функциональных блоков АЛУ на базе однородных вычислительных структур. № 3
- Шыхалиев Р. Г.** Повышение эффективности мониторинга компьютерных сетей на основе оптимизации системы поллинга. № 8

ПРОГРАММНАЯ ИНЖЕНЕРИЯ

- Авдеев Н. А., Бибило П. Н.** Расширение возможностей автоматизированного проектирования цифровых систем при использовании стандарта VHDL'2008. № 7
- Гридин В. Н., Дмитриев Г. Д., Анисимов Д. А.** Методика построения веб-сервисов для расчета чувствительности передаточных функций к вариации параметров. № 11
- Димов Э. М., Маслов О. Н., Трошин Ю. В.** Выбор средств программного обеспечения процесса статистического имитационного моделирования. № 2
- Змеев Д. Н., Климов А. В., Левченко Н. Н., Окунев А. С., Стемпковский А. Л.** Эмуляция аппаратно-программных средств параллельной потоковой вычислительной системы "Буран". № 10
- Петров Ю. И., Карнаухов Ю. А.** Программное обеспечение "FK-Monitoring" как средство мониторинга бизнес-процессов. № 7
- Поскоин А. В.** Интеграция SQL-ориентированных СУБД и NoSQL-систем на уровне объектного отображения. № 2
- Сметанин С. И., Игнатюк В. А., Евстифеев А. А.** Способ реализации программной веб-части системы спутникового мониторинга. № 6
- Стеников В. А., Бархатенко Е. А., Соколов Д. В.** Методический подход на основе концепции Model-Driven Engineering и онтологий к разработке программного обеспечения для проектирования теплоснабжающих систем. № 3

ЦИФРОВАЯ ОБРАБОТКА СИГНАЛОВ И ИЗОБРАЖЕНИЙ

- Белим С. В., Селиверстов С. А.** Использование метода анализа иерархий для выявления импульсного шума в графических объектах. № 4

- Волынская А. В., Сапожников И. В.** Моделирование линейного мультипараметрического модема для каналов передачи информации с низким отношением сигнал/помеха. № 8
- Гетманов В. Г., Борзунов Г. И.** Алгоритм параллельных вычислений для задачи спектрально-временного анализа на базисных полигармонических функциях. № 6
- Дворников С. В., Манаенко С. С., Дворников С. С.** Параметрическая мимикрия сигналов, модулированных колебаниями и сформированных в различных функциональных базисах. № 4
- Дворников С. В., Манаенко С. С., Дворников С. С., Погорелов А. А.** Синтез фазоманипулированных вейвлет-сигналов. № 2
- Дворников С. В., Пшеничников А. В., Манаенко С. С.** Помехоустойчивая модель сигнала КАМ-16 с трансформированным созвездием. № 9
- Кухаренко Б. Г., Солнцева-Чалей М. О.** Применение моделей непрерывного профиля для анализа результатов кластеризации многомерных траекторий. № 8
- Мандрикова О. В., Соловьев И. С., Залаяев Т. Л.** Методы анализа вариаций геомагнитного поля и данных космических лучей. № 11
- Потапов В. П., Попов С. Е.** Многопоточный алгоритм конвертации чередования данных каналов гиперспектральных снимков. № 9
- Светушков Н. Н.** Параллельный метод струн для численного решения нелинейных задач теплопроводности. № 9
- Табаков Ю. Г.** Модель и алгоритм обработки низкочастотного сигнала для тренажера на основе вейвлет-преобразований. № 6
- Трофимов А. Г., Колодкин И. В., Ушаков В. Л., Величковский Б. М.** Метод агломеративной сегментации пространственно-распределенных временных рядов на основе характеристик бегущих волн. № 7
- Червяков Н. И., Ляхов П. А., Калита Д. И.** Выбор оптимального набора модулей системы остаточных классов для повышения производительности фильтров с конечной импульсной характеристикой. № 12

ГЕОИНФОРМАЦИОННЫЕ СИСТЕМЫ

- Беляков С. Л., Белякова М. Л., Брехачева А. И.** Геоинформационные модели для принятия решений на основе опыта. № 7
- Бешенцев А. Н.** Геоинформационная система долгосрочного мониторинга природопользования. № 8
- Дулин С. К., Дулина Н. Г., Никишин Д. А.** Особенности моделей геоданных и методов их обработки в аспекте обеспечения семантической геоинтероперабельности. № 3
- Наумова В. В., Горячев И. Н., Дьяков С. В., Белоусов А. В., Платонов К. А.** Современные технологии формирования информационной инфраструктуры для поддержки и сопровождения научных геологических исследований на Дальнем Востоке России. № 7
- Пенджиев А. М., Пенжиев М. А.** Геоинформационные системы в солнечной энергетике Туркменистана. № 11

СИСТЕМЫ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ

- Гаврилов С. В., Иванова Г. А., Рыжова Д. И., Соловьев А. Н., Стемпковский А. Л.** Методы синтеза помехозащищенных комбинационных блоков. № 11
- Гурарий М. М., Русаков С. Г., Алиев Р. Р.** Применение методов схмотехнического моделирования для анализа биоэлектрических процессов в многоклеточных системах. № 9
- Кузьмина И. А.** Проектирование оптимальной структуры распределительной сети энергоснабжения с учетом перспектив развития города. № 11
- Ноженкова Л. Ф., Исаева О. С., Грузенко Е. А., Вогоровский Р. В., Колдырев А. Ю., Евсюков А. А.** Комплексная поддержка конструирования бортовых систем контроля и управления космических аппаратов на основе интеллектуальной имитационной модели. № 9
- Петросянц К. О., Харитонов И. А., Кожухов М. В., Самбурский Л. М.** Схмотехнические SPICE-модели биполярных и МОП-транзисторов для автоматизации проектирования радиационно стойких БИС. № 12
- Соловьев Р. А., Тельпухов Д. В., Рухлов В. С., Поперечный П. С.** Особенности проектирования модулярных многоходовых сумматоров с помощью современных САПР. № 9
- Стемпковский А. Л., Тельпухов Д. В., Соловьев Р. А., Рухлов В. С.** Тестовая система для сравнения алгоритмов, увеличивающих надежность комбинационных схем. № 12
- Струченков В. И.** Новый алгоритм поэлементного расчета трасс в САПР линейных сооружений. № 4

СИСТЕМЫ СВЯЗИ И ТЕЛЕКОММУНИКАЦИИ

- Милюковский А. Г., Данилюк А. Ю., Карутин С. Н., Глухов П. Б. Основы обеспечения целостности системы ГЛОНАСС. . № 8
- Перепелкин Д. А. Концептуальный подход динамического формирования трафика программно-конфигурируемых телекоммуникационных сетей с балансировкой нагрузки. . № 8

МЕТОДЫ И ТЕХНОЛОГИИ ОБРАБОТКИ ДАННЫХ

- Бочков М. В., Бойков П. Н. Инфодинамическая модель поиска пользователя в социальной сети. . № 1
- Костин В. Н., Данышин Д. В. Метод оценки глубины прогноза развития (эволюции) характеристик сложных систем на основе энтропийного подхода. . № 1
- Печенко И. С., Венгер О. В. Способ автоматизированного создания диаграмм последовательности операций для сценариев поведения системы, описанных в виде текста. . № 1

СТРУКТУРНЫЙ СИНТЕЗ И ПРИНЯТИЕ РЕШЕНИЙ

- Вишнеков А. В., Иванова Е. М. К вопросу об автоматизации процедур определения победителей конкурсов научно-технических проектов и стартапов. . № 10

БАЗЫ ДАННЫХ

- Цвященко Е. В. Анализ адекватности модели согласования реплик в конечном счете в базах данных NoSQL. . № 11

КОМПЬЮТЕРНАЯ ГРАФИКА

- Барский А. Б. "Живое" моделирование и объемная экранизация. . № 11

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ОРГАНИЗАЦИОННЫХ И СОЦИАЛЬНО- ЭКОНОМИЧЕСКИХ СИСТЕМАХ

- Андрейчикова О. Н., Вавилов А. А., Козырев А. Н. Исследование поведения потребителей на рынке программного обеспечения. . № 5
- Мамедова М. Г., Джабраилова З. Г., Мамедзаде Ф. Р. Методы управления согласованием спроса и предложения на рынке труда специалистов по информационным технологиям. № 4
- Мистров Л. Е. Метод синтеза стратегий управления конфликтной устойчивостью социально-экономических организаций. . № 4
- Спивак С. И., Кантор О. Г. Чебышевское приближение в задачах эконометрического моделирования. . № 10
- Таран В. В. Информационно-коммуникационные технологии и их социально-экономическое и культурологическое влияние на инновационно-ориентированное развитие. . № 3

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В БИОМЕДИЦИНСКИХ СИСТЕМАХ

- Абдуллаев Н. Т., Дышин О. А., Керимова М. И. Дифференциальная диагностика заболевания желудочно-кишечного тракта на основе мультифрактального анализа электрогастроэнтерографических сигналов. . № 11
- Захаров М. С. Динамика во времени вегетативных показателей на основе компьютерного спектрально-временного анализа ритмограмм сердца. . № 5

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ОБРАЗОВАНИИ

- Берчун Ю. В., Загидуллин Р. Ш. Комплексный подход к организации удаленного доступа к лабораторному оборудованию в учебном процессе. . № 3
- Лаптев В. В., Грачев Д. А. Семантическая модель программы и ее реализация. . № 3
- Черняховская Л. Р., Герасимова И. Б., Давлетбаева А. Р. Разработка системы нечеткого логического вывода для анализа результативности процесса дистанционного обучения. № 5

25 ЛЕТ НИИСИ РАН

- Барских М. Е., Бобков С. Г. Исследование влияния динамического предсказания ветвлений на производительность периферийных микропроцессоров НИИСИ РАН. . № 10
- Бетелин В. Б., Галатенко В. А., Костохин К. А. Контролируемое выполнение приложений на многопроцессорных платформах. . № 10
- Бобков С. Г., Сердин О. В. Проблемы унификации микросхем космического применения на примере международного проекта космического телескопа "ГАММА-400". . № 10

Журнал в журнале НЕЙРОСЕТЕВЫЕ ТЕХНОЛОГИИ

- Аникин В. И., Карманова А. А. Кластеризация и классификация многомерных данных клеточной нейронной сетью Кохонена. . № 6
- Барский А. Б. Обучаемые и самообучающиеся системы распознавания, управления и принятия решений на логических нейронных сетях. . № 10
- Галушкин А. И. Мемристоры в развитии высокопроизводительной вычислительной техники. . № 2
- Галушкин А. И. Нейрочипы и нейроморфные ЭВМ: проблемы моделирования. . № 12
- Горбатков С. А., Касимова Л. И. Нейросетевой итерационный метод построения динамических моделей вероятности риска банкротства при неполных данных. . № 12
- Гридин В. Н., Солодовников В. И., Евдокимов И. А. Нейросетевой алгоритм симметричного шифрования. . № 4
- Кульчин Ю. Н., Ким А. Ю., Ноткин Б. С., Люхтер А. Б. Обработка сигналов распределенной волоконно-оптической сети для распознавания динамических образов с применением нейронных сетей. . № 4
- Куравский Л. С., Мармалюк П. А., Юрьев Г. А., Думин П. Н. Методы численной идентификации марковских моделей и их сравнительный анализ. . № 10
- Лёзина И. В. Сравнительный анализ погрешности прогноза биржевых индексов методами экспоненциального сглаживания и нейросетевого моделирования. . № 8
- Осипов В. Ю. Ассоциативно-пространственная адресация к памяти рекуррентных нейронных сетей. . № 8
- Усков А. А., Жукова А. Г. Нейросетевая оценка сложности IDEF-диаграмм. . № 6
- Федосов В. В., Федосова А. В. Использование нейросетей для графической оценки загрязнения территории выбросами группы источников. . № 2

Адрес редакции:

107076, Москва, Стромьинский пер., 4

Телефон редакции журнала (499) 269-5510

E-mail: it@novtex.ru

Технический редактор *Е. В. Конова*.

Корректор *Е. В. Комиссарова*.

Сдано в набор 07.10.2015. Подписано в печать 19.11.2015. Формат 60×88 1/8. Бумага офсетная.

Усл. печ. л. 8,86. Заказ IT1215. Цена договорная.

Журнал зарегистрирован в Министерстве Российской Федерации по делам печати, телерадиовещания и средств массовых коммуникаций.

Свидетельство о регистрации ПИ № 77-15565 от 02 июня 2003 г.

Оригинал-макет ООО "Авансд солишнз". Отпечатано в ООО "Авансд солишнз".

119071, г. Москва, Ленинский пр-т, д. 19, стр. 1.