

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Том 21
2015
№ 6

ТЕОРЕТИЧЕСКИЙ И ПРИКЛАДНОЙ НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ

Издается с ноября 1995 г.

УЧРЕДИТЕЛЬ
Издательство "Новые технологии"

СОДЕРЖАНИЕ

ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

- Фомичев В. А., Разоренов А. А.** Значение теории К-представлений для исследований по автоматическому выявлению семантических ролей 403
- Борисовский П. А., Еремеев А. В.** Построение расписаний многопродуктового производства с использованием целочисленного линейного программирования и эволюционных вычислений 412
- Воробьев В. В., Паршикова Е. А.** Применение мультимножеств для оценки ситуации мобильным агентом. 421

МОДЕЛИРОВАНИЕ И ОПТИМИЗАЦИЯ

- Булычев Г. Г.** Моделирование динамического разрушения полых тел продуктами детонации 427

БЕЗОПАСНОСТЬ ИНФОРМАЦИИ

- Щеглов К. А., Щеглов А. Ю.** Технология защиты данных, обрабатываемых в распределенных информационных системах 433
- Имамвердиев Я. Н.** Нечеткая когнитивная модель стратегического управления информационной безопасностью электронного правительства. 440

ПРОГРАММНАЯ ИНЖЕНЕРИЯ

- Сметанин С. И., Игнатюк В. А., Евстифеев А. А.** Способ реализации программной веб-части системы спутникового мониторинга 448

ЦИФРОВАЯ ОБРАБОТКА СИГНАЛОВ И ИЗОБРАЖЕНИЙ

- Гетманов В. Г., Борзунов Г. И.** Алгоритм параллельных вычислений для задачи спектрально-временного анализа на базисных полигармонических функциях 456
- Табаков Ю. Г.** Модель и алгоритм обработки низкочастотного сигнала для тренажера на основе вейвлет-преобразований 464

Журнал в журнале

НЕЙРОСЕТЕВЫЕ ТЕХНОЛОГИИ

- Аникин В. И., Карманова А. А.** Кластеризация и классификация многомерных данных клеточной нейронной сетью Кохонена 470
- Усков А. А., Жукова А. Г.** Нейросетевая оценка сложности IDEF-диаграмм 478

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.

CONTENTS

INTELLIGENT SYSTEMS AND TECHNOLOGIES

- Fomichov V. A., Razorenov A. A.** The Significance of the K-Representations Theory for the Studies on Automatic Semantic Role Labeling 403
- Borisovsky P. A., Ereemeev A. V.** Production Scheduling of a Multi-Product Plant Using Integer Linear Programming and Evolutionary Computations 412
- Vorobev V. V., Parshikova E. A.** Application of Multisets for Assessment of the Situation by the Mobile Agent 421

MODELING AND OPTIMIZATION

- Bulychev G. G.** Numerical Modeling of Dynamical Destruction Hollow Bodies by Detonation Products. 427

CRYPTOSAFETY INFORMATION

- Shcheglov K. A., Shcheglov A. Yu.** Network Informational System Data Securing Technology 433
- Imamverdiyev Y. N.** A Fuzzy Cognitive Model for the Strategic Management of Information Security of E-Government 440

SOFTWARE ENGINEERING

- Smetanin S. I., Ignatyuk V. A., Evstifeev A. A.** Implementation of the Software Part of the System of Satellite Monitoring 448

DIGITAL PROCESSING OF SIGNALS AND IMAGES

- Getmanov V. G., Borzunov G. I.** The Algorithm of Parallel Calculations for a Problem of the Spectral-Time Analysis on Basic Polyharmonic Functions. 456
- Tabakov Yu. G.** Problems Processing LF Signals in of Intellectual Information Systems . 464

Journal-in-journal

NEUROTECHNOLOGIES

- Anikin V. I., Karmanova A. A.** Clustering and Classification of Multidimensional Data by Kohonen's Cellular Neural Network 470
- Uskov A. A., Zhukova A. G.** Neural Network Assessment Challenges IDEF-Charts . 478

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.
Kukhareno B. G., Cand. Sci. (Phys.-Math.)
Ljvovich 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"

В. А. Фомичев, д-р техн. наук, проф., e-mail: vfomichov@hse.ru,

А. А. Разоренов, аспирант, e-mail: razorenov@mail.ru,

Национальный исследовательский университет "Высшая школа экономики", Москва

Значение теории К-представлений для исследований по автоматическому выявлению семантических ролей

Показана неполнота теоретических основ ветвей компьютерной семантики, называемых Выявлением семантических ролей (ВСР) и Фреймово-семантическим анализом (ФСА), как следствие кажущегося отсутствия семантического формализма, позволяющего описывать смысловую структуру сложных предложений и дискурсов из произвольных предметных областей. Сделан вывод о том, что теория К-представлений (концептуальных представлений) предоставляет такой формализм, определяя новый класс формальных языков — класс СК-языков (стандартных концептуальных языков). Проиллюстрированы некоторые новые выразительные механизмы СК-языков. Изложены центральные идеи метода выполнения семантико-синтаксического анализа текстов на естественном языке (ЕЯ), предложенного теорией К-представлений. Для построения семантических представлений текстов использован класс СК-языков. В заключительной части статьи рассматривается применение метода к проектированию ЕЯ-интерфейсов для управления программным обеспечением. Разработан файловый менеджер с ЕЯ-интерфейсом NLC-1 (Natural Language Commander — версия 1), система реализована на функциональном языке программирования Haskell.

Ключевые слова: естественный язык, семантико-синтаксический анализ текстов, выявление семантических ролей, фреймово-семантический анализ, семантическое представление, теория К-представлений, СК-язык, лингвистическая база данных, управление программным обеспечением, естественно-языковое управление файловой системой, Natural Language Commander

Введение

Исследования, направленные на автоматическое извлечение смысла из текстов на естественном языке (ЕЯ), т. е. из текстов на русском, английском и других языках, проводились с конца 1960-х годов и сформировали важное направление в области искусственного интеллекта, называемое компьютерной обработкой естественного языка (КОЕЯ). В англоязычной научной литературе используется термин Natural Language Processing (NLP). Уменьшение интереса к проблеме в 1990-х сменилось в 2000-х быстрым ростом числа проектов в этой области. Представляется, что основные причины данной ситуации заключаются в том, что: (а) из-за бурного роста Всемирного Веба (World Wide Web) стал технически доступен огромный объем информации в виде естественно-языковых текстов (ЕЯ-текстов), и это вызвало потребность их смысловой обработки, позволяющей обычному, непрограммирующему пользователю осуществлять поиск информации в текстах по определенным критериям; (б) было разработано семейство онтологий, базирующихся в Вебе (Web-based ontologies), что повлекло за собой возможность использовать накопленные знания для

семантической обработки ЕЯ-текстов прикладными компьютерными системами; (в) возникла необходимость в естественно-языковых интерфейсах (ЕЯ-интерфейсах) для взаимодействия с онтологиями, разработанными в рамках проекта Семантического Веба (Semantic Web) и сформированных с помощью языков RDF, RDFS и OWL [1, 2], а также для взаимодействия с Системой взаимосвязанных открытых данных (Linked Open Data) [3, 4].

В сентябре 2012 г. в Германии, в замке Дагштуль, состоялся семинар по Мультилингвистическому Семантическому Вебу (МСВ). Труды этого семинара [5] содержат следующие данные: в 2010 г. число пользователей Интернета, говорящих не на английском языке, в три раза превысило число англоговорящих пользователей (1430 млн против 536 млн). Поэтому проблема создания МСВ чрезвычайно актуальна. Эта проблема выдвигает новые требования к компьютерной семантике. Представляется разумным анализировать эти новые требования в контексте проблем, с которыми столкнулась компьютерная лингвистика в целом.

Анализ многих публикаций в области КОЕЯ показывает существование разрыва (очень часто —

большого разрыва) между используемыми в проектах теоретическими инструментами и реальными потребностями исследуемых проблем. Рассмотрим только один пример. Лингвистический процессор BLUE (Boeing Language Understanding Engine) был разработан для компании Боинг как передовой инструмент обработки информации. Эта система может строить семантические представления (СП) предложений многих видов [6]. Однако мы узнаем из второго раздела той же статьи [6], что система BLUE "разрешает высказываниям быть аргументами других высказываний, образуя встроенные структуры". Так, система строит СП фразы "Мужчина хотел покинуть дом". Этот шаг немедленно выводит нас за пределы логики предикатов первого порядка (ЛППП). Дело в том, что атомарная формула ЛППП не может включать аргументы, являющиеся формальными семантическими образами инфинитивных конструкций ("покинуть дом" и т. п.). Поэтому, по существу, компьютерная система BLUE компании Боинг не имеет адекватной теоретической основы.

Эта проблема является частным случаем фундаментальной проблемы, с которой столкнулось научное направление, называемое *Выявлением семантических ролей (Semantic Role Labeling)* [7]. Главной решаемой задачей является нахождение семантико-синтаксических отношений (называемых семантическими ролями или, во многих публикациях, тематическими, ролями) между глагольными формами (или некоторыми другими предикативными словами и сочетаниями) и зависимыми в предложении словами или сочетаниями слов. Например, можно выявить семантические роли *Агент*, *Феномен* и *Время* в предложении "Российский нобелевский лауреат Иван Петрович Павлов открыл условные рефлексы в начале XX столетия".

Цель алгоритмов выявления семантических ролей заключается, во-первых, в нахождении ролей, реализованных в рассматриваемой фразе, и, во-вторых, в построении формального выражения, называемого СП фразы, для того чтобы обработать очередную фразу в контексте диалога (либо предыдущей части связного текста) и базы знаний о предметной области (онтологии). Фундаментальной проблемой этого направления является *кажущееся отсутствие* формальных средств, позволяющих отражать семантическую структуру произвольных предложений. В частности, направление Выявление семантических ролей (ВСП) не предлагает эффективные формальные средства для построения СП предложений со сложной прямой или косвенной речью, составными обозначениями множеств, инфинитивными конструкциями. Последнее относится, например, к предложению "Павел решил покинуть фирму "Старт" и постараться получить более высокую рабочую позицию в компании "Восход".

Анализируя развитие компьютерной лингвистики (КЛ) в последние два десятилетия, можно заметить сдвиг к многочисленным инженерным проектам для решения частных практических задач и недостаток внимания к фундаментальным исследованиям.

Представляется, что одно из наиболее ярких описаний недавней и текущей ситуаций в компьютерной лингвистике было дано доктором Ш. Винтнером из отделения информатики университета Хайфы (Израиль) [8]. Отправной точкой для Ш. Винтнера была высокая оценка значения математических теорий для развития многих инженерных областей. Например, на основе аэродинамики конструируются самолеты, а гидродинамика помогает проектировать корабли. В связи с этим в [8] задаются следующие вопросы: "Какая область науки лежит в основе инженерии компьютерных систем обработки ЕЯ? Какова теоретическая инфраструктура, на основе которой мы разрабатываем приложения? И какой вид математики нужен для обоснования рассуждений о языках человека?"

Необходимость разработки комплексного формального подхода для создания МСВ делает чрезвычайно актуальным вопрос о математических основах компьютерной семантики, являющейся ядром современной КЛ.

Обстоятельный ответ на вопрос об адекватных математических основах компьютерной семантики был опубликован уже через несколько месяцев (в том же году) после опубликования указанных фундаментальных вопросов. Такой ответ был дан теорией К-представлений (концептуальных представлений) — оригинальной теорией проектирования семантико-синтаксических анализаторов ЕЯ-текстов с широким применением формальных средств описания входных, промежуточных и выходных данных. Основная часть этой теории была опубликована на английском языке в декабре 2009 г. в монографии [9], развивающей идеи монографий [10, 11].

Первая цель данной статьи — показать, что ветви КОЕЯ, называемые Выявлением семантических ролей (Semantic Role Labeling) и Фреймово-семантическим анализом (Frame-Semantic Parsing) [12], должны быть дополнены семантическим формализмом, удобным для построения семантических представлений сложных предложений и дискурсов.

Вторая цель — показать, что семантический формализм такого рода уже существует, это класс СК-языков (стандартных концептуальных языков), определяемый теорией К-представлений.

Наконец, третья цель заключается в демонстрации непосредственной практической значимости одного из алгоритмов семантико-синтаксического анализа, предлагаемых теорией К-представлений. В частности, этот алгоритм нашел применение при разработке ЕЯ-интерфейса прикладной интеллектуальной системы, позволяющей управлять файловой системой компьютера.

Состояние исследований по автоматическому выявлению семантических ролей

Задача извлечения смысла из ЕЯ-текстов привлекла большое внимание специалистов во многих областях применений в первой половине 2000-х годов. Эта задача, предполагающая конструирование полного или частичного формального представления смысла текста, стимулировала появление целого ряда исследовательских проектов во всем мире. Главным направлением в этой области представляются взаимосвязанные научные ветви, называемые Выявлением семантических ролей (ВСР) и Фреймово-семантическим анализом (ФСА).

Как было отмечено выше, цель ВСР заключается в разработке методов и компьютерных программ, направленных на нахождение семантических ролей (другими словами — тематических ролей), реализующихся в парах определенных слов или словосочетаний. Важным бинарным событием в развитии этого научного направления стала публикация пионерской работы [13] о компьютерной программе для статистического выявления семантических ролей и создание депозитария аннотаций PropBank [14]. Эти две публикации послужили отправной точкой для разработки семейства прикладных компьютерных систем, предназначенных для нахождения предикатно-аргументных структур, отражающих семантику предложений и коротких дискурсов.

Формальные аннотации из депозитария PropBank состоят из деревьев синтаксического разбора, представленных в разделе депозитария Penn Treebank [15] с текстами из журнала Wall Street Journal и дополненных предикатно-аргументными структурами для глаголов. Депозитарий PropBank использует базовые роли ARG0—ARG5, и эти роли имеют разные интерпретации для разных предикатов. В предыдущее десятилетие было проведено много исследований в области автоматического выявления семантических ролей, использующих нотацию депозитария PropBank [7, 16, 17]. Проблема использования предикатно-аргументных структур заключается в том, что базовые роли ARG2—ARG5 служат многим различным целям для разных глаголов [18].

Выход из этой ситуации был предложен ветвью компьютерной обработки ЕЯ, названной Фреймово-семантическим анализом (ФСА, в англоязычной литературе Frame-Semantic Parsing) и тесно связанной с ветвью ВСР (многочисленные библиографические ссылки можно найти в статье [12]). Основой ФСА является лингвистический ресурс FrameNet [19]. Этот ресурс хранит значительный объем информации о лексической семантике и предикатно-аргументной семантике предложений английского языка. Лексикон ресурса FrameNet содержит семантические фреймы, каждый из которых включает список лексических единиц — ассоциированных слов и словосочетаний, которые могут сигнализи-

ровать о возможной реализации рассматриваемого семантического фрейма в ЕЯ-выражении. Кроме того, каждый семантический фрейм из ресурса FrameNet указывает на несколько ролей, соответствующих различным аспектам сценария, представляемого фреймом. Мишенями (или тагитами — от английского слова target) называют предикатные выражения (глаголы и т. д.), сигнализирующие о семантических фреймах, а аргументом называют такое слово или выражение, значение которого связано с определенной мишенью (тагитом) некоторой семантической ролью.

Например, фрейм JUDGMENT (Осуждение) из базы данных (БД) FrameNet содержит вручную аннотированное предложение "She blames the Government for failing to do enough to help" ("Она обвиняет правительство за недостаточные усилия помочь"). В этом предложении выделены следующие семантические роли: *Judge* (Осуждающий) в паре (She, blames), *Evaluate* (Осуждаемый) в паре (blames, the Government), *Reason* (Причина) в паре (blames, for failing to do enough to help). В БД FrameNet рассматриваемому предложению будет соответствовать следующая семантико-синтаксическая аннотация:

[*Judge* She] **blames** [*Evaluate* the Government]
[*Reason* for failing to do enough to help].

По сравнению с депозитарием PropBank, содержащим глагольные предикаты, FrameNet включает как их, так и прилагательные, наречия и предлоги.

Центральные идеи теории К-представлений

Постановка задачи. Алгоритмы и компьютерные программы, разработанные в областях КОЕЯ, называемых ВСР и ФСА, помогли преодолеть часть трудностей, связанных со сравнением содержания запроса пользователя Веба с содержанием фрагмента информационного ресурса. Например, БД FrameNet включает семантический фрейм TRANSFER, описывающий взаимосвязи между семантическими ролями мишеней (тагитов) *send* (послать) и *receive* (получить). Поэтому, используя фрейм TRANSFER, можно ответить на вопрос "Какие партии послали избирателям открепительные бюллетени?" после обработки текста "Избиратели как Демократической, так и Республиканской партий получили открепительные бюллетени от своих партий". Однако значительная часть вопросов, касающихся автоматической семантической обработки ЕЯ-текстов, все еще остается открытой.

Пример 1. Пусть V_1 = "Получала ли фирма "Радуга" в этом месяце аксессуары от ЗАО "Весна"?", вопрос V_1 датирован 24 апреля 2015 г., и некоторая полнотекстовая БД Db_1 содержит коммерческий контракт, включающий текст T_1 = "ЗАО "Весна" обязуется поставить фирме "Радуга" три однотонных контейнера с аксессуарами к 15 апреля 2015 г.". Тогда последовательность разумных шагов формиро-

вания ответа "Скорее всего, да" гипотетической компьютерной интеллектуальной системой включает:

Шаг 1. Построение СП вопроса В1.

Шаг 2. Построение СП текста Т1.

Шаг 3. Нахождение в базе знаний Db2 записи, передающей смысл М1 = "Если какая-то организация Х1 обязуется в юридическом документе, датированном моментом t1, выполнить определенное действие Х2 к дате t2, то обычно организация Х1 выполняет действие Х2 в интервале времени t3, где t1 предшествует началу t3, и конец t3 близок к дате t2 и раньше t2 либо совпадает с t2".

Пример 2. Пусть В2 = "Какие достижения были у компании "Горизонт" в прошлом году?". Для обработки вопросов о достижении цели нам нужно выделить, изучить и разделить на несколько групп цели деятельности организации. Примеры таких целей: "Выпуск нового продукта", "Открытие нового офиса компании", "Увеличение прибыли". Такая информация должна храниться в специальной базе целей [20].

Эти два примера и многие другие примеры показывают, что для продвижения вперед в теории семантически-ориентированных систем обработки ЕЯ необходимо создать предметно-независимый семантический формализм, удобный для описания: (а) семантической структуры предложений, включающих, в частности, инфинитивные и герундиальные (для английского языка) конструкции, выражающие цели, обязательства, команды и т. д., придаточные предложения цели, сложную прямую и косвенную речь, составные обозначения множеств; (б) семантической структуры связных текстов (или дискурсов), в том числе включающих ссылки на ранее упомянутые объекты и на смысл фраз и более крупных частей текста; (в) представления фрагментов знаний о мире, в том числе определений понятий; (г) формальных представлений простых и составных целей людей, организаций и роботов.

Такого сочетания выразительных механизмов не предлагают теория представления дискурсов (Discourse Representation Theory), теория концептуальных графов (Theory of Conceptual Graphs), эпизодическая логика (Episodic Logic), теория расширенных семантических сетей, компьютерная семантика русского языка и теория неоднородных семантических сетей.

Краткая характеристика теории К-представлений.

Решение указанной проблемы предлагается теорией К-представлений (концептуальных представлений). Это оригинальная теория проектирования семантико-синтаксических анализаторов ЕЯ-текстов с широким применением формальных средств описания входных, промежуточных и выходных данных. Теория К-представлений (ТКП) представлена в большой серии публикаций на русском и английском языках, в том числе в работах [9—11, 20—27].

ТКП вносит также вклад в расширение теоретических основ МСВ [9, 25—27]. Первоначально ТКП называлась теорией К-исчислений и К-языков.

Сущность базовой математической модели. Базовая модель ТКП описывает систему, состоящую из 10 частичных операций на концептуальных структурах. Применяя эти операции шаг за шагом, можно построить формальное представление структурированного значения, или СП, произвольно сложного предложения или связного текста (дискурса) на русском, английском, немецком, французском и других языках. Модель, в частности, определяет новый класс формальных языков — класс СК-языков [9—11, 21—23]. По сравнению с указанными выше подходами к формальному представлению значений (или смысла, содержания) ЕЯ-текстов общим преимуществом СК-языков является уникальный набор новых выразительных механизмов, открывающих возможности моделирования семантической структуры словосочетаний и текстов следующих видов (при сохранении выразительных средств многосортной логики предикатов первого порядка): (а) инфинитивных и герундиальных конструкций (выражающих действия, цели, обязательства и т. д.); (б) фраз со сложными придаточными предложениями, в том числе с прямой и косвенной речью, с придаточными цели; (в) произвольно сложных составных обозначений объектов, ситуаций, множеств и понятий; (г) дискурсов со ссылками на смысл предыдущих или последующих фраз и более крупных частей текста; (д) обозначений упорядоченных наборов объектов [9—11, 21—23, 27].

Некоторые выразительные возможности СК-языков. Пусть $Exrg$ — выражение на ЕЯ, и Sem — некоторое СП этого выражения. Тогда будем говорить, что Sem — возможное К-представление (КП) выражения $Exrg$, если Sem является выражением некоторого СК-языка (т. е. СК-языка в некотором концептуальном базисе). В примерах КП входного текста Т будет значением строковой переменной $Semrepr$ (Semantic Representation).

Представление смысла предложений с прямой речью. Пусть Т1 = "Когда мистер Питер Смит сообщил, что он посетит Монпелье в апреле?" Тогда $Semrepr = \text{Вопрос}(t1, \text{Ситуация}(e1, \text{информирование1} * (\text{Время, нек момент} * (\text{Раньше, \#сейчас\#}): t1) (\text{Агент1, нек чел} * (\text{Имя, "Питер"})) (\text{Фамилия, "Смит"}): x1) (\text{Информ-содержание, Ситуация}(e2, \text{визит1} * (\text{Агент1, x1}) (\text{Место2, нек город1} * (\text{Наименование1, "Монпелье"}): x2) (\text{Время, Ближайший-будущий-месяц}(Апрель, \#сейчас\#))))))$.

Представление смысла сложноподчиненных предложений с придаточными цели. Пусть Т2 = "Мистер Питер Смит, вице-президент компании "Rainbow" заявил вчера, что посетит Монпелье в апреле, чтобы подписать соглашение с компанией CIRAD". Тогда $Semrepr = \text{Ситуация}(e1, \text{информирование1} * (\text{Время,$

Предыдущий-день (#сейчас#)) (Агент1, нек чел * (Имя, "Питер") (Фамилия, "Смит"): x1) (Информ-содержание, Ситуация (e2, визит1 * (Агент1, x1) (Место2, нек город * (Наименование1, "Montpellier"): x2) (Время, Ближайший-будущий-месяц (Апрель, #сейчас#)) (Цель, предписание2 * (Информ-объект, нек соглашение1: x3) (Партнер2, нек компания1 * (Наименование1, "CIRAD"): x4))))).

Отображение смысла дискурсов. Пусть ТЗ = "Все гранулоциты являются полиморфонуклеарными; это значит, что у них многодольные ядра". Тогда $Semrepr = (Свойство (произв\ гранулоцит : x1, полиморфонуклеарн): P1 \wedge \text{Пояснение} (P1, \text{Если-то} (Ситуация (e1, обладание1 * (Субъект1, x1) (Объект1, нек ядро : x2)), Свойство (x2, многодольн))))).$

Здесь P1 — переменная, используемая как метка смысла первой фразы из ТЗ; идентификаторы *Субъект1*, *Объект1* обозначают семантические роли.

Сущность метода семантико-синтаксического анализа текстов, предлагаемого теорией К-представлений

ТКП не только предложила класс СК-языков для построения СП сложных предложений и дискурсов, но и использовала определение класса СК-языков для создания широко применимой математической модели лингвистической базы данных (ЛБД) (см. гл. 6 книги [10] и гл. 7 книги [9]). Эта модель ЛБД лежала в основе разработки сильно структурированных алгоритмов семантико-синтаксического анализа (ССА) ЕЯ-текстов *SemSyn* и *SemSynt1*. Модель и алгоритмы применены при разработке рекомендательной системы с ЕЯ-интерфейсом [28], информационно-поисковой системы, реализующей семантический поиск специалистов по ЕЯ-описаниям их компетенций [29] и при проектировании системы управления файлами с ЕЯ-интерфейсом [30, 31].

В монографиях [9, 10] предлагается альтернативный подход к выявлению семантических ролей. В отличие от идеологии статистических подходов к этой проблеме, предполагается, что для многих практических задач возможно применение алгоритмов, точно отражающих на выходе семантико-синтаксическую структуру входного текста. В частности, это важно для разработки ЕЯ-интерфейсов прикладных систем, управляющих выполнением определенных действий (см. заключительный раздел данной статьи).

Рассмотрим центральные идеи метода анализа текстов, предложенного в [9] и лежавшего в основе разработки мультилингвистического алгоритма ССА *SemSynt1*. Он является композицией алгоритмов, называемых *BuildMatr1* и *BuidlSem1*. Алгоритм *BuildMatr1* можно квалифицировать как *оригинальный алгоритм выявления семантических ролей*. Входные тексты могут быть вопросами многих видов, командами и описаниями фактов (ситуаций)

из практически интересных подязыков русского, английского и немецкого языков. Выход алгоритма *BuildMatr1* (точнее, главная часть выходных данных) является специальной строково-числовой матрицей *Matr*, называемой матричным семантико-синтаксическим представлением (МССП) входного текста. Матрица *Matr* динамически связана со вспомогательной структурой данных — двумерным массивом *ArIs*. Этот массив устанавливает соответствие между лексическими единицами из входного текста и семантическими единицами. Если элементарная значащая единица текста (или токен) *wd* имеет *N* различных значений, массив *ArIs* включает *N* последовательных строк, где для $k = 1, \dots, N$ *k*-я строка хранит сведения, относящиеся к *k*-му значению единицы *wd*.

Конфигурация МССП *Matr* изменяется в ходе семантико-синтаксической обработки текста. Каждая конфигурация определяет, в частности, размеченный ориентированный граф с вершинами, являющимися выделенными элементарными значащими единицами текста (токенами), и задает отображение из подмножества вершин графа, соответствующих лексическим единицам, в множество значений (смыслов), связанных с данными лексическими единицами через массив *ArIs*.

Рис. 1 иллюстрирует эту ситуацию для команды "Download the green container on the platform" ("Сгрузи зеленый контейнер на платформу"). Здесь V1[1] — значение *downloading1* (считывание файла), V1[2] — значение *downloading2* (выгрузка перемещаемого физического объекта); V2[1] — значение *green-colour* (зеленый цвет), V2[2] — значение *not-ripe* (незрелый), V2[3] — значение *a-member-of-green-movement* (участник экологического движения зеленых); V3[1] — значение *thing-container* (контейнер-вещь), V3[2] — значение *data-structure-of-RDF* (контейнер — структура данных языка RDF); V4[1] — значение *computer-platform* (компьютерная платформа), V4[2] — значение *station-platform* (станционная платформа), V4[3] — значение *political-platform* (политическая платформа). Рис. 2 иллюстрирует финальную ситуацию.

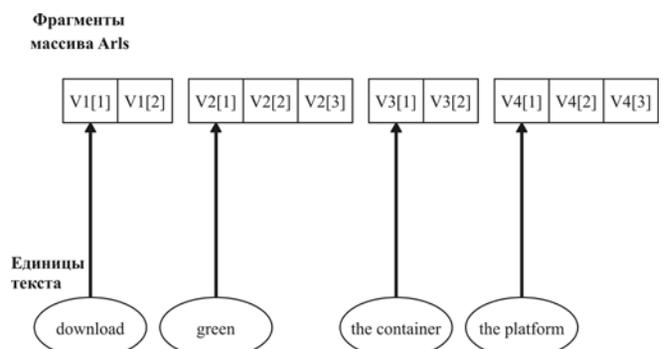


Рис. 1. Начальные граф и соответствие, заданные МССП *Matr*

Фрагменты
массива ArIs

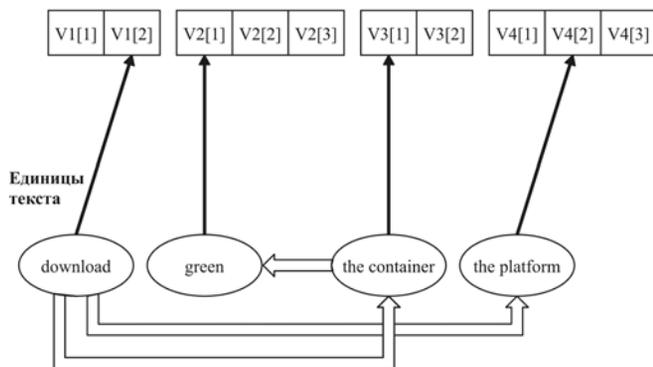


Рис. 2. Финальные граф и соответствие, заданные МССП matr

Выход (его главная часть Matr) алгоритма выявления семантических ролей BuildMatr1 является входом алгоритма семантической сборки BuildSem1. Этот второй алгоритм преобразует информацию, хранимую матрицей Matr, в возможное СП входного текста, являющееся его К-представлением.

Естественно-языковой интерфейс для управления файловой системой

Проблемы использования графических интерфейсов. В последние 15—20 лет стандартом пользовательского интерфейса де-факто является графический интерфейс пользователя (ГИП). Эволюционное усложнение программного обеспечения (ПО) по мере роста мощности вычислительной техники привело к тому, что ГИП стал все более и более усложняться, все чаще сталкиваясь с ограниченными возможностями отображения графики на экране. Это привело к двум неприятным последствиям:

- часть возможностей ПО оказывается неизвестной пользователю, так как он их не находит в бесконечных меню, вложенных друг в друга элементах интерфейса и открывающихся друг за другом окнах;
- поиск файлов по папкам или программ в меню системы все чаще превращается в утомительный процесс. Пользователь затрачивает длительное время на поиск или (если он помнит, где находится требуемый ему элемент интерфейса: кнопка, пункт меню, флажок и др.) на доступ к данному элементу.

Анализ показывает, что перспективным выходом из данной сложной ситуации является разработка интерфейсов, предоставляющих пользователю прикладной компьютерной системы (ПКС) возможность взаимодействия с ней на ограниченном ЕЯ — русском, английском и др. При этом ЕЯ-интерфейсы для управления ПКС рассматриваются как дополнения и/или альтернативы графическим интерфейсам пользователя. Большая часть работ в области ЕЯ-интерфейсов посвящена интерфейсам

к базам данных. Тема управления программным обеспечением исследована слабее, однако и в этом направлении ведутся работы, например [32].

В настоящее время имеются различные программы, использующие ЕЯ-интерфейс для управления программным обеспечением, в том числе: (а) Siri — средство речевого управления телефонами iPhone 4S/5 [33]; (б) ее русскоязычный аналог: *Собеседник HD* [34]; (в) *Ubiquity* — дополнение к Web-обозревателю Firefox [35]; (г) *Braina Project* — программное обеспечение для управления компьютером в целом [36]; (д) *NLUI Server* — серверное программное обеспечение для построения сценариев взаимодействия с программами на ЕЯ и встраивания поддержки ЕЯ-интерфейса в приложения [37].

Описанные решения выполняют простые одноактные команды: "Открыть файл [имя файла]", "Создать папку [наименование папки]", "Скопировать файл [имя файла] в папку [имя папки]" и т. д. В то же время диапазон пользовательских команд неизмеримо шире, он может включать в себя такие сложные конструкции, как "Скопировать все текстовые и музыкальные файлы, созданные в этом году, из папки А в папку Б, запаковать их и отправить по электронной почте на адрес `someperson@example.org`". Учень все многообразие возможных пользовательских команд, используя прямолинейный подход к обработке входных предписаний, невозможно.

Для создания ЕЯ-интерфейсов управления компьютерными системами необходимы разработка формального аппарата, описывающего управление ПО с помощью естественно-языковых предписаний (команд), а также программная реализация управления на его основе.

Метод извлечения смысловых структур из ЕЯ-текстов, охарактеризованный выше, был модифицирован [30, 31] и использован для разработки алгоритма семантико-синтаксического анализа SemSynt2. Отправной точкой стал алгоритм SemSynt1, описанный в гл. 9 и 10 монографии [9]. Мы разработали первую версию системы Natural Language Commander (NLC-1) — файлового менеджера с ЕЯ-интерфейсом. Его основным алгоритмом является SemSynt2. Главная цель проекта — создать работающую теорию построения ЕЯ-интерфейсов для прикладных компьютерных систем и практически полезную программную систему для человеко-машинного взаимодействия. Данное приложение конструирует первичное К-представление (КП), соответствующее инструкции, введенной пользователем, и преобразует его во вторичное КП.

В этом вторичном КП сложные понятия ("музыка", "видео" и т. д.) выражены через базовые понятия. Например, понятие "музыка" заменяется на "файл с расширением "mp3", "ogg" или "acc". Затем приложение трансформирует вторичное КП в сценарий командной оболочки операционной системы — в настоящее время это Bourne-again shell

(bash) и для Unix-подобных систем и для ОС семейства Windows — и исполняет данный сценарий.

Пример обработки команды. Рассмотрим обработку программой NLC-1 инструкции "Скопировать файлы из папки "My Video" в папку с названием "Video" на сетевой диск, если их размер не превышает свободное место на сетевом диске". Этой инструкции соответствует следующее первичное КП, сформированное алгоритмом SemSynt2:

*Если-то (Меньше (Размер (все фильм * (Расположение, нек папка1 * (Наименование1, "My video")):o1), Свободное-место(нек диск1 * (Тип1, Сетевой1):o2)), Команда(#Оператор#, #Исполнитель#, #сейчас#, Копирование1 * (Источник1, o1)(Назначение1, нек Папка1 * (Наименование1, ("Video"))(Расположение, o2))).*

В этом случае, если база знаний NLC-1 содержит правила преобразования *Фильм* \vdash *Файл* * (*Расширение1*, ("avi" OR "mov" OR "mkv")), *диск1* * (*Тип*, *Сетевой*) \vdash *диск1* * (*Наименование1*, "N"), система NLC-1 преобразует первичное КП инструкции пользователя во вторичное КП вида

*Если-то (Меньше (Размер (все Файл1 * (Расширение1, ("avi" OR "mov" OR "mkv")) (Расположение, нек папка1 * (Наименование1, "My video")):o1), Свободное-место (нек диск1 * (Наименование1, "N"): o2)), Команда (#Оператор#, #Исполнитель#, #сейчас#, Копирование1 * (Источник1, o1)(Назначение1, нек папка1 * (Наименование1, ("Video")) (Расположение, o2))).*

Тогда итоговый сценарий командной оболочки будет иметь вид *if [\$(du -cb "Download/*.avi" "Download/*.mov" "Download/*.mkv" | grep total) sed -e "s/ \.s.*\$/g") -le \$(free -space N:)]; then cp "Download/*.avi" "Download/*.mov" "Download/*.mkv" "N:/Video".fi.*

Разработанное на языке программирования Haskell [38—40] приложение NLC-1 является гибким и расширяемым приложением. Оно может быть настроено исследователем для различных областей применения и различных командных оболочек.

Заключение

Существенно расширены теоретические основы ветвей компьютерной семантики, называемых Выявлением семантических ролей и Фреймово-семантическим анализом. Класс СК-языков открывает новые перспективы для формализации лексической семантики, представления семантического содержания сложных (по-видимому, практически произвольных) предложений и дискурсов на ЕЯ, а также для создания моделей онтологий нового поколения и баз целей в самых разных научных центрах. Реализация системы NLC-1 с русскоязычным интерфейсом показывает, что метод и алгоритмы семантико-синтаксического анализа, предложенные в [9, 10, 24], являются полезными для приложений и обогащают ВСП и ФСА альтернативным подходом к автоматическому выявлению семантических ролей.

Многочисленные существующие проекты КОЕЯ получили адекватный теоретический аппарат для формального описания семантических структур предложений и дискурсов. Для разработки Мультилингвистического Семантического Веба также очень важно, что СК-языки дают удобный промежуточный уровень для перехода от входного ЕЯ-текста к выражениям языков RDF, RDFS и OWL из онтологий (см. гл. 6 монографии [9]).

Список литературы

1. Wang C., Xiong M., Zhou Q., Yu Y. PANTO: a portable NL-interface to ontologies // 4th European Semantic Web Conference Proceedings. Springer, 2007. P. 473—487.
2. Cimiano P., Haase P. et al. Towards Portable Natural Language Interfaces to Knowledge Bases — the Case of the ORAKEL System // Data and Knowledge Engineering. 2008. V. 65. N. 2. P. 325—354.
3. Aggarwal N., Polajnar T., Buitelaar P. Cross-Lingual Natural Language Querying over the Web of Data // Metais E., Mezziane F., Saraee M., Sugumaran V., Vadera S. (eds.). Natural Language Processing and Information Systems. 18th Intern. Conference on Applications of Natural Language to Information Systems, NLDB 2013, Salford, UK, June 2013. Proceedings. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer. 2013. V. 7934. P. 152—163.
4. Ferre S. SQUALL: A Controlled Natural Language as Expressive as SPARQL 1.1 // Metais E., Mezziane F., Saraee M., Sugumaran V., Vadera S. (eds.). Natural Language Processing and Information Systems. NLDB 2013. Berlin. Proceedings. Lecture Notes in Computer Science. Salford, UK, June 2013. Heidelberg: Springer. 2013. V. 7934. P. 114—125.
5. Buitelaar P., Choi K.-S., Cimiano P., Hovy E. H. (Eds.) Report from Dagstuhl Seminar 12362 "The Multilingual Semantic Web" (2—9 September, 2012). Schloss Dagstuhl: Leibniz-Zentrum fuer Informatik. 2012. 94 p.
6. Clark P., Harrison P. Boeing's NLP System and the Challenges of Semantic Representation // Proc. SIGSEM Symposium on Text Processing (STEP'08), Venice, Italy, 2008. ACL. 2008. P. 263—276.
7. Marquez L., Carreras X., Litkowski K. C., Stevenson S. Semantic Role Labeling: an Introduction to the Special Issue // Computational Linguistics. 2008. V. 34, N. 2. P. 145—159.
8. Wintner S. What Science Underlies Natural Language Engineering? // Computational Linguistics. 2009. V. 35, N. 4. P. 641—644.
9. Fomichev V. A. Semantics-Oriented Natural Language Processing: Mathematical Models and Algorithms. New York, Dordrecht, Heidelberg, London: Springer, 2010. 354 p.
10. Фомичев В. А. Формализация проектирования лингвистических процессов. М.: МАКС Пресс, 2005. 368 с.
11. Фомичев В. А. Математические основы представления содержания посланий компьютерных интеллектуальных агентов. М.: ГУ-ВШЭ, ТЕИС. 2007. 176 с.
12. Das D., Chen D., Martins A. F. T., Schneider N., Smith N. A. Frame-Semantic Parsing // Computational Linguistics. 2014. V. 40, N. 1. P. 9—56.
13. Gildea D., Jurafsky D. Automatic Labeling of Semantic Roles // Computational Linguistics. 2002. V. 28, N. 3. P. 245—288.
14. Kingsbury P., Palmer M. 2002. From TreeBank to PropBank // Proceeding of Third Intern. Conference on Language Resources and Evaluation — LREC. 2002, Las Palmas, Spain. 2002. P. 1989—1993.
15. Marcus M. P., Marcinkiewicz M. A., Santorini B. Building a large annotated corpus of English: the Penn TreeBank // Computational Linguistics. 1993. V. 19, N. 2.
16. Punyakanok V., Roth D., Yih W. T. The importance of syntactic parsing and inferencing in semantic role labeling // Computational Linguistics. 2008. V. 34, N. 2. P. 257—287.
17. Blanco E., Moldovan D. Leveraging verb-argument structures to infer semantic relations // Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, Gothenburg, Sweden, April 26—30, 2014. ACL, 2014. P. 145—154.
18. Yi S., Loper E., Palmer M. Can semantic roles generalize across genres? // Proceedings of the Human Language Technologies

Conference of the North American Chapter of the ACL. Rochester, NY, 2007. P. 548–555.

19. **Fillmore C. R., Johnson C. R., Petruck M. R. L.** Background to FrameNet // International Journal of Lexicography. 2003. V. 16, N. 3. P. 235–250.

20. **Fomichov V. A., Kirillov A. V.** A Formal Model for Constructing Semantic Expansions of the Search Requests about the Achievements and Failures // Artificial Intelligence: Methodology, Systems, and Applications / Ramsay A., Agre G. (Eds.). Lecture Notes in Computer Science. V. 7557. Berlin, Heidelberg: Springer, 2012. P. 296–304.

21. **Fomichov V. A.** A Mathematical Model for Describing Structured Items of Conceptual Level // Informatica. An International Journal of Computing and Informatics (Slovenia). 1996. V. 20, N 1. P. 5–32.

22. **Фомичев В. А.** Математические основы представления смысла текстов для разработки лингвистических информационных технологий. Часть I // Информационные технологии. 2002. № 10. С. 16–25.

23. **Фомичев В. А.** Математические основы представления смысла текстов для разработки лингвистических информационных технологий. Часть II // Информационные технологии. 2002. № 11. С. 34–45.

24. **Фомичев В. А.** Новый метод преобразования естественно-языковых текстов в семантические представления // Информационные технологии. 2005. № 10. С. 25–35.

25. **Fomichov V. A.** A Comprehensive Mathematical Framework for Bridging a Gap between Two Approaches to Creating a Meaning-Understanding Web // Intern. Journal of Intelligent Computing and Cybernetics. 2008. V. 1, N. 1. P. 143–163.

26. **Fomichov V. A.** Theory of K-representations as a Comprehensive Formal Framework for Developing a Multilingual Semantic Web // Informatica. An Intern. Journal of Computing and Informatics (Slovenia). 2010. V. 34, N. 3. P. 387–396.

27. **Fomichov V. A.** SK-languages as a Comprehensive Formal Environment for Developing a Multilingual Semantic Web // Decker H., Lhotská L., Link S., Spies M., Wagner R. R. (Eds.). Database and Expert Systems Applications, 25th Intern. Conference, DEXA 2014, Munich, Germany, September 1–4, 2014, Part I, Proceedings. Lecture Notes in Computer Science. Cham: Springer International Publishing Switzerland. 2014. V. 8644. P. 394–401.

28. **Правиков А. А., Фомичев В. А.** Разработка рекомендательной системы с естественно-языковым интерфейсом на ос-

нове математических моделей семантических объектов // Бизнес-информатика. Междисциплинарный научно-практический журнал ГУ-ВШЭ. 2010. № 4(14). С. 3–11.

29. **Захлебни И. В., Фомичев В. А.** Разработка метода семантического поиска специалистов в корпоративной базе данных по естественно-языковым запросам // Информационные технологии. 2015. № 5. С. 323–331.

30. **Fomichov V. A., Razorenov A. A.** A New Method of Extracting Structured Meanings from Natural Language Texts and Its Application // E. Metais, M. Roche, M. Teisseire (Eds.). Natural Language Processing and Information Systems. 19th International Conference on Applications of Natural Language to Information Systems, NLDB 2014, Montpellier, France, June 18–20, 2014, Proceedings. Cham: Springer International Publishing Switzerland. 2014. P. 81–84.

31. **Razorenov A. A., Fomichov V. A.** The Design of a Natural Language Interface for File System Operations on the Basis of a Structured Meanings Model // Procedia Computer Science, Elsevier. V. 31. P. 1005–1011; open access, URL: <http://authors.elsevier.com/sd/article/S1877050914005304>.

32. **Chong S., Pucella R.** A Framework for Creating Natural Language User Interfaces for Action-Based Applications // Proceedings of the Third International AMAST Workshop on Algebraic Methods in Language Processing. TWLT Report 21. 2003. P. 83–98.

33. **Домашняя** страница Siri: <http://www.apple.com/ios/siri/> — Электронный ресурс (проверено 17.12.2014).

34. **Домашняя** страница Sobesednik HD: <http://sobesednik.idevelops.com/> — Электронный ресурс (проверено 17.12.2014).

35. **Домашняя** страница Ubiquity Home Page: <https://wiki.mozilla.org/Labs/Ubiquity> — Электронный ресурс (проверено 17.12.2014).

36. **Домашняя** страница Braina Project Home Page: <http://www.brainasoft.com/braina/#overview> — Электронный ресурс (проверено 17.12.2014).

37. **Домашняя** страница NLUI Server Home Page: http://www.linguasys.net/web_production/server-item/NLUI%20Server — Электронный ресурс (проверено 17.12.2014).

38. **Marlow S.** Parallel and Concurrent Programming in Haskell. O'Reilly. 2013.

39. **O'Sullivan B., Stewart D., Goerzen J. J.** Real World Haskell. O'Reilly. 2008.

40. **Peyton Jones S.** Haskell 98 Language and Libraries: the Revised Report. Cambridge University Press. 2003.

V. A. Fomichov, Professor, e-mail: vfomichov@hse.ru,
A. A. Razorenov, Ph. D. student, e-mail: razorenov@mail.ru

National Research University Higher School of Economics (HSE), Moscow, Russia

The Significance of the K-Representations Theory for the Studies on Automatic Semantic Role Labeling

The paper shows an incompleteness of theoretical foundations of the Computational Semantics branches called Semantic Role Labeling and Frame-Semantic Parsing. This situation is a consequence of a seeming lack of a semantic formalism allowing to describe semantic structures of complex sentences and discourses pertaining to arbitrary application domains. It is concluded that the theory of K-representations (knowledge representations) provides a formalism of the kind, determining a new class of formal languages — the class of SK-languages (standard knowledge languages). Some new expressive mechanisms of SK-languages are illustrated. The central ideas of a method of semantic parsing of natural language (NL) texts proposed by the theory of K-representations are set forth. The method employs the class of SK-languages for constructing semantic representations of texts. The final part of the paper considers the application of the method to designing NL-interfaces for software management. A file manager with a NL-interface NLC-1 (Natural Language Commander — Version One) has been developed, the system is implemented with the help of the functional programming language Haskell.

Keywords: natural language, semantic-syntactic analysis of texts, semantic parsing, semantic role labeling, frame-semantic parsing, semantic representation, theory of K-representations, SK-language, linguistic database, software management, file system natural language management, Natural Language Commander, Haskell

References

1. Wang C., Xiong M., Zhou Q., Yu Y. PANTO: a portable NL-interface to ontologies. *4th European Semantic Web Conference Proceedings*. Springer, 2007. P. 473–487.
2. Cimiano P., Haase P. et al. Towards Portable Natural Language Interfaces to Knowledge Bases — the Case of the ORAKEL System. *Data and Knowledge Engineering*. 2008. V. 65, N. 2. P. 325–354.
3. Aggarwal N., Polajnar T., Buitelaar P. Cross-Lingual Natural Language Querying over the Web of Data. Metais E., Mezziane F., Saraee M., Sugumaran V., Vadera S. (eds.). *Natural Language Processing and Information Systems. 18th Intern. Conference on Applications of Natural Language to Information Systems, NLDB 2013*, Salford, UK, June 2013, Proceedings. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer. 2013. V. 7934. P. 152–163.
4. Ferre S. SQUALL: A Controlled Natural Language as Expressive as SPARQL 1.1. Metais E., Mezziane F., Saraee M., Sugumaran V., Vadera S. (eds.). *Natural Language Processing and Information Systems. NLDB 2013*. Salford, UK, June 2013. Proceedings Lecture Notes in Computer Science. Berlin, Heidelberg: Springer. 2013. V. 7934. P. 114–125.
5. Buitelaar P., Choi K.-S., Cimiano P., Hovy E. H. (Eds.) *Report from Dagstuhl Seminar 12362 "The Multilingual Semantic Web" (2–9 September, 2012)*. Schloss Dagstuhl: Leibniz-Zentrum fuer Informatik. 2012. 94 p.
6. Clark P., Harrison P. Boeing's NLP System and the Challenges of Semantic Representation. *Proc. SIGSEM Symposium on Text Processing (STEP'08)*. Venice, Italy, 2008. ACL. 2008. P. 263–276.
7. Marquez L., Carreras X., Litkowski K. C., Stevenson S. Semantic Role Labeling: an Introduction to the Special Issue. *Computational Linguistics*. 2008. V. 34, N. 2. P. 145–159.
8. Wintner S. What Science Underlies Natural Language Engineering? *Computational Linguistics*. 2009. V. 35, N. 4. P. 641–644.
9. Fomichov V. A. Semantics-Oriented Natural Language Processing: Mathematical Models and Algorithms. New York, Dordrecht, Heidelberg, London: Springer, 2010. 354 p.
10. Fomichov V. A. Formalizatsiya proektirovaniya lingvisticheskikh protsessorov. Moscow: MAKS Press, 2005. 368 p.
11. Fomichov V. A. *Matematicheskie osnovy predstavleniya sodержaniya poslanii kompyuternykh intellektualnykh agentov*. Moscow: State University — Higher School of Economics, Publishing House "TEIS". 2007. 176 p.
12. Das D., Chen D., Martins A. F. T., Schneider N., Smith N. A. Frame-Semantic Parsing. *Computational Linguistics*. 2014. V. 40, N. 1. P. 9–56.
13. Gildea D., Jurafsky D. Automatic Labeling of Semantic Roles. *Computational Linguistics*. 2002. V. 28, N. 3. P. 245–288.
14. Kingsbury P., Palmer M. 2002. From TreeBank to PropBank. *Proc. LREC*. 2002.
15. Marcus M. P., Marcinkiewicz M. A., Santorini B. Building a large annotated corpus of English: the Penn TreeBank. *Computational Linguistics*. 1993. V. 19, N. 2.
16. Punyakanok V., Roth D., Yih W. T. The importance of syntactic parsing and inferring in semantic role labeling. *Computational Linguistics*. 2008. V. 34, N. 2. P. 257–287.
17. Blanco E., Moldovan D. Leveraging verb-argument structures to infer semantic relations. *Proc. of the 14th Conference of the European Chapter of the Association for Computational Linguistics, Gothenburg, Sweden, April 26–30, 2014*. ACL, 2014. P. 145–154.
18. Yi S., Loper E., Palmer M. Can semantic roles generalize across genres? *Proc. of the Human Language Technologies Conference of the North American Chapter of the ACL*. Rochester, NY, 2007. P. 548–555.
19. Fillmore C., Johnson C. R., Petruck M. R. L. Background to FrameNet. *International Journal of Lexicography*. 2003. V. 16, N. 3. P. 235–250.
20. Fomichov V. A., Kirillov A. V. A Formal Model for Constructing Semantic Expansions of the Search Requests about the Achievements and Failures. *Artificial Intelligence: Methodology, Systems, and Applications*. Ramsay A., Agre G. (Eds.). Lecture Notes in Computer Science. V. 7557. Berlin, Heidelberg: Springer, 2012. P. 296–304.
21. Fomichov V. A. A Mathematical Model for Describing Structured Items of Conceptual Level. *Informatica. An International Journal of Computing and Informatics (Slovenia)*. 1996. V. 20, N. 1. P. 5–32.
22. Fomichov V. A. Matematicheskie osnovy predstavleniya smysla tekstov dlya razrabotki lingvisticheskikh informatsionnykh tekhnologii. Part I. *Informatsionnye Tekhnologii*. 2002. N. 10. P. 16–25 (in Russian).
23. Fomichov V. A. Matematicheskie osnovy predstavleniya smysla tekstov dlya razrabotki lingvisticheskikh informatsionnykh tekhnologii. Part II. *Informatsionnye Tekhnologii*. 2002. N. 11. P. 34–45 (in Russian).
24. Fomichov V. A. Novyi metod preobrazovaniya yestestvenno-yazykovykh tekstov v semanticheskie predstavleniya. *Informatsionnye Tekhnologii*. 2005. N. 10. P. 25–35 (in Russian).
25. Fomichov V. A. A Comprehensive Mathematical Framework for Bridging a Gap between Two Approaches to Creating a Meaning-Understanding Web. *Intern. Journal of Intelligent Computing and Cybernetics*. 2008. V. 1, N. 1. P. 143–163.
26. Fomichov V. A. Theory of K-representations as a Comprehensive Formal Framework for Developing a Multilingual Semantic Web. *Informatica. An Intern. Journal of Computing and Informatics (Slovenia)*. 2010. V. 34, N. 3. P. 387–396.
27. Fomichov V. A. SK-languages as a Comprehensive Formal Environment for Developing a Multilingual Semantic Web. Decker H., Lhotská L., Link S., Spies M., Wagner R. R. (Eds.). *Database and Expert Systems Applications, 25th Intern. Conference, DEXA 2014, Munich, Germany, September 1–4, 2014. Part I*. Proceedings. Lecture Notes in Computer Science. Cham: Springer International Publishing Switzerland. 2014. V. 8644. P. 394–401.
28. Pravikov A. A., Fomichov V. A. Razrabotka rekomendatsionnoy sistemy s estestvenno-yazykovym interfeisom na osnovе matematicheskikh modeli semanticheskikh obyektoy. *Business Informatics. Interdisciplinary scientific-practical journal of State University — Higher School of Economics, Moscow*. 2010. N. 4(14). P. 3–11 (in Russian).
29. Zakhlebina I. V., Fomichov V. A. Razrabotka metoda semanticheskogo poiska spetsialistov v korporativnoy baze dannykh po estestvenno-yazykovym zaprosam. *Informatsionnye Tekhnologii*. 2015. N. 6. P. 323–331 (in Russian).
30. Fomichov V. A., Razorenov A. A. A New Method of Extracting Structured Meanings from Natural Language Texts and Its Application. E. Metais, M. Roche, M. Teisseire (Eds.). *Natural Language Processing and Information Systems. 19th International Conference on Applications of Natural Language to Information Systems, NLDB 2014, Montpellier, France, June 18–20, 2014*. Proceedings. Cham: Springer International Publishing Switzerland. 2014. P. 81–84.
31. Razorenov A. A., Fomichov V. A. The Design of a Natural Language Interface for File System Operations on the Basis of a Structured Meanings Model. *Procedia Computer Science, Elsevier*. V. 31. P. 1005–1011; open access, URL: <http://authors.elsevier.com/sd/article/S1877050914005304>.
32. Chong S., Pucella R. A Framework for Creating Natural Language User Interfaces for Action-Based Applications. *Proc. of the Third International AMAST Workshop on Algebraic Methods in Language Processing. TWLT Report 21*. 2003. P. 83–98.
33. Home Page Siri: <http://www.apple.com/ios/siri/> — Electronic resource (retrieved 17.12.2014).
34. Home Page Sobesednik HD: <http://sobesednik.idevelops.com/> — Electronic resource (retrieved 17.12.2014).
35. Ubiquity Home Page: <https://wiki.mozilla.org/Labs/Ubiquity> — Electronic resource (retrieved 17.12.2014).
36. Braina Project Home Page: <http://www.brainasoft.com/braina/#overview> — Electronic resource (retrieved 17.12.2014).
37. NLUI Server Home Page: http://www.linguasys.net/web_production/server-item/NLUI%20Server — Electronic resource (retrieved 17.12.2014).
38. Marlow S. *Parallel and Concurrent Programming in Haskell*. O'Reilly. 2013.
39. O'Sullivan B., Stewart D., Goerzen J. J. *Real World Haskell*. O'Reilly. 2008.
40. Peyton Jones S. *Haskell 98 Language and Libraries: the Revised Report*. Cambridge University Press. 2003.

П. А. Борисовский, канд. физ.-мат. наук, доц., e-mail: borisovski@mail.ru
Омский государственный университет им. Ф. М. Достоевского, г. Омск,
А. В. Еремеев, д-р физ.-мат. наук, ст. науч. сотр., e-mail: eremeev@ofim.oscsnras.ru
Омский филиал Института математики им. С. Л. Соболева СО РАН, г. Омск

Построение расписаний многопродуктового производства с использованием целочисленного линейного программирования и эволюционных вычислений

Предложен новый алгоритм построения расписаний многопродуктового производства для задач большой размерности. Алгоритм основан на декомпозиционной схеме, дополненной эвристическим методом скользящего окна и генетическим алгоритмом. Вычислительный эксперимент показал преимущество предложенного алгоритма в сравнении с известным аналогом по качеству получаемых решений и времени вычислений и его применимости к реальным задачам в химической промышленности.

Ключевые слова: производственное расписание, декомпозиция, частично целочисленное линейное программирование, жадный алгоритм, генетический алгоритм, точка событий, реакция, заказ

Введение

При составлении расписания современного многопродуктового производства необходимо рассматривать большое число единиц оборудования, операций, промежуточных и конечных продуктов с учетом сложных взаимосвязей между ними. Задача состоит в назначении операций на единицы оборудования и выборе времени изготовления и объемов обрабатываемой продукции в каждой из этих операций с целью по возможности полно выполнить имеющиеся заказы на продукцию в период планирования.

Известно большое число работ по моделированию такого рода задач в терминах частично целочисленного линейного программирования (ЧЦЛП) [1–5]. Однако решение этих задач средствами ЧЦЛП связано со значительными трудностями при большом числе переменных и ограничений. В целях преодоления этих трудностей в литературе предложено несколько вариантов декомпозиционных схем [3–5], в которых период планирования разбивается на последовательность более коротких периодов (горизонтов планирования) и на каждом из этих периодов решается подзадача построения расписания, называемая далее *задачей нижнего уровня*.

В настоящей работе также используется декомпозиция и модель ЧЦЛП для задачи нижнего уровня, аналогичная модели из работы [5]. Для решения подзадач нижнего уровня предлагается новый эвристический метод *скользящего окна*, в котором используются идеи, аналогичные методу релаксации и фиксации [6]. В целях улучшения качества получаемых решений и сокращения времени вычислений метод скользящего окна можно комбинировать с генетическим алгоритмом, решающим вспомога-

тельную задачу построения расписания для критических (наиболее загруженных) единиц оборудования.

1. Постановка задачи

Входные данные задачи построения расписания многопродуктового производства [3, 5] следующие:

- 1) множество S видов сырья, промежуточных и конечных продуктов;
- 2) множество операций I и временных окон, когда они могут выполняться;
- 3) множество U единиц оборудования (устройств) с указанием момента, когда устройство становится доступным для использования и той операции, которая выполнялась на устройстве последней до этого момента;
- 4) таблица применимости, в которой для каждой операции указана единица оборудования, на которой данная операция может выполняться (предположение о том, что каждой операции соответствует одно устройство, не сужает общности постановки задачи, так как однотипные операции на различных устройствах можно рассматривать как разные операции);
- 5) минимальный и максимальный объемы партий продукта на выходе каждой операции;
- 6) для каждой операции множество потребляемых промежуточных продуктов и сырья, а также производимый продукт и коэффициенты потребления входных продуктов и сырья на единицу производимого продукта;
- 7) производительность каждой операции, т. е. количество продукта, получаемое в результате выполнения операции в единицу времени;
- 8) объемы заказов и директивные сроки, т. е. даты, не позднее которых заказана отгрузка конечного

продукта, причем на один продукт может быть несколько заказов с разными директивными сроками;

9) информация о рабочих и выходных днях, начиная с первого дня периода планирования.

Задача состоит в отыскании допустимого расписания, минимизирующего целевую функцию, которая представляет собой свертку из трех критериев. С наибольшими весами в свертку входят объем недопроизводства (т. е. разница между объемом заявок и объемом выпуска) и суммарное отклонение от директивных сроков. С существенно меньшим весом в свертку входит суммарное время переналадок оборудования.

В работе рассматривается производство, в котором операции по переработке сырья выполняются на сравнительно небольшом числе устройств, называемых далее *реакторами*. Кроме того, имеются вспомогательные единицы оборудования, служащие для загрузки сырья (входные загрузчики), промежуточного хранения (емкости для сырья и емкости для продукции) и отгрузки конечного продукта (погрузчики и упаковщики). Часть операций отгрузки могут выполняться только в рабочие дни. Множество операций с такими временными окнами далее обозначается через I^w .

Каждая технология включает в себя пять операций: загрузка сырья, хранение сырья, переработка сырья на реакторе (реакция), хранение продукта и его отгрузка, совмещенная с упаковкой, если требуется упаковка. Различным способом упаковки одного и того же промежуточного продукта соответствуют разные конечные продукты.

Ввиду того, что в рассматриваемом производстве "узким местом", как правило, оказываются реакторы, особое значение имеет множество промежуточных продуктов S^p , получаемых в результате реакций.

Множество конечных продуктов S^{art} разбивается на два подмножества: продукты, выпускаемые по заказу (подмножество S^{MTO}) и выпускаемые для складирования (подмножество S^{MTS}). Весовой коэффициент при объеме недопроизводства продуктов из S^{MTO} имеет большее значение, чем для продуктов из множества S^{MTS} . Директивные сроки для продуктов из множества S^{MTS} отсутствуют, и время их производства не учитывается в целевой функции.

При переходе от одной реакции к другой требуется выполнение операции переналадки реактора. Длительность операции i зависит от предшествующей и последующей реакций. Предполагается, что длительности переналадок удовлетворяют неравенству треугольника (это условие неявно предполагается и в работе [5]). Заметим, что предложенные ниже алгоритмы и модели ЧЦЛП применимы и в случае, когда неравенство треугольника нарушается, однако оптимальные решения в подзадачах нижнего уровня при этом могут оказаться недостижимы. Для операций переналадки также заданы временные окна.

2. Декомпозиционный метод и задача нижнего уровня

В настоящей работе используется декомпозиционный метод, аналогичный предложенному в работе [5]. Период планирования разбивается на горизонты планирования, и исходная задача составления расписания решается поэтапно, как последовательность задач нижнего уровня с использованием двух моделей ЧЦЛП. В модели горизонта планирования для рабочих дней допускается использование операций отгрузки из множества I^w , а в модели горизонта планирования на выходные дни такие операции не предусматриваются. В отличие от работы [5], где для выделения горизонтов планирования решается оптимизационная задача *верхнего уровня*, в настоящей работе разбиение на горизонты планирования осуществляется непосредственно на основе исходных данных о директивных сроках заказов.

Если первый день периода планирования является рабочим, то первый горизонт состоит из рабочих дней (его длительность не превышает 5 дней). Если же первый день периода планирования выходной, то первый горизонт состоит из выходных. Далее горизонты планирования, состоящие из выходных и рабочих дней, чередуются, пока не будет исчерпан весь период планирования. В модели нижнего уровня для первого горизонта планирования учитываются только те заказы, директивные сроки которых приходятся на дни этого горизонта планирования. После решения задачи нижнего уровня для текущего горизонта планирования происходит переход к следующему горизонту, причем в рассмотрении включаются как заказы с директивными сроками в дни нового горизонта планирования, так и не выполненные в полном объеме заказы из горизонтов, рассмотренных ранее. Если некоторые устройства в новом горизонте планирования оказались заняты операциями, назначенными на предыдущих этапах декомпозиции, то информация об этих операциях передается как часть исходных в новую задачу нижнего уровня.

После решения задачи нижнего уровня для последнего горизонта планирования происходит агрегация результатов всех решенных подзадач в единое решение исходной задачи, и декомпозиционный алгоритм заканчивает работу.

Модель ЧЦЛП для задачи нижнего уровня основана на принципе *непрерывного времени* [4]. Ограничения и критерий оптимизации — те же, что в исходной задаче, но вместо всего множества заявок учитывается его подмножество, назначенное на текущий горизонт планирования.

Выбор множества выполняемых операций и порядка их выполнения осуществляется с помощью бинарных переменных w_{in} , $i \in I$, $n \in N$, где $N = \{1, 2, \dots, N^{\text{max}}\}$ — множество *точек событий*. Каждая точка событий соответствует некоторой пози-

ции в последовательности выполнения операций. Переменная wv_{in} равна 1 тогда и только тогда, когда в точке событий $n \in N$ операция $i \in I$ назначена на соответствующее этой операции устройство (последнее определено однозначно). В таком случае будем говорить, что операция i активна в точке событий n . На каждом устройстве в одной точке событий может выполняться не более одной операции. Общее число точек событий N^{\max} выбирается априори на основе вычислительного эксперимента или теоретической оценки (см., например, [1]) и должно быть достаточно велико.

При $wv_{in} = 1$ время начала и окончания операции i в точке событий n представлено вещественными переменными TS_{in} и TF_{in} , а вещественная переменная B_{in} определяет объем обрабатываемого вещества. В случае $wv_{in} = 0$, переменные TS_{in} , TF_{in} и B_{in} могут принимать произвольные значения.

Ограничения и целевая функция рассматриваемой модели ЧЦЛП для задачи нижнего уровня совпадают с ограничениями и целевой функцией задачи нижнего уровня из работы [5], за исключением ограничений, относящихся к операциям переналадки реакторов, которые в настоящей работе сформулированы в модифицированном виде, представленном в работе [7]. Модифицированная формулировка [7] имеет преимущества при решении задач с большим числом точек событий и операций. Ввиду большого объема полное формальное описание модели здесь не приводится (см. [5, 7]).

3. Метод скользящего окна и жадная эвристика для задачи нижнего уровня

Описанная в разд. 2 задача нижнего уровня разрешима за приемлемое время современными пакетами решения задач ЧЦЛП, такими как CPLEX или Express-MP, если каждый горизонт планирования содержит лишь сравнительно малое число заказов. В работе [5], например, задача нижнего уровня решается для горизонтов планирования длительностью 12 ч, каждый из которых содержит в среднем около 134 продуктов, включая промежуточные. В подр. 3.1 этого раздела предлагается эвристический метод скользящего окна для более продолжительных горизонтов планирования. Далее, в подр. 3.2, на основе этого метода разработан жадный алгоритм.

3.1. Метод скользящего окна

В методе скользящего окна задача нижнего уровня из разд. 2 формулируется только для одного конечного продукта s' и в рассмотрение принимаются только те промежуточные продукты, которые можно использовать для производства продукта s' . Число точек событий выбирают с учетом технологии выпуска выбранного конечного продукта.

Будем говорить, что переменная фиксируется с некоторым значением, если в модели ЧЦЛП эта переменная заменяется постоянной величиной с указанным значением. Под снятием фиксации с переменной будем подразумевать переход к модели, в которой данная переменная снова может принимать любые значения из своей области определения.

В начальном решении полагаем $wv_{in} = 0$ для всех $i \in I$, $n \in N$, т. е. ни одна операция не выполняется. Конечные продукты упорядочиваются некоторым образом и обрабатываются далее в соответствии с этой последовательностью. Для каждого конечного продукта s' выбирается некоторое множество операций $I(s') \subseteq I$ по производству промежуточных продуктов и конечного продукта s' .

Составление расписания для очередного продукта s' выполняется посредством решения серии задач ЧЦЛП из разд. 2, в которых фиксируются бинарные переменные во всех точках событий кроме $n_s, \dots, n_e = n_s + w$, где w — настраиваемый параметр, называемый *шириной окна*. Серия начинается с задачи, в которой $n_s = 1$. При решении каждой задачи ЧЦЛП все бинарные переменные, которые не относятся к операциям из множества $I(s')$ и точкам событий n_s, \dots, n_e , фиксируются со значениями, выбранными при решении предыдущей задачи ЧЦЛП. Если в полученном решении полностью выполнен заказ на продукт s' или $n_s \geq N^{\max} - w$, то переходим к рассмотрению следующего конечного продукта. В противном случае n_s увеличивается на w , и попытки выпуска продукта s' продолжают.

3.2. Жадная эвристика

Описанный выше метод скользящего окна лежит в основе следующего жадного алгоритма.

Алгоритм 1. Жадный алгоритм

1. Положить $wv_{in} := 0$ для всех $i \in I$, $n \in N$.
2. Положить $P := S^{\text{art}}$, т. е. P — множество всех конечных продуктов.
3. Пока P не пусто, выполнять:
 - 3.1. Выбрать конечный продукт $s \in P$ согласно жадному правилу (см. описание ниже).
 - 3.2. Положить $n_s := 1$, $n_e := w$.
 - 3.3. Пока объем производства продукта s меньше объема заказа на этот продукт и $n_e \leq |M|$, выполнять шаги 3.3.1—3.3.4:
 - 3.3.1. Снять фиксацию переменных wv_{in} в точках событий $n = n_s, \dots, n_e$ для операций из множества $I(s)$.
 - 3.3.2. Решить задачу ЧЦЛП нижнего уровня с учетом имеющихся фиксаций.
 - 3.3.3. Зафиксировать все бинарные переменные равными их значениям в решении задачи нижнего уровня.
 - 3.3.4. Положить $n_s := n_s + w$; $n_e := n_e + w$.
 - 3.4. Удалить s из P .

Жадное правило на шаге 3.1 формулируется с учетом директивных сроков и объемов заказов, причем продукты из множества $S^{\text{МТО}}$ имеют больший приоритет, чем продукты из $S^{\text{МТS}}$. С этой целью сначала формируется множество продуктов-кандидатов $P' \subseteq P$ по следующему алгоритму.

Алгоритм 2. Формирование множества продуктов-кандидатов

1. Если P содержит продукты из множества $S^{\text{МТО}}$ с директивными сроками в текущем горизонте планирования, то добавить все такие продукты в P' .

2. Если $P' = \emptyset$, то положить $P' := S^{\text{МТО}}$.

3. Если $P' = \emptyset$, а P содержит продукты из множества $S^{\text{МТS}}$ с директивными сроками в текущем горизонте планирования, то добавить все такие продукты в P' .

4. Если $P' = \emptyset$, то положить $P' := P$.

Жадное правило состоит в выборе продукта-кандидата $s \in P'$ с наибольшим объемом заказа.

В результате фиксации большинства переменных задача ЧЦЛП в алгоритме 1 оказывается легко разрешимой современными пакетами решения задач ЧЦЛП.

Жадный алгоритм аналогичен методу релаксации и фиксации [6]. В отличие от алгоритма 1, в методе релаксации и фиксации [6] дополнительно осуществляется релаксация условий целочисленности для некоторых бинарных переменных. Однако, как показал эксперимент, релаксация условия целочисленности переменных в алгоритме 1 не приводит к существенному сокращению трудоемкости решения задачи ЧЦЛП. В связи с этим релаксация бинарных переменных в методе скользящего окна не используется.

Предложенный здесь метод скользящего окна может быть применен к задачам теории расписаний других типов, например, для производства партиями [3].

4. Построение расписания для реакторов в одном горизонте планирования

4.1. Задача построения расписания для реакторов

Сформулируем упрощенный вариант исходной задачи, рассматривая только реакторные устройства. Данная вспомогательная задача состоит в построении предварительного расписания, которое может быть использовано в гибридном алгоритме для вычисления полного решения в заданном горизонте планирования. Для дальнейшего упрощения задачи предположим, что все требуемое сырье доступно в любой момент времени в полном объеме. Введем следующие обозначения:

I^P — множество реакций ($I^P \subseteq I$);

U^P — множество реакторов ($U^P \subseteq U$);

$D_s > 0$ — объем заказа на продукт s , $s \in S^P$, отнесенный на рассматриваемый горизонт планирования;

$s_i \in S^P$ — продукт на выходе реакции $i \in I^P$;

$u_i \in U^P$ — устройство, на котором выполняется реакция $i \in I^P$;

$r_i > 0$ — производительность реакции $i \in I^P$, т. е. количество продукта s_i , производимого в единицу времени;

a_{ij} — длительность переналадки реактора u_i с реакции i на реакцию j , если $u_i = u_j$;

a'_i — длительность начальной наладки для выполнения реакции i , если эта реакция выполняется первой на реакторе u_i ;

$0 < T_i^{\text{max}}$ — максимальная допустимая длительность реакции. В данной работе предполагается, что эта величина равна времени, которое требуется для выпуска продукции в полном объеме, т. е. $T_i^{\text{max}} = D_{s_i}/r_i$.

Задача состоит в отыскании множества реакций, количества выпуска продукции и момента начала для каждой из реакций при указанных выше ограничениях на допустимую длительность реакций и выполнении всех заказов в полном объеме.

Целевая функция представляет собой свертку трех критериев. С наибольшим весом входит критерий общего времени выполнения всех заказов C_{max} . С меньшими весами в целевую функцию входят критерии минимизации суммарного времени переналадок и суммарного времени использования реакторов. Веса при этих критериях обозначим через P_1 и P_2 .

4.2. Модель частично целочисленного линейного программирования для реакций в одном горизонте планирования

Обозначим через I_u множество реакций, которые могут выполняться на реакторе $u \in U^P$. Запишем модель с использованием точек событий. Пусть $K_u = \{1, 2, \dots, N_u^{\text{max}}\}$ — множество точек событий, в которых может быть использован реактор u . С учетом неравенства треугольника всегда существует оптимальное решение, где каждая реакция выполняется не более чем один раз. Это свойство оптимального решения будет использовано при записи модели ЧЦЛП. В частности, из указанного свойства вытекает, что для гарантированного получения оптимального решения величину N_u^{max} достаточно выбрать равной $|I_u|$.

Введем переменные задачи:

$x_{in} \in \{0, 1\}$ равна 1, если реакция i назначена на реактор u_i в точке событий $n \in K_u$, иначе $x_{in} = 0$;

$\delta_i > 0$ — длительность реакции i ;

α_{un} — длительность переналадки на реакторе u между точками событий n и $n + 1$;

α'_u — длительность начальной наладки для выполнения реакции i , если эта реакция выполняется первой на реакторе u ;

C_{max} — общее время завершения реакций.

Модель ЧЦЛП имеет следующий вид:

$$\min C_{\max} + P_1 \left(\sum_{i \in I^p} \delta_i \right) + P_2 \left(\sum_{u \in U^p} (\alpha'_u + \sum_{n \in K_u} \alpha_{un}) \right); \quad (1)$$

$$\sum_{i \in I_u} x_{in} \leq 1, \quad u \in U^p, \quad n \in K_u; \quad (2)$$

$$\sum_{n \in K_u} x_{in} \leq 1, \quad u \in U^p, \quad i \in I_u; \quad (3)$$

$$\sum_{i \in I_u} x_{i, n-1} \geq \sum_{i \in I_u} x_{in}, \quad u \in U^p, \quad n \in K_u, \quad n > 1; \quad (4)$$

$$\sum_{i: s_i = s} r_i \delta_i \geq D_s, \quad s \in S^p; \quad (5)$$

$$\alpha_{un} \geq \sum_{i \in I_u} a_{ij} x_{i, n-1} - M(1 - x_{jn}),$$

$$u \in U^p, \quad j \in I_u, \quad n \in K_u, \quad n > 1; \quad (6)$$

$$\alpha'_u \geq \sum_{i \in I_u} a'_i x_{i,1}, \quad u \in U^p; \quad (7)$$

$$\sum_{i \in I_u} \delta_i + \alpha'_u + \sum_{n \in K_u} \alpha_{un} \leq C_{\max}, \quad u \in U^p; \quad (8)$$

$$\delta_i \leq T_i^{\max} \sum_{n \in K} x_{in}, \quad i \in I^p; \quad (9)$$

$$x_{in} \in \{0, 1\}, \quad i \in I^p, \quad n \in K; \quad (10)$$

$$\delta_i, \alpha_{un}, \alpha'_u \geq 0, \quad u \in U^p, \quad i \in I_u, \quad s \in S^p, \quad n \in K. \quad (11)$$

Целевая функция (1) отражает основной и второстепенный критерии. Согласно ограничениям (2) и (3) в каждой точке событий имеется не более одной реакции и любая реакция присутствует не более, чем в одной точке событий. Условие (4) обеспечивает "непрерывность" использования точек событий, т. е., если на реакторе некоторая точка событий содержит реакцию, то предыдущая точка событий также занята (это свойство важно для моделирования переналадок). Неравенства (5) связывают количество произведенной продукции и объемы заказов. Время переналадки между точками событий $n - 1$ и n задается неравенством (6). Правая часть в (6) равна a_{ij} в том и только в том случае, когда $x_{i, n-1} = 1$ и $x_{jn} = 1$, где M — достаточно большая константа, например $M = \max_{i,j} a_{ij}$. Начальные длительности наладки реакторов задаются неравенством (7). Суммарное время реакций и переналадок для каждого реактора связано с переменной C_{\max} соотношением (8). Согласно неравенству (9), длительность реакции находится в интервале $[0, T_i^{\max}]$, если реакция i выполняется; иначе $\delta_i = 0$.

Сформулированная задача является NP -трудной в сильном смысле, так как к ней сводится NP -трудная в сильном смысле задача о кратчайшем полуметрическом гамильтоновом пути [8].

4.3. Генетический алгоритм

Рассмотрим адаптацию генетического алгоритма (ГА) для приближенного решения задачи (1)—(11). ГА относится к числу рандомизированных эвристик и основан на эволюционных принципах (см., например, [9]). Пробные решения задачи оптимизации представляются как *особи популяции*. Как правило, особи кодируются с помощью бинарных строк, называемых *генотипами*, однако в некоторых случаях нецелесообразно применять кодировку с использованием большего числа символов. Элементы строки генотипа принято называть *генами*. Алгоритм начинает работу с некоторой популяции $\Pi^{(0)}$, представляющей собой набор генотипов, построенных случайным образом с заданным априори распределением вероятностей. На каждой итерации строится некоторое число новых пробных решений посредством применения оператора *рекомбинации* Cross и оператора *мутации* Mut к родительским решениям, выбранным из текущей популяции $\Pi^{(t)}$. Выбор родительских решений осуществляет оператор *селекции*, который действует случайным образом с вероятностным распределением, зависящим от значений *функции приспособленности* генотипов популяции $\Pi^{(t)}$. Функция приспособленности, как правило, вычисляется по значению целевой функции в том пробном решении, которое кодирует генотип. Если генотип кодирует недопустимое решение, то функция приспособленности может вычисляться с учетом штрафа за нарушение ограниченный задачи. В рассматриваемом ГА используется *стационарная стратегия управления популяцией* [10].

ГА со стационарной стратегией управления популяцией

1. Инициализировать популяцию $\Pi^{(0)}$.
2. Для всех t от 1 до t_{\max} выполнять шаги 2.1—2.6:
 - 2.1 $\Pi^{(t)} := \Pi^{(t-1)}$.
 - 2.2 Выбрать родительские генотипы $\mathbf{p}^1, \mathbf{p}^2$ из $\Pi^{(t)}$ оператором селекции.
 - 2.3 Вычислить два потомка $\mathbf{c}^1, \mathbf{c}^2$ оператором рекомбинации Cross($\mathbf{p}^1, \mathbf{p}^2$).
 - 2.4 Применить мутацию $\mathbf{c}^1 := \text{Mut}(\mathbf{c}^1), \mathbf{c}^2 := \text{Mut}(\mathbf{c}^2)$.
 - 2.5 Выбрать два неперспективных в некотором смысле генотипа $\mathbf{q}^1, \mathbf{q}^2$ из $\Pi^{(t)}$.
 - 2.6 Заменить $\mathbf{q}^1, \mathbf{q}^2$ на $\mathbf{c}^1, \mathbf{c}^2$ в популяции $\Pi^{(t)}$.

Данная стратегия управления популяцией хорошо зарекомендовала себя на практике, однако несколько отличается от используемой в классическом ГА, где на каждой итерации происходит полная замена популяции [9].

Кодировка решений. В рассматриваемом ГА генотип состоит из двух частей: перестановки $\pi = (i_1, \dots, i_{|I^p|})$ и целочисленного вектора $\mathbf{L} = (L_1, \dots, L_{|U^p|})$. При декодировании расписания

В процедуре равномерного кроссинговера для каждого $u \in U^P$ с вероятностью 0,5 независимо от других генов полагаем $K_u^1 := L_u^1, K_u^2 := L_u^2$; в противном случае полагаем $K_u^1 := L_u^2, K_u^2 := L_u^1$.

Оператор мутации. В операторе мутации к перестановке π последовательно применяются следующие три процедуры. В процедуре *обмена* из множества $\{1, \dots, |I^P|\}$ равновероятно выбираются два номера генов j и j' , после чего выполняется обмен значений в генах π_j и $\pi_{j'}$. В процедуре *вставки* равновероятно выбирается один номер гена $j \in \{1, \dots, |I^P|\}$ и ген π_j переносится в новую случайно выбранную позицию j' . Прежние значения в генах между j и j' (включительно) сдвигаются на одну позицию. Процедура *удаления* состоит в выборе наиболее загруженного реактора и удалении с него одной равновероятно выбранной реакции.

Мутация вектора L выполняется следующим образом: каждый ген $L_u, u = 1, \dots, m$, такой что $0 < L_u < |I_u|$, увеличивается на 1 с вероятностью p_m , либо уменьшается на 1 с вероятностью p_m . В противном случае, т. е. с вероятностью $1 - 2p_m$, ген L_u остается без изменений. Если же $L_u = 0$, то этот ген может только увеличиваться, что происходит с вероятностью p_m . Если же $L_u = |I_u|$, то этот ген может только уменьшаться с вероятностью p_m .

Полученный в результате мутации генотип может давать недопустимое решение при декодировании — в таком случае к генотипу применяется процедура корректировки решений и модифицированный генотип добавляется в популяцию.

5. Гибридный алгоритм для задачи нижнего уровня

Предлагаемый гибридный алгоритм основан на эвристике скользящего окна, применяемой с учетом решения, найденного генетическим алгоритмом.

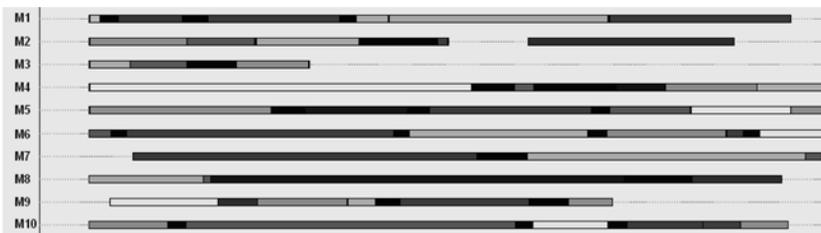


Рис. 2. Решение подзадачи для реакций, найденное генетическим алгоритмом

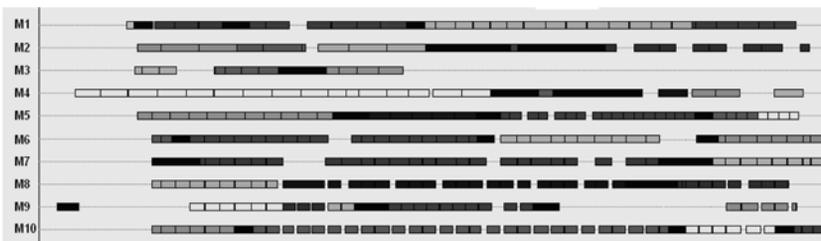


Рис. 3. Решение задачи нижнего уровня, найденное гибридным алгоритмом

В процессе работы декомпозиционного алгоритма в каждом горизонте планирования строится своя подзадача ЧЦЛП для реакций в соответствии с моделью из подразд. 2. Данная задача решается приближенно с помощью генетического алгоритма, далее применяется эвристика скользящего окна, вычисляющая более детальное решение в соответствии с моделью из §3. При работе эвристики скользящего окна последовательность выполнения заказов и назначение операций на реакторы соответствуют расписанию реакций, построенному генетическим алгоритмом. Гибридный алгоритм имеет следующий вид.

1. Построить с помощью генетического алгоритма (ГА) расписание для реакций на текущий горизонт планирования.

2. Упорядочить реакции по моменту начала их выполнения в решении, полученном ГА. Пусть $i_1, \dots, i_{k'}$ — полученная последовательность реакций, $s_1, \dots, s_{k'}$ — последовательность конечных продуктов, соответствующих этим реакциям и $d_1, \dots, d_{k'}$ — объемы производства конечных продуктов.

3. Сформулировать задачу нижнего уровня с множеством конечных продуктов $\{s_1, \dots, s_{k'}\}$ и объемами заказов $d_1, \dots, d_{k'}$. Директивные сроки продуктов соответствуют данным исходной задачи.

4. Методом скользящего окна построить решение задачи в текущем горизонте планирования, просматривая конечные продукты в последовательности $s_1, \dots, s_{k'}$ и полагая $I(s_j) = \{i_j, j = 1, \dots, k'\}$.

На рис. 2 и 3 приведены диаграммы Ганта для решений, полученных ГА и гибридным алгоритмом в случае 10 реакторов M1—M10.

Решение, полученное гибридным алгоритмом, включает в себя также расписание для загрузчиков, емкостей хранения и отгрузки, однако на рис. 3 изображены только операции реакторов. Как видно из рисунков, решение гибридного алгоритма отличается меньшей длительностью реакций и большим их числом. Это связано с ограниченностью емкостей для складирования продуктов реакций. Всякий раз при наполнении этих емкостей необходимо окончить реакцию и отгрузить продукт. Другое отличие состоит в том, что на рис. 2 имеется только один свободный промежуток между реакциями, в то время как на рис. 3 между реакциями имеются многочисленные промежутки. Это различие связано с тем, что в модели из разд. 2 учитываются директивные сроки, в то время как в модели из подразд. 4.2 основным критерием является минимизация общего времени завершения операций. Кроме того, в некоторые промежутки времени операции по

отгрузке могут быть запрещены, что также ведет к отсрочке некоторых реакций и/или сокращению их объема.

6. Вычислительный эксперимент

В настоящем разделе проводится сравнение предложенных алгоритмов с коммерческим пакетом решения задач ЧЦЛП CPLEX 12.3 и с декомпозиционным алгоритмом из работы [5]. В подразд. 6.1 генетический алгоритм из подразд. 4.3 сравнивается с пакетом CPLEX при решении задачи (1)–(11). Далее в подразд. 6.2 тестируется декомпозиционный метод (см. разд. 2) при использовании гибридного либо жадного алгоритмов решения задачи нижнего уровня. Эксперименты выполнялись на ЭВМ с процессором AMD Phenom 2,8 МГц при использовании одного ядра процессора. Исключение составляет процедура решения задач ЧЦЛП в пакете CPLEX, где допускалось использование двух ядер.

6.1. Сравнение генетического алгоритма с пакетом CPLEX

Генетический алгоритм был запрограммирован на языке C++ в системе MS Visual Studio 6.0. При решении задач линейного программирования (12)–(15) использовалась реализация двойственного симплекс-метода из пакета программ QSOPT. Кроме того, модель (1)–(11) была представлена в системе GAMS, интегрированной решателем задач ЧЦЛП из пакета CPLEX 12.3.

Тестовые примеры были построены случайным образом со следующими параметрами: $D_S \in [10; 20]$ для всех $s \in S^P$, $r_i \in [1; 20]$ для всех $i \in I^P$, $a_{ij} \in [0; 30]$ для всех $i \in I^P$ и $j \in I^P$, таких что $u_{i^*} = u_j$. Для каждой реакции $i \in I^P$ выбор устройства u_i , на котором она может выполняться, осуществлялся равновероятно на множестве U^P . Соответствие реакций продуктам устанавливалось таким образом, чтобы каждый продукт выпускался хотя бы одной из реакций: операции с порядковым номером не более $|S^P|$ назначались на выпуск продукта с тем же порядковым номером; для операций с большим номером продукт выбирался равновероятно из S^P . Мощность множеств S^P , U^P и I^P , а также число бинарных переменных в модели ЧЦЛП и верхняя граница на время счета алгоритмов приведены в табл. 1.

Настраиваемые параметры ГА выбраны следующим образом: численность популяции равна 400, размер турнира $k = 20$, вероятность мутации $p_m = 0,05$. Второстепенные критерии входят в целевую функцию задачи (1)–(11) с весами $P_1 = P_2 = 0,001$.

На каждом тестовом примере пакет CPLEX и ГА выполнялись в течение равных промежутков времени, выбранных как указано в табл. 1. Генетический алгоритм за отведенное время выполнялся 5 раз с независимой случайной инициализацией начальной популяции (т. е. время счета при каждом

Таблица 1

Параметры задач и ограничение на время счета

Задача	$ S^P \times U^P \times I^P $	Число бинарных переменных	Ограничение по времени, с
1	$30 \times 2 \times 80$	1885	300
2	$40 \times 2 \times 150$	5476	600
3	$50 \times 3 \times 180$	6604	900
4	$60 \times 3 \times 200$	8642	1200
5	$150 \times 10 \times 500$	21 950	3600

Таблица 2

Сравнение решений, полученных ГА и CPLEX

Задача	CPLEX			ГА		
	C_{\max}	Время переналадок	Время реакций	C_{\max}	Время переналадок	Время реакций
1	143	78	209	141	64	219
2	209	153	264	183	73	293
3	173	208	311	162	123	357
4	275	356	469	229	155	524
5	202	939	1051	183	492	1291

запуске ГА составляло 1/5 от указанного времени). Лучшее из решений, полученных за пять запусков, считалось окончательным ответом ГА. Результаты приведены в табл. 2, где C_{\max} — общее время завершения; время переналадок — суммарное время переналадок; время реакций — суммарное время выполнения операций.

Как видно из табл. 2, в целом результаты ГА превосходят результаты пакета CPLEX. Наиболее существенное преимущество наблюдается по общему времени переналадок. Несмотря на то что общее время реакций меньше в решениях, полученных пакетом CPLEX, легко видеть, что по сумме времени выполнения реакций и переналадок решения ГА имеют преимущество.

6.2. Исследование декомпозиционного алгоритма на задаче большой размерности

Далее рассматривается задача с реальными данными из работы [5], в которой имеется 10 реакторов, порядка 100 вспомогательных устройств и более 100 заказов на период планирования, равный одному месяцу. Как указано в разд. 5, период планирования разбивался на горизонты из рабочих дней и горизонты из выходных дней. Максимальное число точек событий $N^{\max} = 50$ для горизонтов из рабочих дней, а для горизонтов из выходных $N^{\max} = 20$.

Предложенные алгоритмы тестировались в двух вариантах: в первом случае задача нижнего уровня решалась жадной эвристикой, описанной в разд. 3, во втором случае для этого использовался гибридный алгоритм из разд. 5. В обоих вариантах в методе скользящего окна при каждом обращении

Таблица 3

Сравнение результатов для задачи большой размерности

Параметры решений	Алгоритм декомпозиции [5]	Декомпозиция с жадным алгоритмом	Декомпозиция с гибридным алгоритмом
Недопроизводство, %	0,96	0,02	0,07
Число переналадок	324	194	115
Время переналадок, ч	930	851	490
Время счета, ч	16	12	6

к CPLEX решателю отводилось 90 с. Время счета ГА ограничивалось значением 5 мин для горизонтов из выходных дней и 10 мин для рабочих дней. Полученные результаты приведены в табл. 3. Здесь указаны следующие параметры решений: недопроизводство, общее число переналадок, общая длительность переналадок и время работы алгоритма. Лучшие значения выделены жирным шрифтом.

Для сравнения в табл. 3 также приведены результаты, полученные алгоритмом декомпозиции из работы [5] при использовании горизонтов планирования длительностью 12 ч. Эти результаты несколько отличаются от опубликованных в работе [5] вследствие использования более новой версии CPLEX и более производительной ЭВМ в настоящей работе.

Все три алгоритма нашли решения, в которых директивные сроки для продуктов из множества S^{MTO} выполняются. Как видно из табл. 3, во всех полученных решениях недопроизводство незначительно. По числу переналадок и их длительности предложенные алгоритмы имеют преимущество по сравнению с алгоритмом из работы [5]. Время счета также сократилось.

Кроме того, были проведены дополнительные эксперименты, где в задаче построения расписания для реакторов вместо общего времени завершения S_{\max} использовался критерий минимизации недопроизводства на текущем горизонте планирования при дополнительном ограничении, обеспечивающем окончание всех операций в течение горизонта планирования. Было установлено, что данная модификация уступает модели, предложенной в подразд. 4.1, по числу переналадок и равномерности загрузки оборудования.

Заключение

В настоящей работе получил дальнейшее развитие предложенный К. Флудасом с соавторами подход к решению задач построения производственных расписаний с использованием декомпозиции по

времени и частично-целочисленного линейного программирования. В целях повышения качества решений и сокращения времени счета предложены гибридный и жадный алгоритмы решения задачи нижнего уровня. Гибридный алгоритм основан на комбинации разработанного авторами метода скользящего окна с генетическим алгоритмом.

На задаче с реальными данными большой размерности использование метода скользящего окна позволило увеличить длительность горизонта планирования, рассматриваемого на одном этапе декомпозиции, с 12 ч до нескольких суток. Сравнение результатов полученного алгоритма декомпозиции с результатами известного ранее декомпозиционного алгоритма показало преимущество нового алгоритма.

Метод скользящего окна имеет достаточно общий вид и может быть применен к другим задачам теории расписаний, записываемым в терминах частично целочисленного линейного программирования с использованием непрерывного времени и точек событий.

Работа поддержана РФФИ (проекты 12-01-00122 и 13-01-00862).

Список литературы

1. Еремеев А. В., Коваленко Ю. В. О задаче составления расписаний с группировкой машин по технологиям // Дискретный анализ и исследование операций. 2011. Т. 18, № 5. С. 54–79.
2. Cho D. A solution approach to the design of multi-period, multipurpose batch plants // Korean Journal of Chemical Engineering. 2002. V. 19, N. 2. P. 197–205.
3. Janak S. L., Floudas C. A., Kallrath J., Vormbrock N. Production scheduling of a large-scale industrial batch plant. I. Short-term and medium-term scheduling // Industrial & Engineering Chemistry Research. 2006. V. 45, N. 25. P. 8234–8252.
4. Ierapetritou M. G., Floudas C. A. Effective continuous-time formulation for short-term scheduling: I. Multi purpose batch processes // Industrial & Engineering Chemistry Research. 1998. V. 37. P. 4341–4359.
5. Shaik M. A., Floudas C. A., Kallrath J., Pitz H.-J. Production scheduling of a large-scale industrial continuous plant: Short-term and medium-term scheduling // Computers & Chemical Engineering. 2009. V. 33. P. 670–686.
6. Pochet Y., Wolsey L. A. Production Planning by Mixed Integer Programming. Series in Operations Research and Financial Engineering. Heidelberg: Springer, 2006. 500 p.
7. Borisovsky P., Eremeev A., Kallrath J. Reducing the number of changeover constraints in a MIP formulation of a continuous-time scheduling problem // To appear in Proc. of VIII International School-Symposium "Analysis, Modelling, Management and Development of Economical Systems" (AMUR'2014). — Sevastopol: DEN, 2014. URL: <http://arxiv.org/abs/1408.5832> (дата обращения 26.08.2014).
8. Itai A., Papadimitriou C. H., Szwarcfiter J. L. Hamilton paths in grid graphs // SIAM Journal on Computing. 1982. V. 11, N. 4. P. 676–686.
9. Goldberg D. E. Genetic Algorithms in Search, Optimization and Machine Learning. Reading: Addison Wesley, 1989. 412 p.
10. Reeves C. R. Genetic algorithms for the operations researcher // INFORMS Journal on Computing. 1997. V. 9, N. 3. P. 231–250.
11. Michalewicz Z. Genetic Algorithms + Data Structures = Evolution Programs. Berlin: Springer-Verlag, 1996. 387 p.

P. A. Borisovsky, Associated Professor, borisovski@mail.ru, Omsk State University Dostoyevsky,
A. V. Ereemeev, Senior Researcher, ereemeev@ofim.oscsbras.ru,
Omsk Branch of Sobolev Institute of Mathematics

Production Scheduling of a Multi-Product Plant Using Integer Linear Programming and Evolutionary Computations

In this paper, we propose a new algorithm for multi-product plant scheduling problem of high dimension. The algorithm is based on time-decomposition, moving window heuristic method and a genetic algorithm. An experimental study on a large-scale data showed a significant advantage of the solution quality and the solving time comparing to other algorithm, and confirmed the suitability of the proposed approach for the real-life production scheduling in chemical industry.

Keywords: production schedule, decomposition, mixed integer linear programming, greedy algorithm, genetic algorithm, event point, reaction, demand, task, planning horizon

References

1. Ereemeev A. V., Kovalenko Ju. V. O zadache sostavleniya raspisanij s gruppirovkoj mashin po tehnologijam. *Diskretnyj analiz i issledovanie operacij*. 2011. V. 18, N. 5. P. 54–79.
2. Cho D. A solution approach to the design of multi-period, multi purpose batch plants. *Korean Journal of Chemical Engineering*. 2002. V. 19, N. 2. P. 197–205.
3. Janak S. L., Floudas C. A., Kallrath J., Vormbrock N. Production scheduling of a large-scale industrial batch plant. I. Short-term and medium-term scheduling. *Industrial & Engineering Chemistry Research*. 2006. V. 45, N. 25. P. 8234–8252.
4. Ierapetritou M. G., Floudas C. A. Effective continuous-time formulation for short-term scheduling: I. Multi purpose batch processes. *Industrial & Engineering Chemistry Research*. 1998. V. 37. P. 4341–4359.
5. Shaik M. A., Floudas C. A., Kallrath J., Pitz H.-J. Production scheduling of a large-scale industrial continuous plant: Short-term and medium-term scheduling. *Computers & Chemical Engineering*. 2009. V. 33. P. 670–686.
6. Pochet Y., Wolsey L. A. Production Planning by Mixed Integer Programming. Series in Operations Research and Financial Engineering. Heidelberg: Springer, 2006. 500 p.
7. Borisovsky P., Ereemeev A. Simplification of changeover constraints in a MIP formulation of a continuous-time scheduling problem. 2014. URL: <http://arxiv.org/abs/1408.5832> (access date 26.08.2014).
8. Itai A., Papadimitriou C. H., Szwarcfiter J. L. Hamilton paths in grid graphs. *SIAM Journal on Computing*. 1982. V. 11, N. 4. P. 676–686.
9. Goldberg D. E. Genetic Algorithms in Search, Optimization and Machine Learning. Reading: Addison Wesley, 1989. 412 p.
10. Reeves C. R. Genetic algorithms for the operations researcher. *INFORMS Journal on Computing*. 1997. V. 9, N. 3. P. 231–250.
11. Michalewicz Z. Genetic Algorithms + Data Structures = Evolution Programs. Berlin: Springer-Verlag, 1996. 387 p.

УДК 004.855.5, 004.896

В. В. Воробьев, аспирант, e-mail: gatus86@mail.ru,

Е. А. Паршикова, аспирант, e-mail: inara@korolev-net.ru,

Московский институт электроники и математики Национального исследовательского университета "Высшая школа экономики", г. Москва

Применение мультимножеств для оценки ситуации мобильным агентом¹

Рассмотрена возможность применения мультимножеств для оценки ситуации мобильным агентом. Показано, что форма представления свойств объектов окружающей среды в виде мультимножеств позволяет агрегировать сенсорные данные агента, снижая тем самым размерность входного вектора данных.

Ключевые слова: мультимножества, агрегирование данных, мобильный агент, интеллектуальные роботы, распознавание ситуаций, теория управления, обучение с подкреплением, обучение на примерах, база прецедентов, конечный автомат

Введение

Целью работы является исследование возможности применения мультимножеств для оценки ситуации мобильным агентом и эффективности этого метода для решения данной задачи.

¹Работа выполнена при поддержке РФФИ (проект № 14-01-00817).

Для функционирования в динамической среде с неизвестными характеристиками агент должен обладать развитой рецепторикой. Однако увеличение числа сигналов рецепторов приводит к большим сложностям обучения и обработки этих сигналов системой управления. Введение параллелизма (как в случае использования нейронной сети) не спасает ситуацию, так как время обучения возрастает экс-

пониженно. Одним из путей решения данной проблемы является классификация множества входных сигналов (распознавание ситуаций). Вместо стимул-реактивного преобразования "вход—выход" $Y = R(X)$ требуется наличие дополнительного устройства — классификатора C . Классификатор C может быть различного вида — от множества продуктов до реализации в виде нейронной сети или хромосомы генетического алгоритма. Его функция заключается в анализе входного вектора и определении класса, к которому этот вектор относится [1].

Другим способом снижения размерности вектора входных сигналов является использование различного рода процедур агрегирования. Для представления входных сенсорных данных в целях их абстрагирования и агрегирования в работе предложено использовать мультимножества.

1. Мультимножества и обоснование их применения в поставленной задаче

Мультимножество, как и обычное множество, есть совокупность элементов произвольной природы. Однако, в отличие от множеств, один и тот же элемент может присутствовать в мультимножестве многократно, и кратность вхождения элемента является существенной особенностью мультимножеств [2].

Мультимножеством A , порожденным обычным множеством, все элементы которого различны (1.1), называется совокупность групп элементов вида (1.2):

$$U = \{x_1, x_2, \dots\}; \quad (1.1)$$

$$A = \{k_A(x) * x | x \in U, k_A(x) \in Z^+\}, \quad (1.2)$$

где $k_A: U \rightarrow Z^+ = \{0, 1, 2, \dots\}$ — функция числа экземпляров мультимножества, определяющая кратность вхождения элемента $x_i \in U$ в мультимножество A (обозначено символом "*"").

Если $k_A(x) = x_A(x)$, где $x_A(x) = 1$ при $x \in A$ и $x_A(x) = 0$ при $x \notin A$, то мультимножество A становится обычным множеством [3].

Мультимножество является удобной математической моделью для представления многопризнаковых объектов. Они успешно используются в различных предметных областях, где часто возникает необходимость сгруппировать или упорядочить анализируемые объекты, основываясь на их свойствах, выраженных признаками (атрибутами) объектов. Мультимножества допускают использование различных, в том числе и противоречивых, данных для описания объектов [3].

Среда, в которой существует мобильный агент, характеризуется большим числом свойств и их повторяемостью, следовательно, представляется целесообразным исследовать механизм мультимножеств для оценки ситуации мобильным агентом. Ситуация характеризуется набором наблюдаемых объектов, который предлагается представить в виде мультимножества.

2. Описание среды обитания и агента

Рассмотрим типичную задачу: существуют n агентов и некая общая среда обитания (рис. 1). Среда обитания агентов дискретна и представляет собой совокупность клеток. Общая среда обитания разделена на n непересекающихся областей (по числу агентов) и каждый из n агентов функционирует только в своей области.

На рис. 1 показан пример среды обитания всех агентов, квадратом — область действий одного агента, которая, в свою очередь, состоит из дискретных клеток, по которым агент движется. В данной среде существует множество объектов, либо представляющих опасность для агента (например, ловушка), либо являющихся "положительными" (например, кормушка), или "нейтральными". "Кормушки" обозначаются горизонтальными полосами, "ловушки" — вертикальными, "нейтральные объекты" — пустые клетки, агенты обозначены R.

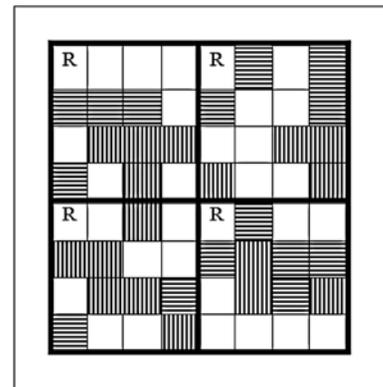


Рис. 1. Пример среды обитания агентов

Агент, двигаясь внутри своего сегмента, получает за наезд на "кормушки" положительные оценки, а за наезд на "ловушки" — отрицательные. Наблюдаемый агентом в произвольный момент времени набор "ловушек" и "кормушек" представляется в виде мультимножества, состоящего из нулей ("кормушки") и единиц ("ловушки").

Каждый агент должен сформировать свой "жизненный опыт" — набор ситуаций, в которых он побывал, проведенных в каждой ситуации действий и полученных за эти действия оценок. Агент при этом должен максимизировать положительные оценки и минимизировать отрицательные. Фактически речь идет о классической задаче моделирования поведения агента в детерминированной среде (см., например, [4]).

3. Исследование области агентом

Мобильный агент представляет собой обучаемый конечный автомат с одним состоянием, описанный в работе [5], действия которого определяются стохастическими векторами.

Рецепторика агента позволяет наблюдать объекты на расстоянии одной клетки вокруг себя. В случае

если агент наблюдает какой-либо объект или множество объектов, он выбирает одно из высокоуровневых действий:

- двигаться в сторону центра масс "кормушек";
- двигаться в сторону центра масс "ловушек";
- двигаться от центра масс "кормушек";
- двигаться от центра масс "ловушек".

Действие выбирается агентом исходя из вероятностного вектора действий. Если агент не наблюдает никаких объектов, то он случайным образом выбирает одно из восьми возможных направлений и двигается в соответствующую сторону. Двигаясь по области действия, агент попадает в ситуации, характеризующиеся набором наблюдаемых объектов ("кормушек", "ловушек" и "нейтральных"). Этот набор представляется в виде мультимножества (3.1):

$$M = \{k_1 * x_1, k_2 * x_2, \dots, k_n * x_n\}, \quad (3.1)$$

где k_1, k_2, \dots, k_n — число вхождений элементов x_1, x_2, \dots, x_n в мультимножество соответственно.

Фактически форма представления входных сенсорных данных с помощью мультимножеств приводит к уменьшению входного вектора данных (т. е. его агрегации) с m^n до $(mn)^{n'}$, $n' < n$, где m — число возможных входных сигналов датчика, n — число входных сигналов до агрегации, n' — число входных сигналов после агрегации.

Затем агент выбирает одно из возможных действий исходя из вероятностного вектора и осуществляет его:

$$\left. \begin{aligned} F(\varphi_{nu+v}) &= f_{u+1}, u = 0, \dots, x-1; v = 1, \dots, n; \\ a_{ij}(t+1, s(t)) &= a_{ij}(t, s(t)) + \\ &+ (-1)^{s(t+1)} g a_{ij}(t, s(t))(1 - a_{ij}(t, s(t))); \\ a_{ik}(t+1, s(t)) &= a_{ik}(t, s(t)) + \\ &+ (-1)^{s(t+1)} g a_{ik}(t, s(t)) a_{ij}(t, s(t)), j \neq k, \end{aligned} \right\} (3.2)$$

где a_{ij} — вероятностный вектор; $s(t)$ — поощрение/наказание [6].

Каждый агент, наблюдая произвольный набор "ловушек" и "кормушек", которые формируют мультимножество, реализует действия исходя из вероятностного вектора в данный момент времени. Так агент накапливает свой "жизненный опыт", получая за действия некоторые оценки и меняя, исходя из этих оценок, вероятностный вектор действия. Таким образом формируется база прецедентов каждого агента. В каждой базе прецедентов хранятся мультимножества, характеризующие ситуации, в которые попадал агент, и вероятностные векторы действий, сформированные с помощью оценок, полученных за выполненные агентом действия.

4. Использование общей базы прецедентов

После того как база прецедентов была сформирована, в системе появляется агент-новичок без "жизненного опыта".

Различные базы прецедентов могут иметь пересечения между собой, т. е. в каждой из них может встречаться одно и то же мультимножество, но вероятностные векторы действий в таком случае, скорее всего, будут различны.

Строка базы прецедентов представляет собой структуру, хранящую комбинацию (мультимножество) и вероятностный вектор действий, благодаря которому выбирается текущее действие агента:

Комбинация (мультимножество)	Вероятностный вектор действий
---------------------------------	----------------------------------

Агенту-новичку необходимо сравнить мультимножество, характеризующее ситуацию, в которой он оказался, с мультимножествами из баз прецедентов, сформированных другими агентами, используя для качественного сравнения одну из мер сходства, например, метрику Хэмминга μ_{ij}^H (4.1) или метрику Роджерса—Танимото μ_{ij}^{R-T} (4.2), и выбрать наиболее подходящий вероятностный вектор действия:

$$\mu_{ij}^H = \frac{n_{ik}}{N}, \quad (4.1)$$

где n_{ik} — число совпадающих признаков у образцов X_i и X_k :

$$\mu_{ij}^{R-T} = n_{ik}''(n_i' + n_k' - n_{ik}''), \quad (4.2)$$

где n_{ik}'' — число совпадающих единичных признаков у образцов X_i и X_k ; n_i' и n_k' — общее число единичных признаков у образцов X_i и X_k соответственно.

Далее в экспериментах будет использоваться метрика Хэмминга исходя из ее простоты.

Дальнейшие действия агента определяются исходя из того, были ли совпадения текущего мультимножества с мультимножествами из базы прецедентов или совпадений не обнаружено. Совпавшими мультимножествами считаются такие мультимножества, в которых метрика Хэмминга показывает меру сходства, равную или выше некоей пороговой меры. Данный механизм необходим для того, чтобы иметь возможность сравнивать весь "жизненный опыт" одного агента с другим при появлении ситуации, где "жизненный опыт" одного агента противоречит "жизненному опыту" другого. Возможны следующие варианты действий агента:

- совпадений не обнаружено — дальнейшие действия агента определяются его (агента) текущим вероятностным вектором для данной ситуации;
- выявлено одно совпадение (текущее мультимножество обнаружено в базе прецедентов одного из агентов) — дальнейшие действия агента определяются вероятностным вектором, характерным для найденной в базе прецедентов ситуации;

- выявлено несколько не противоречащих друг другу совпадений, т. е. текущее мультимножество было обнаружено в базах прецедентов нескольких агентов и экстремумы стохастических векторов найденных прецедентов совпадают — дальнейшие действия агента определяются вектором с наибольшей вероятностью конкретного действия;
- выявлено несколько противоречащих друг другу совпадений, т. е. текущее мультимножество обнаружено в базах знаний нескольких агентов, но экстремумы стохастических векторов найденных прецедентов не совпадают — сравнивается весь "жизненный опыт" нового агента с "жизненным опытом" тех агентов, с которыми были совпадения.

База прецедентов агента-новичка, которую он успел накопить, сравнивается с базами прецедентов, в которых были обнаружены совпадения с текущей ситуацией (с помощью меры сходства Хэмминга сравниваются все мультимножества, описывающие все ситуации, в которых побывали агенты). Новым агентом будут использованы действия того агента, с чьей базой прецедентов было выявлено больше совпадающих или похожих ситуаций.

Таким образом, новый агент, используя "жизненный опыт" других агентов, способен действовать не только в ситуациях, где эффективные действия однозначно определены благодаря опыту других агентов, но и в похожих ситуациях, а также в ситуациях, где опыт предыдущих агентов противоречит опыту других.

6. Результаты имитационного моделирования

Ниже будут приведены результаты имитационного моделирования для четырех показательных экспериментов.

Эксперимент 1. Четыре агента исследуют каждый свою область из общей среды обитания. На рис. 2 показана начальная расстановка агентов в среде обитания и итоговые показатели качества агентов (оценки действий, представляющие собой отношение

$\frac{N}{N+M}$, где N — число положительных оценок, а M — число отрицательных). На рисунках используются следующие обозначения: x — агент, 1 — "ловушка", 0 — "кормушка", пустая клетка — "нейтральная".

Каждый агент использует накапливаемую им при исследовании базу знаний, но в области жизненного пространства одного из агентов (обведена на рис. 2 квадратом) находятся только "ловушки", таким образом, "жизненный опыт" этого агента будет только отрицательным. Показатели качества этого агента будут нулевыми, но вероятностные векторы, несмотря на это, будут сформированы, опираясь исключительно на ошибки. Таким образом, данная база прецедентов может быть успешно использована другими агентами, например, агентом-новичком. Используя ее, он допустит существенно меньше ошибок, чем агент, который сформировал эту базу прецедентов.

На рис. 3 показана зависимость показателей качества действий агентов от шага итерации для эксперимента 1. Из графика видно, что показатели качества агента с отрицательным жизненным опытом (агент 1) нулевые.

Эксперимент 2. Четыре агента исследуют каждый свою область из общей среды обитания, и действует один агент-новичок. Каждый агент, аналогично эксперименту 1, пользуется накапливаемой им при исследовании базой знаний, а агент-новичок пользуется накопленной ранее общей базой знаний. На рис. 4 показана зависимость показателей

x	0	0	1	x	1		1
			0	1	1		1
		0	0			1	1
	0		1	1			
x			0	x	1	0	0
	0		0	1			
	0						0
0	1	0	0	1	0	0	

Quality Indexes of robot 0:
0.828979
Quality Indexes of robot 1:
0.884784
Quality Indexes of robot 2:
0.000000
Quality Indexes of robot 3:
0.987525

Рис. 2. Эксперимент 1: начальная расстановка и итоговые показатели качества агентов

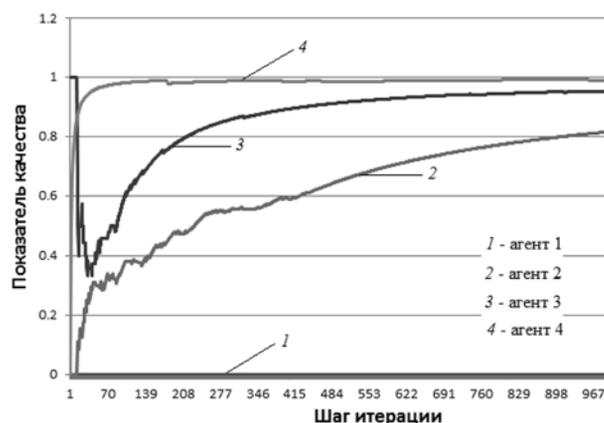


Рис. 3. Эксперимент 1: зависимость показателей качества действий агентов от шага итерации

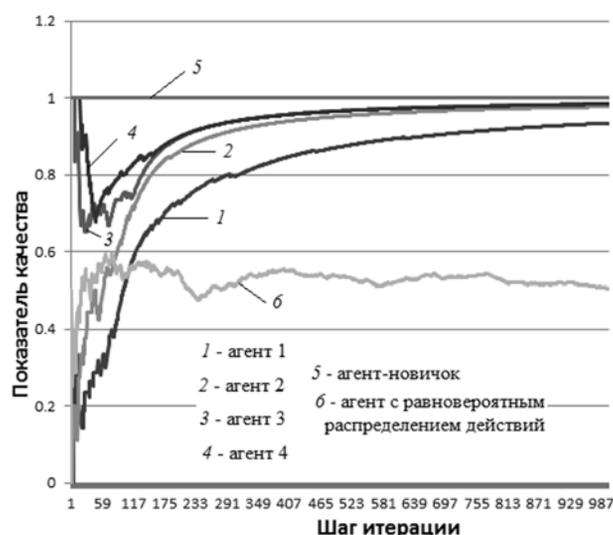


Рис. 4. Эксперимент 2: зависимость показателей качества действий агентов от шага итерации

качества действий агентов от шага итерации для эксперимента 2.

В результате данного эксперимента было установлено, что агент-новичок, используя базы прецедентов других агентов, за все время проведения эксперимента не совершил ни одной ошибки, и его показатель качества в конце эксперимента равен 1 (линия 5 на графике). Кроме того, графики других агентов асимптотически стремятся к 1, что говорит о том, что они совершают больше правильных действий с течением времени, т. е. обучаются. Линией 6 на графике представлен агент, который не обучается.

Рассчитанные значения дисперсий для показателей качества каждого из агентов:

$$D_{\text{агент1}} = 0,0362336;$$

$$D_{\text{агент2}} = 0,0288783;$$

$$D_{\text{агент3}} = 0,0070026;$$

$$D_{\text{агент4}} = 0,0040037;$$

$$D_{\text{агент-новичок}} = 0.$$

Эксперимент 3. Данный эксперимент интересен тем, что агент 1 помещен в среду, где присутствует множество опасностей и одна "кормушка" (011111) с высоким уровнем опасности. Если агент пытается убежать от опасностей, то он попадет в "ловушку" и получит наказание. В конечном итоге агент использует действие "ехать к кормушке", несмотря на обилие опасностей. То есть безопасней ехать к одной "кормушке" (безопасное место), нежели убежать в неизвестность от опасностей. На рис. 5 показана начальная расстановка агента 1 и его база прецедентов. Выделенная строка на рис. 5 показывает ячейку базы прецедентов с мультимножеством (0,1,1,1,1,1) и его стохастическим вектором, экстремум которого характеризует действие "ехать к кормушке", его вероятность равна 1.

На рис. 6 показана зависимость показателей качества действий агентов от шага итерации для эксперимента 3.

Эксперимент 4. В этом эксперименте показана ситуация, обратная предыдущей — множество "кормушек" и одна "ловушка". Однако стремление агента ехать к "кормушкам" приведет его к "ловушке". Таким образом, агент избегает совершения этого действия и выбирает действие "убежать от опасности", несмотря на низкий ее уровень. На рис. 7 показана начальная расстановка агента 1 и его база прецедентов. Выделенная строка на рис. 7 показывает ячейку базы прецедентов с мультимножеством {0,0,0,1} и его стохастическим вектором, экстремум которого характеризует действие "бежать от ловушки", его вероятность равна 1.

На рис. 8 показана зависимость показателей качества действий агентов от шага итерации для эксперимента 4.

Кроме приведенных выше показательных экспериментов было проведено имитационное моделирование для 1000 случайных расстановок с агентом-новичком. Была получена следующая статистика показателей качества агента-новичка за 1000 экспе-

-----	Robot №1
1 1	01 GoTo RightAction 1,00 0,00 0,00 0,00
-----	1111 GoTo WrongAction 0,55 0,45
X 1	011 GoTo RightAction 1,00 0,00 0,00 0,00
-----	0111111 GoTo RightAction 1,00 0,00 0,00 0,00
0 1	0111 GoTo RightAction 1,00 0,00 0,00 0,00
-----	1 GoTo WrongAction 0,15 0,85
	GoTo Neutral
	11 Escape Neutral 0,05 0,95
	111 GoTo Neutral 0,45 0,55

Рис. 5. Эксперимент 3: начальная расстановка агента 1 и его база прецедентов

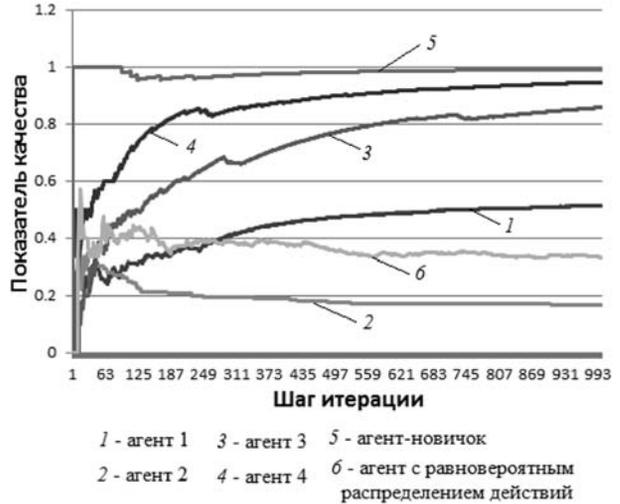


Рис. 6. Эксперимент 3: зависимость показателей качества действий агентов от шага итерации

-----	Robot №1
0	1 Escape Neutral 0,00 1,00
-----	0 GoTo RightAction 1,00 0,00
0 X 1	00 Escape RightAction 0,40 0,60
-----	001 Escape Neutral 0,25 0,58 0,00 0,17
0	GoTo Neutral
	0001 Escape RightAction 0,00 0,00 0,00 1,00

Рис. 7. Эксперимент 4: начальная расстановка агента 1 и его база прецедентов

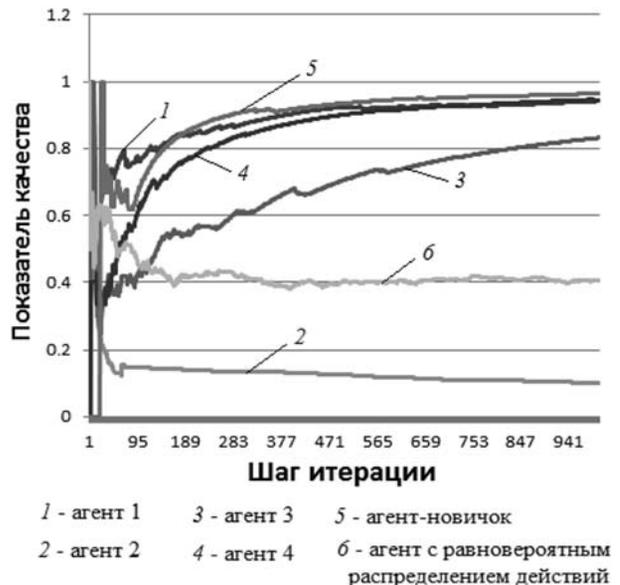


Рис. 8. Эксперимент 4: зависимость показателей качества действий агентов от шага итерации

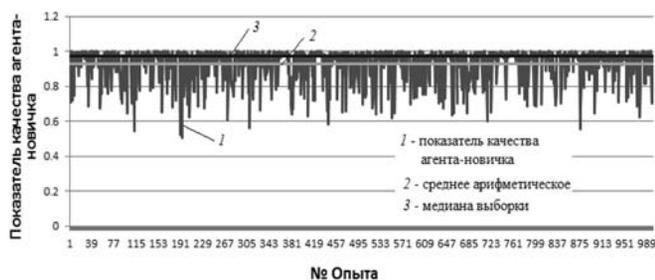


Рис. 9. Статистика показателей качества агента-новичка за 1000 экспериментов

риментов (рис. 9): среднее арифметическое данной выборки (линия 2 на графике) равно 0,92, а медиана выборки (линия 3 на графике) равна 0,97. Эти значения говорят о высоких показателях качества действий агента, использующего общую базу прецедентов и действующего в различных случайных начальных расстановках.

Заключение

В работе на примере обучения с подкреплением мобильного агента показано, что форма представления свойств объектов окружающей среды в виде мультимножеств является удобной для представления входных сенсорных данных и позволяет агрегировать сенсорные данные агента, снижая тем самым размерность входной задачи. Из экспериментов видно, что показатели качества всех обучающихся агентов выше, чем агента с равновероятностной

возможностью действия, что говорит, как минимум, о целесообразности их поведения [6].

Кроме того, данные, накопленные несколькими агентами в процессе своего обучения и хранящиеся в базе прецедентов в виде связки "мультимножество, характеризующее ситуацию — стохастический вектор действий", успешно используются агентом-новичком. Это подтверждает возможность разрешения коллизий, возникающих, когда в базе прецедентов находятся несколько экземпляров одного и того же мультимножества, но экстремумы стохастических векторов противоположны.

Список литературы

1. Добрынин Д. А., Карпов В. Э. Моделирование некоторых простейших форм поведения: от условных рефлексов к индуктивной адаптации // Сборник научных трудов I Международной конференции "Системный анализ и информационные технологии САИТ-2005" (12—16 сентября 2005 г., Переславль-Залесский, Россия). М.: КомКнига, 2005. Т. 1. С. 188—193.
2. Петровский А. Б. Пространства множеств и мультимножеств. М.: Едиториал УРСС, 2003. 248 с.
3. Петровский А. Б. Многокритериальное принятие решений по противоречивым данным: подход теории мультимножеств // Информационные технологии и вычислительные системы. 2004. № 2.
4. Варшавский В. И., Поспелов Д. А. Оркестр играет без дирижера. Размышления об эволюции некоторых технических систем и управления ими. М.: Наука. Главная редакция физико-математической литературы, 1984. 208 с.
5. Карпов В. Э. К вопросу об управлении мобильным роботом в условиях общей постановки задачи // Вестник компьютерных и информационных технологий. 2008. № 1. С. 2—9.
6. Цетлин М. И. Исследование по теории автоматов и моделированию биологических систем. М.: Наука. Главная редакция физико-математической литературы, 1969. 316 с.

V. V. Vorobyev, Postgraduate student, e-mail: gatus86@mail.ru,

E. A. Parshikova, Postgraduate student, e-mail: Parshikoff87@gmail.com,

Moscow Institute of Electronics and Mathematics National Research University "High School of Economics, Moscow

Application of Multisets for Assessment of the Situation by the Mobile Agent

The mobile agent needs the developed receptors for existence in a dynamic environment with unknown characteristics. But the increase in the number of receptors leads to a significant increase in the complexity of the processing of the signals coming from these sensors. The problem of the dimension of the input sensory data of agent occurs. The article discusses the possibility of using of multisets to reduce the dimension of the input data vector from sensors and the possibility of using of multisets for further assessment of the situation by the mobile agent. On the example of the mobile agent reinforcement learning it is shown that the presentation of the properties of the environment in the form of multisets allows to aggregate the sensory data of agent, thereby reducing the dimension of the input data vector.

Keywords: multisets, data aggregation, mobile agent, intelligent bots, recognition of situations, control theory, reinforcement learning, learning by example, the base of precedents, finite state machine

References

1. Dobrynin D. A., Karpov V. E. Modelirovanie nekotorykh prosteyshih form povedeniya: ot uslovykh refleksov k individualnoy adaptacii. *Sbornik nauchnykh trudov I Mezhdunarodnoy konferencii "Sistemnyy analiz i informacionnyye tehnologii SAIT-2005"* (12—16 September 2005, Pereslavl-Zaleskiy, Russia). M.: KomKniga, 2005. V. 1. P. 188—193.
2. Petrovskiy A. B. *Prostranstva mnozhestv i multimnozhestv*. M.: Editorial URSS, 2003. 248 p.
3. Petrovskiy A. B. *Mnogokriterialnoe prinyatie resheniy po protivorechivym dannym: podhod teorii multimnozhestv. Informacionnyye tehnologii i vychislitelnye sistemy*. 2004. N. 2.
4. Varshavskiy V. I., Pospelov D. A. *Orkestr igraet bez dirizhera. Razmyshleniya ob cvolucii nekotorykh tehnicheskikh system i upravleniya imi*. M.: Nauka. Glavnaya redakciya fiziko-matematicheskoy literatury, 1984. 208 p.
5. Karpov V. E. K voprosu ob upravlenii mobilnym robotom v usloviyah obschey postanovki zadachi. *Vestnik kompyuternykh i informacionnykh tehnologiy*. 2008. N. 1. P. 2—9.
6. Cetlin M. I. *Issledovaniye po teorii avtomatov i modelirovaniyu biologicheskikh system*. M.: Nauka. Glavnaya redakciya fiziko-matematicheskoy literatury, 1969. 316 p.

МОДЕЛИРОВАНИЕ И ОПТИМИЗАЦИЯ MODELING AND OPTIMIZATION

УДК 539.3

Г. Г. Булычев, д-р физ.-мат. наук, проф., e-mail: geo-bulychev@mail.ru,
Московский государственный технический университет
радиоэлектроники и автоматики (МГТУ МИРЭА)

Моделирование динамического разрушения полых тел продуктами детонации

Приведены модель, процесс и результаты численного моделирования динамики и динамического разрушения полых изотропных тел, подверженных действию внутренних импульсных нагрузок, имитирующих взрыв конденсированных взрывчатых веществ внутри этих тел. Моделирование проводили на языке ФОРТРАН FPS-11/2012 с использованием процессора FX8350 и программ авторской графики. Определены области разрушения и предельные нагрузки.

Ключевые слова: численное моделирование, механика деформированного твердого тела, динамика, разрушение, строительные конструкции

Введение

Дальнейшее развитие вычислительных средств, их программного обеспечения, а также алгоритмов метода пространственных характеристик для трехмерных задач динамики и динамического разрушения позволило проводить указанным методом моделирование динамического разрушения изотропных тел, содержащих внутри себя полости. Использование авторской машинной графики дало возможность локализовать области возможного разрушения и подобрать соответствующие критерии. Введение специальных кодовых таблиц в качестве выходных файлов позволило упростить определение характера разрушения в любой точке тела и анализировать движение трещин.

1. Этапы проекта и их краткое содержание

Общая схема проекта представлена на рис. 1. С ее помощью решаются две задачи моделирования.

1. Исследование процесса разрушения полого куба под действием взрывной нагрузки, создаваемой конденсированным взрывчатым веществом (ВВ), активированным в центральной точке внутренней

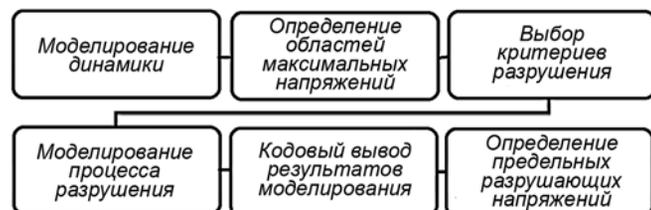


Рис. 1

полости. Взрыв предполагается точечным, действием отраженных волн пренебрегаем.

2. Исследование того же процесса, вызванного подрывом с одного конца тонкого цилиндрического шнура, состоящего из того же ВВ и уложенного по контуру одной из внутренних граней куба.

По результатам моделирования проводят анализ зависимостей напряжений, действующих на стенку куба в момент подхода к ней детонационной волны, от отношения скорости детонационной волны к скорости продольной волны в теле, толщины стенок тела и коэффициента Пуассона материала тела.

На первом этапе проводят моделирование динамики полого куба. Массивы данных, полученные в результате моделирования на этом этапе, используют для графической обработки в целях определения максимальных напряжений и областей их локализации.

На втором этапе проводят указанную обработку. При этом используют примитивы OpenGL и авторские подпрограммы построения контуров и изолиний, разработанные на базе пакета ГРАФОР [1].

На третьем этапе выбирают критерии начала разрушения и условия на трещинах. В качестве критериев начала разрушения используют условия накопления неупругих нормальных или касательных напряжений до заранее заданного значения. Дальнейшее поведение образовавшейся при этом трещины определяется с помощью одной из приведенных ниже табл. 1 и 2.

В этих таблицах знаки плюс и минус соответствуют различным берегам трещины, σ_n^\pm , $\sigma_{\tau 1}^\pm$, $\sigma_{\tau 2}^\pm$ — нормальные и касательные напряжения с обеих сто-

Таблица 1

Условия на трещине отрыва

Трещины нет	Трещина есть		
	сжатие ($\Delta u_n < 0$)		растяжение ($\Delta u_n \geq 0$)
	$\sigma_n < -\sigma^*$	$-\sigma^* \leq \sigma_n \leq 0$	
$\sigma_n^+ = \sigma_n^-$	$\sigma_n^+ = \sigma_n^-$	$\sigma_n^+ = \sigma_n^-$	$\sigma_n^+ = 0$
$\sigma_{\tau 1}^+ = \sigma_{\tau 1}^-$	$\sigma_{\tau 1}^+ = \tau^*$	$\sigma_{\tau 1}^+ = \beta^* \sigma_n^+$	$\sigma_n^- = 0$
$\sigma_{\tau 2}^+ = \sigma_{\tau 2}^-$	$\sigma_{\tau 1}^- = \tau^*$	$\sigma_{\tau 1}^- = \beta^* \sigma_n^-$	$\sigma_{\tau 1}^+ = 0$
$u_n^+ = u_n^-$	$\sigma_{\tau 2}^+ = \tau^*$	$\sigma_{\tau 2}^+ = \beta^* \sigma_n^+$	$\sigma_{\tau 1}^- = 0$
$u_{\tau 1}^+ = u_{\tau 1}^-$	$\sigma_{\tau 2}^- = \tau^*$	$\sigma_{\tau 2}^- = \beta^* \sigma_n^-$	$\sigma_{\tau 2}^+ = 0$
$u_{\tau 2}^+ = u_{\tau 2}^-$	$u_n^+ = u_n^-$	$u_n^+ = u_n^-$	$\sigma_{\tau 2}^- = 0$

Таблица 2

Условия на трещине сдвига

Трещины нет	Трещина есть	
	сжатие ($\Delta u_n < 0$)	растяжение ($\Delta u_n \geq 0$)
$\sigma_n^+ = \sigma_n^-$	$\sigma_n^+ = \sigma_n^-$	$\sigma_n^\pm = 0$
$\sigma_{\tau 1}^+ = \sigma_{\tau 1}^-$	$\sigma_{\tau 1}^\pm = \mp \beta \sigma_n \operatorname{sgn}(V_{\tau 1}^+ - V_{\tau 1}^-)$	$\sigma_{\tau 1}^\pm = 0$
$\sigma_{\tau 2}^+ = \sigma_{\tau 2}^-$	$\sigma_{\tau 2}^\pm = \mp \beta \sigma_n \operatorname{sgn}(V_{\tau 2}^+ - V_{\tau 2}^-)$	$\sigma_{\tau 2}^\pm = 0$
$u_i^+ = u_i^-, i = 1, 2, 3$	$u_n^+ = u_n^-$	

рон трещины; $u_x^\pm, u_y^\pm, u_z^\pm$ — перемещения вдоль соответствующих осей с обеих сторон трещины; σ^*, β^* — опытные константы; $\tau^* = -\beta^* \sigma^*$, где τ^* — прочность материала на сдвиг; $\Delta u_n = u_n^+ - u_n^-$ вычисляются на каждом шаге моделирования по формуле $\Delta u_n = \int_0^h (V_n^+ - V_n^-) dt$, где V^\pm — скорости берегов трещины; h — шаг моделирования по времени. Перед расчетом переменных на следующем шаге по времени локальные условия $\Delta u_n < 0$ корректируются условиями $V_n^+ = V_n^-$ (так как ячейки сетки не могут входить друг в друга).

В табл. 2 показана последовательность условий, реализуемых на трещине сдвига, образовавшейся после выполнения условия $\sigma_{\tau 1} \geq \tau^*$ (или $\sigma_{\tau 2} \geq \tau^*$).

На четвертом этапе проводится моделирование динамического разрушения конструкции при различных нагружающих напряжениях P_f . В качестве критерия полного разрушения конструкции принимается полное разрушение какого-либо ее сечения. Анализ разрушения проводится автоматически,

а выполнение указанного критерия используется для окончания счета. Предполагается также, что разрушение происходит за время, не превышающее 30 времен пробега продольной волны по максимальному размеру куба.

На пятом этапе массивы, полученные в момент полного разрушения конструкции, подвергаются анализу с целью определения областей катастрофического разрушения и характера разрушений в них.

И, наконец, на последнем, шестом, этапе решается задача определения минимальных нагрузок, приводящих к полному разрушению конструкции.

2. Модель и метод

При моделировании используется метод пространственных характеристик, для чего исходная система уравнений динамики приводится (как, например, в работе [2]) к характеристической форме. В безразмерном виде для волн, движущихся вдоль оси x_α , эти уравнения записываются в виде систем:

$$(\partial/\partial\tau \pm \partial/\partial x_\alpha)(\sigma_{\alpha\alpha} \mp V_\alpha) = \partial/\partial x_\beta (v_1 V_\beta \pm \sigma_{\alpha\beta}) + \partial/\partial x_\gamma (v_1 V_\gamma \pm \sigma_{\alpha\gamma}) - F_{\alpha\alpha} - v_1(F_{\beta\beta} + F_{\gamma\gamma}); \quad (1)$$

$$(\partial/\partial\tau \pm \xi\partial/\partial x_\alpha)(\sigma_{\alpha\beta} \mp \xi V_\beta) = \partial/\partial x_\beta (v_2 V_\alpha/2 \pm \xi\sigma_{\beta\beta}) \pm \partial/\partial x_\gamma (\xi\sigma_{\beta\gamma}) - F_{\alpha\beta}; \quad (2)$$

$$(\partial/\partial\tau \pm \xi\partial/\partial x_\alpha)(\sigma_{\alpha\gamma} \mp \xi V_\gamma) = \partial/\partial x_\gamma (v_2 V_\alpha/2 \pm \xi\sigma_{\gamma\gamma}) \pm \partial/\partial x_\beta (\xi\sigma_{\beta\gamma}) - F_{\alpha\gamma}; \quad (3)$$

$$\partial/\partial\tau(\sigma_{\beta\beta} - v_1\sigma_{\alpha\alpha}) = v_2(1 - v)^{-1}(\partial V_\beta/\partial x_\beta + v\partial V_\gamma/\partial x_\gamma - F_{\beta\beta} - vF_{\gamma\gamma}); \quad (4)$$

$$\partial/\partial\tau(\sigma_{\gamma\gamma} - v_1\sigma_{\alpha\alpha}) = v_2(1 - v)^{-1}(\partial V_\gamma/\partial x_\gamma + v\partial V_\beta/\partial x_\beta - F_{\gamma\gamma} - vF_{\beta\beta}); \quad (5)$$

$$\partial/\partial\tau\sigma_{\beta\gamma} = v_2[(\partial V_\beta/\partial x_\gamma + v\partial V_\gamma/\partial x_\beta)/2 - F_{\beta\gamma}], \quad (6)$$

где переменные приведены к безразмерному виду с помощью соотношений

$$\tilde{x}_i = x_i/x_0, \tau = c_0 t/x_0, \tilde{\sigma}_{ij} = \sigma_{ij}/k_s, \tilde{S} = S/k_s, \tilde{V}_i = V_i/V_s, \quad (7)$$

$$k_s = \rho c_0 V_s, c_0 = \sqrt{(2\mu + \lambda)/\rho}, \delta = x_0/(V_s \tau_0),$$

в которых x_i, t — координаты и время; σ_{ij} — напряжения; $\Phi(S - k_s)$ — функция пластичности; $S = \sqrt{s_{ij}s_{ij}/2}$ — интенсивность напряжений в теле; $s_{ij} = \sigma_{ij} - \delta_{ij}p$ — девиатор напряжений, $p = \sigma_{ii}/3$ — гидростатическое давление; k_s — статический предел текучести материала тела при сдвиге; V_i — скорости частиц тела; μ, λ и ν — параметры Ламе и коэффициент Пуассона; x_0 — нормировочная константа, за которую принята длина грани куба; τ_0 — время

задержки текучести; c_0 — скорость продольной волны в теле; V_s — скорость частиц тела, при которой начинается пластическое течение; $v_1 = v/(1 - v)$, $v_2 = 1 - v_1$, $\alpha \neq \beta \neq \gamma \neq \alpha$, $\alpha, \beta, \gamma = 1, 2, 3$, $\xi = c_{\perp}/c_0$ — отношение скорости поперечной волны $c_{\perp} = \sqrt{\mu/\rho}$ к скорости продольной волны c_0 , по повторяющимся латинским индексам проводится суммирование, а по повторяющимся греческим индексам суммирования нет.

Для упруговязкопластического тела функции в правых частях уравнений имеют вид: $F_{\eta\zeta} = \delta \frac{\bar{\Phi}(S-1)}{S} s_{\eta\zeta}$, $\eta, \zeta = \alpha, \beta, \gamma$. Для упрощения вычислений выбираем $\bar{\Phi}(S-1) = \sqrt{S-1}$ при $S > 1$ и $\bar{\Phi}(S-1) = 0$ при $S \leq 1$.

Критерии образования трещин выбираем с учетом накопления повреждений в виде $J_n = A$ или $J_{\tau} = B$, где $J_n = \int_{\tau_0}^{\tau} \langle \sigma_n(\tau)/\sqrt{3} - 1 \rangle d\tau$, $J_{\tau i} = \int_{\tau_1}^{\tau} \langle \sigma_{\tau i}(\tau) - 1 \rangle d\tau$, $i = 1, 2$, а константы A и B определяются экспериментально.

Здесь $\langle f \rangle = f$ при $f > 0$ и $\langle f \rangle = 0$ при $f \leq 0$; $A = B = 1$; τ_0 и τ_1 — моменты времени, в которые соответствующие напряжения впервые выходят в пластическую область.

До выполнения критериев разрушения во всех внутренних точках тела предполагается непрерывность скоростей частиц и напряжений. Далее предполагается, что трещина развивается в соответствии с условиями, представленными в табл. 1 и 2.

3. Постановка задач

На рис. 2, 3 представлены схемы нагружения для рассматриваемых задач.

В первой задаче полый куб подвергается импульсному нагружению продуктами точечной детонации

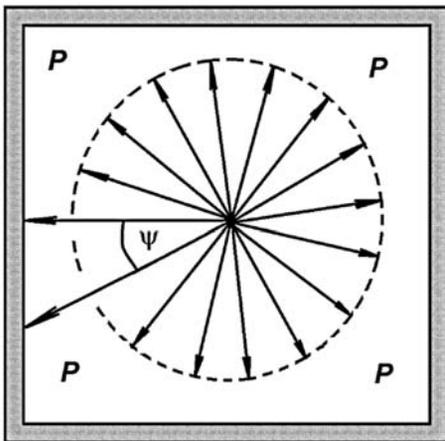


Рис. 2

заряда, расположенного в геометрическом центре куба. Задача исследования разлета продуктов взрыва решается приближенно аналитически [2, с. 581—604] и сводится к определению напряжений на внутренних поверхностях куба. Далее эти напряжения используются в качестве граничных условий при определении нижнего предела P_f разрушения куба. На верхней и боковых внешних гранях куба напряжения отсутствуют; на нижней внешней грани заданы условия жесткой заделки.

Схема нагружения показана на рис. 2. Угол ψ образован лучами, исходящими из точки подрыва в направлении нормали к внутренней границе куба и в произвольную (рассматриваемую) точку.

Задача разлета продуктов детонации сферически симметрична, исходные уравнения имеют следующий вид:

$$\begin{cases} \frac{\partial u}{\partial t} + u \frac{\partial u}{\partial r} + \frac{1}{\rho} \frac{\partial p}{\partial r} = 0, p = A\rho^k, \\ \frac{\partial p}{\partial t} + u \frac{\partial p}{\partial r} + \rho \frac{\partial u}{\partial r} + \frac{2\rho u}{r} = 0, \end{cases} \quad (8)$$

где ρ — плотность газа; p — давление в газе; u — скорость частиц газа; k — показатель изоэнтропы, для конденсированных взрывчатых веществ (ВВ) примерно равный 3.

Решение задачи о точечном взрыве конденсированного ВВ с использованием соотношений (7) позволяет определить асимптотику зависимостей давления p в продуктах детонации от радиуса r и времени t в следующем виде [2, с. 586]:

$$p = Br^{-3}t^{-1,2}. \quad (9)$$

Для определения величины B обратимся к рис. 3. При длине ребра внутренней полости $l_* = l - 2\Delta$ расстояние от точки подрыва до внутренней грани равно $l_*/2$, а до любой точки на этой грани — $r_0 = l_*/2\cos\psi$. Максимальное значение угла ψ легко определить из геометрических соображений, оно равно $\psi_{\max} = \arccos(1/\sqrt{3}) = 60^\circ 48'$.

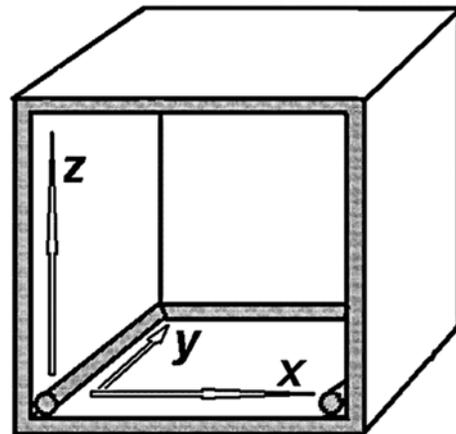


Рис. 3

Рассмотрим регулярное косое отражение волны, распространяющейся из центра взрыва в плоскости, образованной лучами (изображенными на рис. 3), составляющими угол ψ друг с другом, если [2, с. 427, 439]

$$p_{\perp}/p_0 = 2,4 = 1/(1 - \cos\psi_*), \quad (10)$$

где p_0 — напряжение на фронте падающей детонационной волны в точке соприкосновения ее с внутренней поверхностью куба; p_{\perp} — напряжение в отраженной детонационной волне (с $k = 3$), распространяющейся перпендикулярно грани куба; правая часть формулы соответствует косому отражению плоской детонационной волны от стенки и пригодна для углов $\psi \geq \psi_*$.

Формула (9) позволяет определить $\psi_* = \arccos(1 - p_0/p_{\perp}) = 60^{\circ}21'$. Сравнивая ψ_{\max} и ψ_* можно считать, что они совпадают, и записать уравнение (9) в виде

$$p = 2,4p_0 B' r^{-3} t^{-1,2}. \quad (11)$$

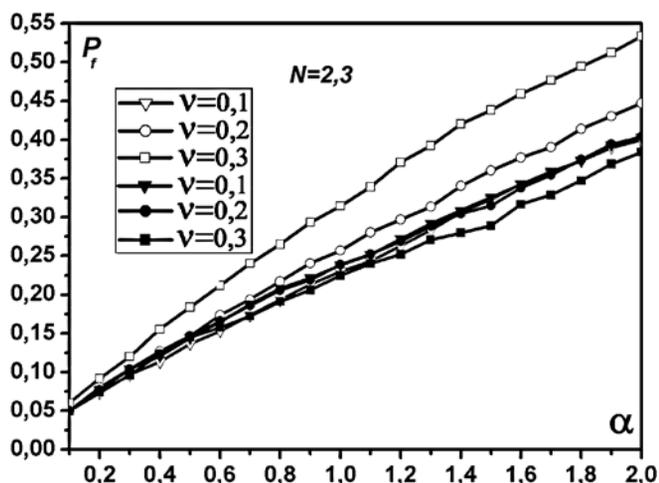


Рис. 4

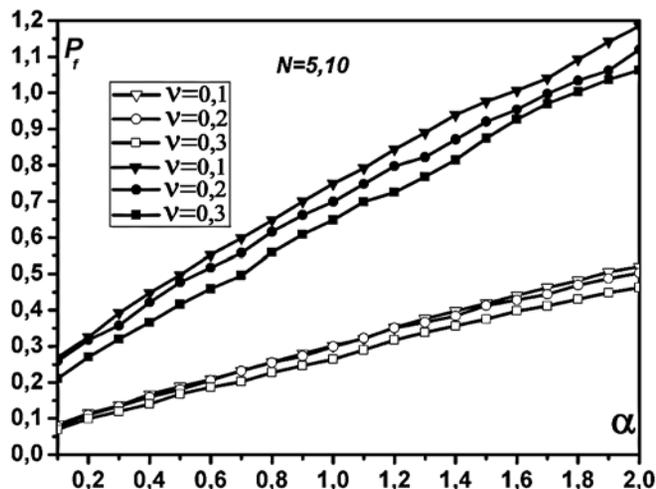


Рис. 5

Считая, что отсчет времени начинается с момента касания падающей волной стенки и что при этом нормальное напряжение на стенке в точке касания равно σ_0 , можно найти B' и записать (10) в следующем виде:

$$p = 2,4\sigma_0 \cos^3\psi (2\alpha\beta\tau + 1)^{-1,2}, \quad (12)$$

где $\alpha = D/c_0$, $\beta = l/l_*$, а σ_0 будет определяться по результатам моделирования из условия $\sigma_0 = P_f$, τ — безразмерное время, определенное в формуле (7).

К разным точкам внутренних поверхностей куба волны подходят не одновременно. Время нарастания "пятна", на котором происходит взаимодействие волны с гранями куба

$$\Delta\tau_{\max} = 2^{-1}\alpha^{-1}\beta^{-1}(\cos^{-1}\psi_{\max} - 1). \quad (13)$$

Радиусы "пятна" при $0 \leq \tau \leq \Delta\tau_{\max}$ будут иметь такой вид:

$$r = l\sqrt{\alpha\beta\tau(1 + \alpha\beta\tau)}. \quad (14)$$

Формулы (1)–(6) и (12)–(14) будем использовать при моделировании разрушения куба продуктами детонации; при этом формула (13) задает граничное условие для нормального напряжения на стенках куба (касательные напряжения будем считать нулевыми). Моделирование начинается в момент подхода детонационной волны к грани куба.

Для тонкостенных кубов ($\delta_* = 0,02$ и $\delta_* = 0,03$) для каждой грани задавались критерии для J_n и J_{τ} . В толстостенных кубах ($\delta_* = 0,05$ и $\delta_* = 0,1$) возникает сложноподвиженное состояние, поэтому для них J_n и J_{τ} и соответствующие им условия на трещинах рассчитывали по всем осям координатной сетки (шесть критериев).

Параметры моделирования. Моделирование проводили на сетке $100 \times 100 \times 100$ ячеек, толщина стенок составляла 2, 3, 5 и 10 ячеек. Отношения $\alpha = D/c_0$ и $\beta = l/l_*$ задавали как параметры. Коэффициент Пуассона задавали равным 0,1, 0,2 и 0,3.

Результаты моделирования. Графики зависимостей $P_f(\alpha)$ при различных толщинах стенок куба (в ячейках) N и коэффициентах Пуассона ν представлены на рис. 4 и 5. Темными маркерами отмечены графики при $N = 3$ и $N = 10$ соответственно. Остальные обозначения приведены на рисунках. Все графики имеют нарастающий характер.

Зафиксируем величины c_0 и ν , связанные с кубом, и рассмотрим формулу (12). Для заданного p с ростом α и β , т. е. скорости детонации и толщины куба, величина σ_0 , определяющая минимальную разрушающую нагрузку, должна возрасти, что и следует из графиков. При этом происходит как бы возрастание и укорочение разрушающего импульса, действующего на куб. При больших N ($N = 10$) в стенках куба возникает сложный процесс трещинообразования; этот процесс наиболее заметен при больших α , когда нагружающий импульс становится

малым и остаточное разрушение происходит именно за счет движения и слияния трещин. Представляет интерес и зависимость P_f от v . Для кубов с тонкими стенками ($N=2$) наибольшие значения P_f достигаются при $v=0,3$. Разрушение происходит в основном за счет растяжения стенок.

С возрастанием толщины стенок ($N=3, N=5$) зависимость P_f от v ослабевает и меняется на противоположную для кубов с толстыми стенками ($N=10$). В этом случае процесс разрушения достаточно сложен: трещины могут двигаться как в одном направлении, так и в разных (частично блокируя друг друга); вблизи угловых точек куба может происходить разрушение сдвигом.

Во второй задаче тонкий детонационный шнур круглого сечения размещен по внутреннему периметру того же куба и активирован с одного конца (см. рис. 3). Скорость детонации D , радиус шнура $r \ll l$, детонация происходит в центре сечения шнура, размерами области детонации в шнуре пренебрегаем.

Таким образом, задача определения нагрузок, действующих на куб, сводится к определению давления, создаваемого точечным движущимся источником детонации и волны разгрузки, распространяющейся от него со скоростью $c = D/2$ [2, рис. 72, с. 558], в системе координат, связанной с источником. Скорость частиц в этой волне $V = D - c = D/2$ и направлена в ту же сторону, что и волна, движущаяся по шнуру. Форма этой волны — четверть сферического сектора с вершиной в точке детонации и осью, отстоящей на расстоянии r_0 — радиуса детонационного шнура — от смежных внутренних граней куба. Плотность продуктов детонации в этой волне $\rho = A/R^2$, где R — расстояние от точки детонации до рассматриваемой точки на смежных гранях ($r_0 \leq R \leq Dt/2$), а константа A может быть определена из условия

$$p(R = r_0, \psi = 0) = 2,4AD^2/4R^2 = 2,4\rho_0 D^2/4, \quad (15)$$

откуда $A = \rho_0 r_0^2$, где ρ_0 — плотность взрывчатого вещества; при этом общий вид нормального давления на стенки в этой волне, с учетом (10),

$$p(R, \psi) = \frac{\rho_0 r_0^2 D^2}{4(1 - 7\cos\psi/12)R^2}, \quad (16)$$

а области на левой внутренней и нижней гранях куба, на которые действует это давление, при $0 < t \leq l_*/D$ могут быть заданы системой неравенств

$$R = \sqrt{(Dt - y)^2 + \xi^2}, \quad r_0 \leq R \leq Dt/2, \quad Dt/2 \leq y \leq Dt, \\ r_0 \leq \xi \leq Dt - y, \quad \xi = x, z. \quad (17)$$

При $t = l_*/D$ детонация перейдет на нижнее ребро задней стенки куба и напряжение (17) будет действовать на заднюю и нижнюю стенки, однако

продукты детонации, полученные при $0 < t \leq l_*/D$, будут продолжать двигаться к задней стенке со скоростью V , создавая на ней дополнительное давление

$$p_1 = \rho_1 V^2, \quad (18)$$

где $\rho_1 = A/R^2$, а константа A определяется из условия равенства масс использованного взрывчатого вещества в шнуре и продуктов детонации (образующих четверть сферического сектора) в момент $t = l_*/D$, т. е. условия

$$M = \pi r_0^2 \rho_0 l_* = \frac{\pi}{2} \int_0^{\pi/4} \int_0^{l_*/2} \frac{A}{R^2} R^2 \sin\phi dR d\phi, \quad (19)$$

откуда $A = 13,66\rho_0 r_0^2$ и $p_1 = 3,414\rho_0 r_0^2 D^2/R^2$, где $R^2 = V^2 t_1^2 + x^2 + z^2 \leq l_*^2/4$ и $t_1 = t - l_*/D$. Площадь пятна, на которое действует это давление, определяется следующими условиями: $r_0 \leq R_* = Vt_1$ при

$0 \leq t_1 \leq l_*/\sqrt{2} D$ и $R_* = \sqrt{l_*^2/4 - V^2 t_1^2}$ при $l_*/\sqrt{2} D \leq t_1 \leq l_*/D$. При $t_1 = l_*/D$ оба процесса — движения точки детонации вдоль нижнего ребра задней грани и давления продуктов детонации на эту грань — заканчиваются.

Далее принимаем $t_2 = t_1 - l_*/D$ и рассматриваем эти процессы на правой внутренней грани куба, затем при $t_3 = t_2 - l_*/D$ на ближней к нам боковой грани. Давление на нижнюю грань определяется формулой (16) и сохраняется до окончания процесса детонации. К этому времени средняя плотность продуктов детонации станет $4\pi r_0^2 \rho_0 / l_*^2 < 0,002\rho_0$ и ее давлением будем пренебрегать.

Параметры моделирования — те же, что и в предыдущей задаче, радиус шнура r_0 равняется одной ячейке.

Результаты моделирования представлены на рис. 6, 7.

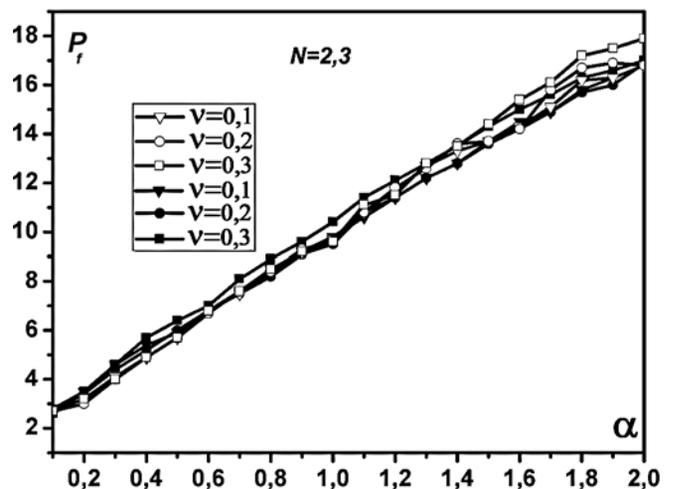


Рис. 6

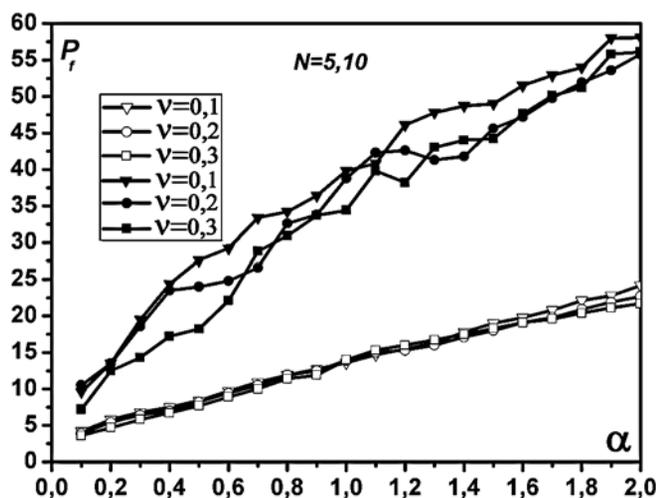


Рис. 7

Рассчитываемой величиной является минимальное давление $P_f(\alpha, v, N)$, определяемое формулой (16), при которой происходит полное разрушение куба в одном из его поперечных сечений. Значение этой величины получаем в результате серии численных экспериментов. Обозначения на рис. 6, 7 те же, что и на рис. 4, 5.

Для определения наиболее эффективного (из двух рассмотренных способов) способа расположения взрывчатого вещества предположим, что сферический заряд, находящийся в центре куба, изготовлен из того же количества ВВ, что и шнур. При этом его радиус $R_s = \sqrt[3]{3r_0^2 l^*}$, а начальное давление, создаваемое им на поверхности куба, определяем из выражения $\sigma_0 = p_H(2R_s/l^*) = (0,002...0,004)p_H$, где

p_H — начальное давление в продуктах детонации. В то же время начальное давление, действующее на боковую стенку куба при детонации шнура, согласно (16) составляет $0,6p_H$, что значительно больше. Таким образом, использование детонационного шнура выгоднее.

Заключение

Использование метода пространственных характеристик позволяет исследовать динамику и динамическое разрушение тел с полостями, в частности, исследовать динамическое разрушение тел прямоугольного сечения при взрывной нагрузке, при этом вновь разработанные алгоритмы оказались достаточно устойчивыми и быстрыми.

Решение задач газовой динамики выполнено приближенно, в соответствии с методиками, изложенными в работе [3]. В данном проекте они носят вспомогательную роль, обеспечивая импульсное нагружение внутренней поверхности разрушаемого тела, в том числе и движущимися нагрузками.

Под действием таких нагрузок исследуемое тело находится в сложнапряженном состоянии, поэтому используются различные критерии разрушения и условия на трещинах.

Список литературы

1. **Баяковский Ю. М., Галактионов В. А., Михайлова Т. Н.** Графор. Графическое расширение Фортрана. М.: Наука, 1985. 288 с.
2. **Бульчев Г. Г.** Построение матричной характеристической формы уравнений динамики анизотропных упруговязкопластических сред // Известия РАН. Механика твердого тела. 1995. № 1. С. 91–95.
3. **Станюкович К. П.** Неустановившиеся движения сплошной среды. М.: Наука, 1971. 854 с.

G. G. Bulychev, Professor, e-mail: geo-bulychev@mail.ru,
Moscow State University of RadioEngineering, Electronics and Automation

Numerical Modeling of Dynamical Destruction Hollow Bodies by Detonation Products

By numerical method of spatial characteristics there is solved the problem of impulsive destruction of thin-walled isotropic body. The impulse is made by point explosion in the center of body or by moving detonation point of explosive cord, located along the perimeter of the inner cavity of the body. Limit of destruction's stress is determined by as function of velocity of detonation, thickness of walls and Poisson's factor of a body material.

Keywords: a numerical simulation, a mechanics of the deformed solid body, dynamics, destruction, thin-walled bodies

References

1. **Bayakonskii Yu. M., Gakaktionov V. A., Mikhailova T. N.** *Графор. Графическое расширение Фортрана.* Moscow: Nauka, 1985. 288 p.

2. **Bulychev G. G.** Postroenie matrichnoi kharakteristicheskoi formi uravnenij dinamiki anizotropnich uprugoplastichnikh sred. *Izvestiya RAN. Mekhanika tverdogo tela.* 1995. N. 1. P. 91–95.

3. **Stanyukovich K. P.** *Neustanovivshiesya dvizheniya sploshnoi sredy.* Moscow: Nauka, 1971. 854 p.

БЕЗОПАСНОСТЬ ИНФОРМАЦИИ CRYPTOSAFETY INFORMATION

УДК 004.056.53

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

Технология защиты данных, обрабатываемых в распределенных информационных системах

Рассмотрен новый подход к защите данных, обрабатываемых в распределенных информационных системах, основанный на реализации методов контроля доступа к создаваемым объектам — к файловым объектам и к буферу обмена, позволяющих исключить объект доступа из разграничительной политики за счет автоматической разметки создаваемых объектов. Практическая реализация данного подхода при условии сохранения разметки (создаваемых атрибутов) непосредственно в создаваемом файле позволяет сформулировать и решить задачу реализации разграничительной политики доступа к данным, обрабатываемым в распределенной информационной системе, с учетом различных возможных способов обмена данными между компонентами (компьютерами) подобной системы. При этом реализуется управление потоками данных уже в рамках системы в целом.

Ключевые слова: распределенная информационная система, защита данных, несанкционированный доступ, контроль и разграничение прав доступа, разграничительная политика, создаваемый объект, управление потоками

Введение

Решение задачи защиты информации от несанкционированного доступа в любой информационной системе основано на реализации контроля и разграничений прав доступа субъектов к защищаемым объектам (далее — контроля доступа), прежде всего к файловым объектам, поскольку именно они предназначены для хранения обрабатываемых данных. При этом существуют различные способы обмена данными (в рассматриваемом случае — файлами) между компонентами (компьютерами) распределенной информационной системы. Это обуславливает актуальность задачи управления потоками данных в распределенной системе, при котором при передаче файла с одного компьютера информационной системы на другой права доступа к этому файлу будут передаваться вместе с файлом. Как следствие, и на другом компьютере будут действовать исходно задаваемые разграничительной политикой права доступа к передаваемым между компьютерами файлам. Разграничительная же политика доступа при этом должна задаваться не в рамках отдельного компонента (компьютера) распределенной информационной системы, а применительно к системе в целом. Поскольку широко используемым на практике способом дополнительной защиты обрабатываемых в информационной системе данных является шифрование, то все сказанное в части проводимых в работе исследований относится и

применительно к задаче криптографической защиты данных, направленной на реализацию управления потоками зашифрованных данных в распределенной информационной системе.

Принципы контроля доступа к создаваемым объектам

Реализация контроля доступа основывается на использовании одной из соответствующих абстрактных моделей [1, 2]. Наиболее широко сегодня используются модели дискреционного, мандатного и ролевого контроля доступа.

Дискреционный (иногда также называют избирательным) контроль доступа (Discretionary Access Control — DAC) основан на реализации модели "Харрисона—Руззо—Ульмана" [3]. Основу построения разграничительной политики доступа в данном случае составляет задание администратором матрицы доступа — списка правил доступа субъектов к объектам либо, наоборот, к объектам субъектов, что реализуется транспонированием матрицы доступа. Дискреционный метод контроля доступа может быть реализован с произвольным либо с принудительным для пользователей управлением потоками данных (в зависимости от того, включен ли непривилегированный пользователь, как "владелец" создаваемого объекта, в схему администрирования) [4].

Мандатный контроль доступа (Mandatory Access Control — MAC) основан на реализации абстракт-

ной модели "Белла—ЛаПадулы" [5]. Это контроль доступа с принудительным управлением потоками информации, основанный на формализации задания правил с использованием меток безопасности (мандатов) — числовых значений, отражающих соответствующие уровни безопасности субъектов (уровни доступа) и объектов (уровни конфиденциальности) в заданной иерархии. Каждому субъекту и объекту системы назначается некоторый уровень безопасности — присваивается метка безопасности. Реализация разграничительной политики доступа предполагает арифметическое сравнение этих меток на основе исходно заданного правила.

Идея *ролевой* модели контроля доступа (Role-Based Access Control — RBAC) [1] основана на максимальном приближении логики работы системы к реальному разделению функций персонала в организации. Применение данного метода подразумевает определение ролей в системе, где роль интерпретируется как совокупность действий и обязанностей, связанных с соответствующим видом деятельности. На самом деле, ролевая модель — это не что иное, как дискреционный контроль доступа при реализации соответствующей групповой политики доступа (разграничительной политики для групп пользователей). К достоинствам же данной модели можно отнести возможность определенной формализации ролей и, как следствие, возможность задания и последующего тиражирования неких типовых разграничительных политик доступа для соответствующих ролей. Таким образом, к базовым можно отнести абстрактные модели дискреционного и мандатного методов контроля доступа.

Важным для нас в данном случае является то, что как существующие абстрактные модели контроля доступа, так и (как следствие) реализующие их технические решения используют две равноправные сущности — субъект и объект доступа, а назначение правил предполагает задание тем или иным способом того, какие субъекты к каким объектам (или наоборот) какие права доступа имеют. При этом в качестве субъектов доступа в разграничительной политике выступают пользователи, идентифицируемые учетными записями, — именно в отношении пользователей задаваемыми правилами ограничиваются возможные действия, которые потенциально могут нанести вред.

Применительно к решению задачи защиты данных, обрабатываемых в информационной системе, следует говорить о защите создаваемых пользователями в процессе работы объектов (создаваемые файлы и буфер обмена), поскольку именно такие объекты предназначены для хранения обрабатываемых в системе данных. Подобная постановка задачи позволяет предложить совершенно новые подходы и разработать новые методы контроля доступа, устраняющие недостатки известных методов, приме-

нительно к решению этой задачи выявленные и изложенные, например, в работе [6].

Предлагаемые принципы контроля доступа к создаваемым объектам [6], основанные на их автоматической разметке при создании или модификации объекта, позволяют исключить сущность "объект доступа" из разграничительной политики доступа. Состоят они в следующем:

- сущность "объект" исключается из схемы контроля доступа, при реализации разграничительной политики используются две сущности: идентификатор (учетная информация) субъекта, создавшего объект, и идентификатор субъекта, запрашивающего доступ к созданному объекту;
- правила доступа устанавливаются между сущностями: "субъект доступа (учетная информация), запрашивающий доступ к объекту" и "субъект доступа (учетная информация), создавший этот объект";
- при создании (модификации) субъектом объекта объект наследует учетную информацию субъекта доступа, создавшего этот объект — объект размечается (учетная информация субъекта сохраняется в атрибутах созданного им объекта);
- при запросе доступа к любому объекту диспетчер доступа (решающий элемент) получает разметку этого объекта, считывая его атрибуты, и анализирует запрос на непротиворечивость заданным правилам доступа, в результате чего предоставляет запрошенный субъектом доступ к объекту либо отказывает в нем.

Таким образом, реализуется разграничительная политика (задаются правила доступа) не для субъектов к объектам, а между субъектами доступа к создаваемым ими объектам.

Контроль доступа к создаваемым объектам*

Мандатный метод контроля доступа к создаваемым файлам. Метки безопасности (уровни доступа), или мандаты, присваиваются исключительно пользователям (интерактивным пользователям) [8]. Для любого заведенного в системе пользователя может быть задан (выбран) уровень доступа. При этом метки безопасности могут назначаться не всем пользователям, а только тем, которые создают файлы, доступ к которым будет контролироваться и разграничиваться (обрабатываемые этими пользователями данные требуется защищать). Вот и все настройки разграничительной политики доступа (как и последующие иллюстрации, приводимые в работе) рассмотрены на примере интерфейса, реализованного в "Комплексной системе защиты данных "Панцирь+" для ОС Microsoft Windows" (рис. 1).

*Техническое решение, реализующее рассматриваемые далее методы контроля доступа к создаваемым объектам, запатентовано [7].

Отметим, что при настройке разграничительной политики доступа в данном случае не требуется назначения меток безопасности файловым объектам, чем обуславливаются ключевые недостатки известного метода мандатного контроля доступа (возникают проблемы включения в схему контроля системных объектов, необходимость разделения каким-либо образом неразделяемых каталогов, например временных папок, и др.).

Рассмотрим, как работает диспетчер доступа.

Метки безопасности назначаются контролируемым пользователям — тем пользователям, которые создают файлы, к которым требуется разграничивать права доступа. При создании файла любым подобным пользователем этот файл автоматически размечается диспетчером доступа: в атрибуты файла автоматически помещаются учетные данные субъекта (в данном случае его уровень доступа — мандат), создавшего этот файл. Подобным образом будет размечаться и неразмеченный ранее файл при его модификации контролируемым пользователем.

При обращении к созданному в процессе работы системы файлу диспетчер доступа анализирует наличие у файла разметки и при ее наличии (в противном случае права доступа не разграничиваются) диспетчер анализирует соответствие запроса мандатным правилам доступа посредством арифметического сравнения соответствующих меток безопасности (мандатов) пользователя, запросившего доступ к файлу, и файла, унаследовавшего данную метку от пользователя, создавшего этот файл.

Отметим, что принципиальным достоинством данного метода, кроме кардинального упрощения задачи администрирования, является корректность реализации мандатной схемы контроля доступа в общем случае, так как где бы (в какой бы папке) и при каких условиях не создавался (модифицировался) бы файл контролируемым пользователем, этот файл будет однозначно размечен и в отношении него будет действовать заданная разграничительная политика доступа при последующих обращениях.

Дискреционный метод контроля доступа к создаваемым файлам. Дискреционный метод контроля доступа может быть использован для решения двух наиболее актуальных современных задач защиты информации: защиты от вредоносных программ (включая атаки на повышение привилегий, предполагающие запуск вредоносной программы с системными правами) и защиты данных от атак со стороны приложений, наделяемых вредоносными свойствами.

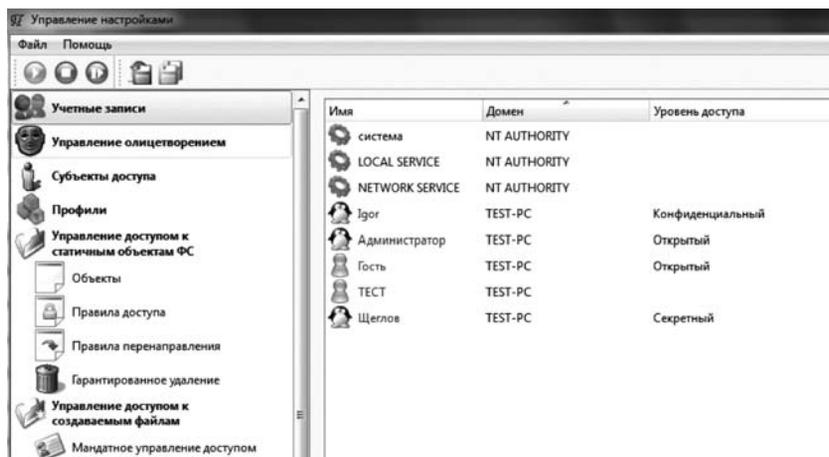


Рис. 1. Иллюстрация разграничительной политики доступа для мандатного метода контроля доступа к создаваемым файлам

Субъект доступа в данном случае уже идентифицируется тремя сущностями [9]:

- исходный идентификатор пользователя (от лица которого запущен процесс);
- эффективный идентификатор пользователя (от лица которого процесс обращается к объекту);
- процесс (полнопутевое имя исполняемого файла процесса) задается из интерфейса, приведенного на рис. 2 (см. четвертую сторону обложки).

При задании идентификатора пользователя (как исходного (первичного), так и эффективного) может использоваться маска "*" — "Любой" (в этом случае заданные правила будут распространяться на всех пользователей). Имя процесса может задаваться либо полнопутевым именем его исполняемого файла, либо маской (возможно также использование переменных среды окружения). Например, маской C:\ProgramFile* покрываются все исполняемые файлы из данного каталога, маской "*" задается то, что правило будет применимо к любому процессу.

Правила доступа к создаваемым файлам задаются администратором из интерфейса и отображаются в интерфейсе, приведенном на рис. 3 (см. четвертую сторону обложки; субъекты доступа здесь отображаются присвоенными им при создании именами, см. рис. 2).

Отметим, что в назначаемые права доступа (см. рис. 3) не внесено право "исполнение", так как запрет исполнения создаваемых файлов должен задаваться по умолчанию, что является эффективным решением по защите от вредоносных программ [10].

Задание разграничительной политики доступа осуществляется следующим образом. Из списка заданных субъектов доступа (см. рис. 3) в поле "Выберите субъекты создателей" задаются контролируемые субъекты доступа.

Применительно к выбранному контролируемому субъекту создателю файла назначаются права доступа к создаваемым им файлам других субъектов.

С этой целью субъект, которому назначаются права доступа, выбирается в поле "Выберите субъектов, осуществляющих доступ" (рис. 3). Для выбранной пары субъектов (в левом и в правом полях интерфейса) соответствующим образом разрешаются либо запрещаются соответствующие права доступа (чтение, запись, удаление, переименование). Заданное правило отображается соответствующей строкой в интерфейсе.

Диспетчер доступа в данном случае, по сути, работает так же, как и при реализации мандатного метода, с поправкой на то, что правила доступа для анализа корректности запроса выбираются из соответствующей матрицы доступа. Принципиальным отличием здесь является то, что при создании размечаются все файлы, в том числе файлы, создаваемые и не контролируемые пользователями, так как они должны быть идентифицированы в целях предотвращения возможности их последующего исполнения, в том числе и системными правами.

Важнейшим применением данного метода контроля доступа к создаваемым файлам (соответственно, к данным) является возможность изолирования (по обрабатываемым данным) работы критических приложений [11].

В общем случае при реализации разграничительной политики доступа к создаваемым файлам мандатный и дискреционный механизмы контроля доступа могут использоваться совместно. При этом запрос доступа будет считаться санкционированным в том случае, если он не будет противоречить ни мандатным, ни дискреционным правилам доступа. При этом диспетчером доступа анализируются сначала мандатные правила доступа, затем дискреционные.

Отметим, что применение данных методов существенно сказывается и на реализации иных методов защиты данных, например, на гарантированном удалении. В данном случае правила гарантированного удаления должны задаваться не для папок, сохраняемые файлы в которых будут автоматически удаляться, а для субъектов доступа, создаваемые файлы которыми (в какой бы папке они не создавались) должны гарантированно удаляться [12]. Отметим, что в данном случае опять же можно говорить о простоте администрирования и корректности решения соответствующей задачи защиты. Ведь если устанавливать правило гарантированного удаления для папок, необходимо его задавать и для всех папок хранения временных файлов, которые создаются большинством приложений (ведь в них также сохраняются защищаемые данные и, как следствие, остаются в виде остаточной информации на диске при их удалении).

Шифрование и метод контроля доступа к шифруемым создаваемым файлам. В случае использования метода (мандатного, дискреционного или обоих одновременно) контроля доступа к создаваемым

файлам может быть решена задача принудительного хранения информации в зашифрованном виде для субъектов доступа. При этом при настройке политики шифрования файлов уже потребуются задавать не объекты доступа (папки), сохраняемые данные в которых будут автоматически зашифровываться, а субъекты доступа (при мандатном контроле — уровни доступа или метки безопасности — мандаты), при сохранении которыми данных они будут автоматически зашифровываться. Для соответствующих субъектов должны назначаться и ключи шифрования. Учетной же информации субъекта, сохраняемой в качестве атрибута создаваемого (модифицируемого) файла в открытом виде (она не является секретной информацией), достаточно, чтобы выбрать ключ шифрования для расшифрования файла, где бы (в какой бы папке) этот шифруемый файл не был бы создан. Данное техническое решение запатентовано [13].

Контроль доступа к буферу обмена. Поскольку буфер обмена предназначен для временного хранения данных, используемых для обмена данными приложениями, и на момент задания администратором разграничительной политики доступа эти данные еще не созданы, здесь также можно говорить о контроле и разграничении прав доступа к создаваемым объектам (к данным, записываемым в буфер обмена) и, как следствие, применить изложенные выше принципы контроля и разграничения прав доступа.

В данном случае, с определенной оговоркой, следует говорить о целесообразности реализации дискреционного метода контроля доступа, предполагающего включение в субъект доступа процесса. Дело в том, что между учетными записями в общем случае система сама по умолчанию разграничивает права доступа к буферу обмена. Оговорка состоит именно в том, что разграничение в общем случае касается сессий различных пользователей. При запуске же процесса с правами другой учетной записи в одной сессии (без перезагрузки системы или смены пользователя), например, с использованием утилиты "runas", буфер обмена между учетными записями разграничен не будет — данную возможность следует предотвратить.

В части же реализации метода дискреционного контроля доступа к буферу обмена можем отметить, что она полностью аналогична реализации метода контроля доступа к создаваемым файлам. Субъекты доступа, идентифицируемые соответствующими тремя сущностями, задаются из интерфейса, представленного на рис. 2, правила доступа (с учетом их особенностей — разрешение либо запрет получения информации из буфера обмена) — из интерфейса, представленного на рис. 4. Аналогичным же образом работает и диспетчер доступа.

Таким образом, применяя рассмотренные методы (соответствующие их реализации — механизмы за-

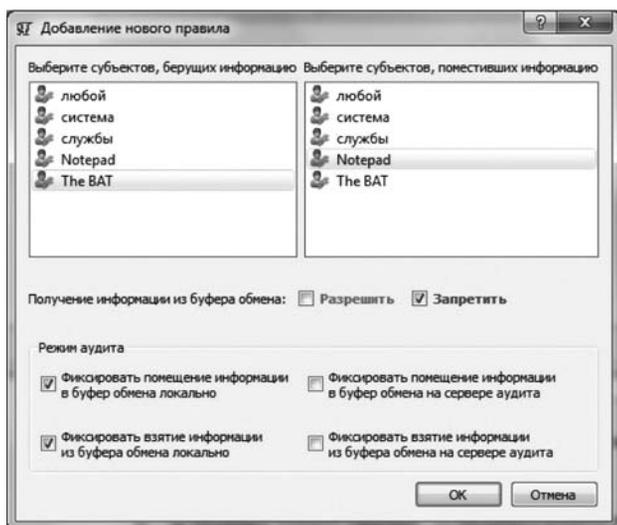


Рис. 4. Задание правил доступа к буферу обмена

щиты) можно реализовать полностью изолированную обработку данных как отдельными пользователями (группами пользователей), так и отдельными приложениями (группами приложений) в информационной системе.

Отметим, что использование предложенных методов контроля доступа к создаваемым объектам, позволяющее рассмотреть задачи защиты данных и системных объектов как совершенно различные задачи защиты даже собственно в своей постановке, принципиально меняет требования к реализации многих механизмов защиты — не только контроля доступа к файловым объектам. Это позволяет говорить о новой технологии защиты данных, обрабатываемых в информационной системе.

Технология защиты данных, обрабатываемых в распределенных информационных системах

Прежде всего рассмотрим постановку задачи защиты. Распределенная информационная система предполагает наличие нескольких компьютеров, осуществляющих обработку информации в рамках единой системы, при возможности обмена между ними данными. Данные в виде файла могут передаваться как с использованием внешних файловых накопителей, так и по сети. В обоих случаях передаваемый файл будем рассматривать как создаваемый объект, к которому должны разграничиваться права доступа субъектов, причем не на отдельном компьютере, а на всех компьютерах в составе распределенной информационной системы. При решении данной задачи разграничительная политика доступа уже может реализовываться не для отдельно взятого компьютера, а для распределенной информационной системы в целом.

Как ранее отмечено, для реализации контроля и разграничения прав доступа каждому созданному

файлу должна быть сопоставлена учетная информация создавшего файл субъекта доступа (соответственно идентификатор субъекта либо метка безопасности, возможно и то и другое, в зависимости от реализуемого метода контроля и разграничения прав доступа). Именно эта учетная информация, являющаяся атрибутом файла, используется диспетчером при анализе запроса на непротиворечивость заданной разграничительной политики доступа. Естественно, что при реализации разграничительной политики доступа для распределенной информационной системы атрибуты файла должны сохраняться непосредственно в файле, что позволит передавать их тем или иным способом вместе с размеченным подобным образом файлом между компьютерами информационной системы.

Рассмотрим реализацию возможных разграничительных политик доступа с учетом реализуемых в системе методов контроля доступа к создаваемым файлам.

Естественно, что наиболее простой в настройке будет разграничительная политика доступа на основе меток безопасности (уровней доступа). Именно уровни доступа (их количественные значения) помещаются в качестве атрибутов в файлы. Список уровней доступа и правила арифметического сравнения меток безопасности создаются для распределенной информационной системы в целом. При создании пользователя на каждом компьютере ему назначается уровень доступа из заданного для системы в целом списка (см. рис. 1). При этом в соответствии с реализуемой технологией распределенной обработки информации на отдельно взятых компьютерах системы может обрабатываться информация не всех уровней доступа. Как следствие, пользователям на одном из компьютеров системы могут назначаться не все уровни доступа из их полного для системы списка.

Рассмотрим, как будет реализован контроль доступа в этом случае. Любой созданный в процессе работы системы файл как на компьютере, на котором он создан, так и на компьютере, на который он каким-либо способом, в том числе по сети, передан, будет иметь в своем составе метку безопасности создавшего его пользователя. Как следствие, доступ к нему на любом компьютере будет возможен только в рамках разграничительной политики, заданной для информационной системы в целом. Это относится и к шифрованию — расшифровать такой файл сможет только пользователь с соответствующим уровнем доступа — меткой безопасности, поскольку ключи шифрования в данном случае присваиваются меткам безопасности, а не конкретным пользователям.

Контроль доступа к буферу обмена как к создаваемому объекту в данном случае необходим для того, чтобы разрешить межмашинный обмен дан-

ными в информационной системе исключительно в виде файлов, что проиллюстрируем далее.

Естественно, что важнейшим требованием к корректности реализации рассмотренного метода контроля доступа будет предотвращение возможности смены атрибута переданного на другой компьютер файла, при его сохранении на этом компьютере. Проиллюстрируем необходимость выполнения данного требования примером, также рассмотрим, каким образом может быть выполнено подобное требование. Пусть данные передаются по сети с использованием электронной почты и пусть, например, для этого используется приложение "The BAT". Для решения рассматриваемой задачи требуется разрешить данному приложению чтение из буфера обмена только записанных в буфер обмена им же данных, что настраивается из интерфейса, представленного на рис. 4 (на работе приложения это не отразится, но передача данных становится возможной только посредством передачи соответствующего файла). Письмо, включая прикрепленный к нему файл, поступающее на удаленный компьютер, полностью (с вложенным в него файлом) при получении автоматически сохраняется приложением в соответствующем файле, который размечается как вновь создаваемый на этом компьютере файл — в разметке этого файла окажется метка безопасности пользователя, под которым запущено приложение (последующий доступ к письму будет возможен только пользователем, обладающим соответствующим уровнем доступа). Таким образом, имеем размеченный файл с письмом, в котором (в соответствующем формате) находится вложение — размеченный файл, полученный по почте с другого компьютера. Если попытаться далее открыть прикрепленный к письму файл, то приложением "The BAT" пользователю будет предложено либо сохранить, либо открыть этот файл. При выборе сохранения файла этот файл будет записан (создан) в выбранном пользователем месте. Как следствие, изменится исходная разметка данного файла — в новой его разметке будет указана метка безопасности пользователя, запустившего приложение "The BAT". Данную смену атрибутов необходимо предотвратить. При выборе же открытия файла приложением "The BAT" будет создан временный файл, который уже далее будет прочитан соответствующим приложением, например редактором "Word". Как видим, и в этом случае меняется разметка исходного файла, который в обоих случаях создается на удаленной машине приложением "The BAT".

Для решения рассматриваемой проблемы достаточно запретить приложению "The BAT" размечать создаваемые им файлы вложений (не файлы писем), т. е. файлы, создаваемые в соответствующей служебной папке. У администратора должна быть возможность задавать приложения и папки, при создании которыми в этих папках файлы раз-

мечаться не будут (либо, наоборот, в зависимости от разграничительной политики, будут).

Что касается реализации дискреционного метода контроля доступа применительно к решению рассматриваемой задачи, то он реализуется по полной аналогии, отличие составляет лишь содержимое атрибутов создаваемых файлов, в том числе передаваемых в составе файлов между компьютерами, и, естественно, способ анализа диспетчером заданных правил доступа. Кроме того, существенно расширяются возможности разграничительной политики доступа в распределенной информационной системе в случае включения в субъект доступа сущности "идентификатор компьютера" (например, имя). При этом решается задача однозначной идентификации в разграничительной политике доступа пользователей, заведенных на различных компьютерах системы с одинаковыми именами (учетными записями). Настройка подобной разграничительной политики доступа в составе распределенной информационной системы в целом существенно сложнее, чем при реализации мандатного контроля доступа, но при этом имеет существенно более широкие практические возможности за счет реализаций разграничения права доступа между приложениями, используемыми в распределенной информационной системе.

Заключение

Рассмотренная технология защиты данных в информационной системе, основанная на применении методов контроля доступа, реализующих автоматическую разметку создаваемых объектов, позволяет получить принципиально новое свойство разграничительной политики доступа в распределенной информационной системе, причем с использованием различных способов обмена данными между компонентами (компьютерами) системы, а также кардинально повысить эффективность системы защиты информации за счет реализации возможности управления потоками данных в системе в целом.

Рассмотренная в работе технология защиты данных с учетом соответствующих особенностей может быть реализована для различных типов операционной системы. Иллюстрация же в работе приводится на примере реализованного и апробированного технического решения [14], в котором технология защиты данных реализована применительно к операционной системе семейства Microsoft Windows.

Список литературы

1. **Девянин П. Н.** Модели безопасности компьютерных систем. М.: Издательский центр "Академия", 2005.
2. **Цирлов В. Л.** Основы информационной безопасности автоматизированных систем. Р.: Феникс, 2008.
3. **Harrison M., Ruzzo W., Ullman J.** Protection in operating systems // Communication of the ACM. 1976. V. 19, N. 8. P. 461—471.

4. **Щеглов А. Ю.** Защита компьютерной информации от несанкционированного доступа. СПб.: Наука и техника, 2004.
5. **Bell D. E., LaPadula L. J.** Security Computer Systems: Unified Exposition and MULTICS Interpretation. Revision 1, US Air Force ESD-TR-306, MITRE Corporation MTR-2997, Bedford MA, March 1976.
6. **Щеглов К. А., Щеглов А. Ю.** Принцип и методы контроля доступа к создаваемым файловым объектам // Вестник компьютерных и информационных технологий. 2012. № 7. С. 43—47.
7. **Щеглов А. Ю., Щеглов К. А.** Система контроля доступа к файлам на основе их автоматической разметки. Патент на изобретение № 2524566. Приоритет изобретения 18.03.2013.
8. **Щеглов К. А., Щеглов А. Ю.** Реализация метода мандатного доступа к создаваемым файловым объектам // Вопросы защиты информации. 2013. Вып. 103, № 4. С. 16—20.
9. **Щеглов К. А., Щеглов А. Ю.** Практическая реализация дискреционного метода контроля доступа к создаваемым файловым объектам // Вестник компьютерных и информационных технологий. 2013. № 4. С. 43—49.
10. **Щеглов К. А., Щеглов А. Ю.** Защита от вредоносных программ методом контроля доступа к создаваемым файловым

объектам // Вестник компьютерных и информационных технологий. 2012. № 8. С. 46—51.

11. **Щеглов К. А., Щеглов А. Ю.** Защита от атак на уязвимости приложений // Информационные технологии. 2014. № 9. С. 34—39.

12. **Щеглов К. А., Щеглов А. Ю.** Принципы реализации дополнительной защиты информации при контроле доступа к создаваемым файловым объектам на основе их автоматической разметки // Вопросы защиты информации. 2014. Вып. 104, № 1. С. 29—34.

13. **Щеглов А. Ю., Щеглов К. А.** Система контроля доступа к шифруемым создаваемым файлам. Положительное решение на выдачу патента на изобретение по заявке № 2013129406/08(043781) от 26.06.2013.

14. **Щеглов А. Ю., Щеглов К. А., Павличенко И. П., Корнетов С. В.** Комплексная система защиты информации "Панцирь+" для ОС Microsoft Windows. Свидетельство о регистрации программы для ЭВМ № 2014660889 от 17.10.2014. Правообладатель ЗАО "НПП "Информационные технологии в бизнесе".

К. А. Shcheglov, Graduate Student, **A. Yu. Shcheglov**, Professor,
Saint-Petersburg National Research University of Information Technologies, Mechanics and Optics, Russia
e-mail: info@npp-itb.spb.ru

Network Informational System Data Securing Technology

We review the new network informational system data securing approach based on newly created objects (file system objects and clipboard) access control methods implementation, which allows to exclude object from access policy (with help of created objects automatic labeling). Practical implementation of such approach (while saving labeling directly in created file) allows to formulate and solve the task of implementing data (processed in network informational system) access policy in a view of different possible ways of data exchange between computers in such system. Herewith we implement data streams managing already within whole system. Reviewed protection method is based on practical realization which was patented by authors of "File objects access control system based on auto-labeling" solution. This solution allows to rethink known realization of both access control methods including discretionary and mandate ones. This not only dramatically simplifies setting file objects access policy (by eliminating the "access object" essence from access control scheme), but also settings correct implementation in the general case is provided in the same time.

Keywords: network informational system, data securing, unauthorized access, data access control and policy, access policy, newly created object, data streams, management

References

1. **Devjanin P. N.** Modeli bezopasnosti komp'yuternykh sistem. M.: Izdatel'skiy centr "Akademija", 2005.
2. **Cirlov V. L.** Osnovy informacionnoj bezopasnosti avtomatizirovannykh sistem. R.: Feniks, 2008.
3. **Harrison M. A., Ruzzo W. L. and Ullman J. D.** Protection in operating systems. *Communication of the ACM*. 1976. V. 19, N. 8. P. 461—471.
4. **Shcheglov A. Yu.** Zashhita komp'yuternoj informacii ot nesankcionirovannogo dostupa. SPb.: Nauka i tehnika, 2004.
5. **Bell D. E., LaPadula L. J.** Security Computer Systems: Unified Exposition and MULTICS Interpretation. Revision 1, US Air Force ESD-TR-306, MITRE Corporation MTR-2997, Bedford MA, March 1976.
6. **Shcheglov K. A., Shcheglov A. Yu.** Princip i metody kontrolja dostupa k sozdavaemym fajlovym ob'ektam. *Vestnik komp'yuternykh i informacionnykh tehnologij*. 2012. N. 7. P. 43—47.
7. **Shcheglov A. Yu., Shcheglov K. A.** Sistema kontrolja dostupa k fajlam na osnove ih avtomaticheskoi razmetki. Patent na izobretenie N 2524566. Prioritet izobretenja 18.03.2013.
8. **Shcheglov K. A., Shcheglov A. Yu.** Realizacija metoda mandatnogo dostupa k sozdavaemym fajlovym ob'ektam. *Voprosy zashhity informacii*. 2013. Vyp. 103. N. 4. P. 16—20.

9. **Shcheglov K. A., Shcheglov A. Yu.** Prakticheskaja realizacija diskreционного метода контроля доступа к создаваемым файловым объектам. *Vestnik komp'yuternykh i informacionnykh tehnologij*. 2013. N. 4. P. 43—49.

10. **Shcheglov K. A., Shcheglov A. Yu.** Zashhita ot vredonosnykh programm metodom kontrolja dostupa k sozdavaemym fajlovym ob'ektam. *Vestnik komp'yuternykh i informacionnykh tehnologij*. 2012. N. 8. P. 46—51.

11. **Shcheglov K. A., Shcheglov A. Yu.** Zashhita ot atak na uязvimosti prilozhenij. *Informacionnye tehnologii*. 2014. N. 9. P. 34—39.

12. **Shcheglov K. A., Shcheglov A. Yu.** Principy realizacii dopolnitel'noj zashhity informacii pri kontrole dostupa k sozdavaemym fajlovym ob'ektam na osnove ih avtomaticheskoi razmetki. *Voprosy zashhity informacii*. 2014. Vyp. 104. N. 1. P. 29—34.

13. **Shcheglov A. Yu., Shcheglov K. A.** Sistema kontrolja dostupa k shifruemym sozdavaemym fajlam. Polozhitel'noe reshenie na vydachu patenta na izobretenie po zajavke N 2013129406/08(043781) от 26.06.2013.

14. **Shcheglov A. Yu., Shcheglov K. A., Pavlichenko I. P., Kornetov S. V.** Kompleksnaja sistema zashhity informacii "Pancir+" dlja OS Microsoft Windows. Svidetel'stvo o registracii programmy dlja JeVM N 2014660889 от 17.10.2014. Pravoobladatel' ZAO "NPP "Informacionnye tehnologii v biznese".

Нечеткая когнитивная модель стратегического управления информационной безопасностью электронного правительства

Рассматриваются сущность и особенности применения нечеткого когнитивного моделирования в стратегическом управлении информационной безопасностью на уровне государства. Определены факторы стратегического управления информационной безопасностью, и на основе экспертного оценивания построена нечеткая когнитивная карта для управления информационной безопасностью электронного правительства. На основе разработанной когнитивной модели проанализированы результаты разных стратегий управления информационной безопасностью электронного правительства.

Ключевые слова: информационная безопасность, управление информационной безопасностью, стратегия информационной безопасности, когнитивное моделирование, нечеткие когнитивные карты

Введение

В результате широкого внедрения информационных и коммуникационных технологий (ИКТ) в процессы государственного управления происходят фундаментальные преобразования в природе государства. Граждане широко участвуют в формировании и реализации государственной политики, формируется эффективная система взаимодействия и сотрудничества между государством, частным сектором и гражданским обществом. Этот феномен обозначается термином "электронное правительство" [1].

Следует подчеркнуть, что термин "electronic government" (e-government) часто переводится как "электронное правительство". Такой перевод сужает понятие и сводит вопрос только к государственному управлению, осуществляемому органами исполнительной власти. Но термин "e-government" подразумевает поддержку с помощью информационных и коммуникационных технологий деятельности во всех трех ветвях власти — законодательной, исполнительной и судебной. В работе [2] обсуждаются различные подходы к определению терминов "электронное правительство" и "электронное государство" и подчеркивается, что более правильным является использование термина "электронное государство", при этом многие источники допускают употребление двух этих терминов в качестве синонимов. Следуя устоявшейся терминологии в официальных документах, в данной статье используется термин "электронное правительство".

В условиях всесторонней глобализации, возрастающих рисков и неопределенностей общественных процессов информационная безопасность (ИБ) становится одной из основных функций самосохранения электронного правительства [2, 3]. Поэтому актуальным является управление системой обеспечения ИБ электронного правительства.

Управление ИБ электронного правительства является слабоструктурированной задачей [4, 5]: объект управления является сложной социотехнической системой, состоящей из автономных компонен-

тов, каждый из которых действует целенаправленно. В системе происходят многочисленные процессы (социальные, политические, технологические), значительно взаимодействующие друг с другом. Эти процессы изменяются по времени, в них участвуют различные виды неопределенностей, но количественная информация о динамике процессов остается недоступной. Внешняя среда, окружающая электронное правительство, является потенциально "враждебной" средой. Как сами компоненты электронного правительства, так и внешняя среда являются источником многочисленных угроз, направленных на нарушение ИБ электронного правительства.

Для анализа и управления такого рода системами в настоящее время широко применяется когнитивный подход, который позволяет увидеть и осознать логику развития событий при большом числе взаимозависимых факторов [6, 7].

В этой работе на основе нечетких когнитивных карт предлагается когнитивная модель стратегического управления ИБ электронного правительства. Предлагаемый подход когнитивного моделирования перспективен в контексте создания интеллектуальных систем поддержки принятия решений, и его применение может существенно повысить эффективность стратегического управления и качество принимаемых решений в области обеспечения ИБ электронного правительства.

Нечеткие когнитивные карты

Когнитивные карты (КК) впервые были предложены американским психологом Э. Толменом (E. Tolman) при изучении элементарных когнитивных процессов у крыс [8]. На основе экспериментов по обучению крыс в разных типах лабиринтов Толмен пришел к выводу, что в процессе взаимодействия с окружающей средой у животного формируется некая "когнитивная карта", или "мысленный план", всех характеристик лабиринта, которая совершенствуется при каждом следующем взаимодействии со средой.

Когнитивные карты являются робастными системами, которые могут моделировать очень сложные поведения. В своей работе Р. Аксельрод применил когнитивные карты при изучении структуры решений политических элит [9]. Он ввел понятия взвешенные КК и функциональные КК. Во взвешенных КК знак заменен положительным или отрицательным числом, которое показывает направление эффекта, а также его значение. В функциональных КК с каждой причинной связью ассоциируется функция, которая более точно показывает направление и значение эффекта. Эти два типа когнитивных карт дают больше гибкости, поскольку они могут обработать и предоставить более подробную информацию.

Нечеткие когнитивные карты (НКК) были предложены В. Kosko [10] как нечеткое расширение когнитивных карт. На самом деле НКК являются когнитивными картами, взвешенными с нечеткими весами. Обычно КК строятся путем сбора информации от экспертов, и эксперты, скорее всего, склонны выразить себя в качественных, а не в количественных терминах. С этой точки зрения более целесообразно использовать НКК, в которых концепции представляются лингвистически, с соответствующим нечетким множеством.

НКК комбинирует некоторые аспекты нечеткой логики и нейронных сетей в схеме представления эвристики и правила здравого смысла нечеткой логики с эвристикой обучения нейронных сетей. Эта структура представляется как взвешенный ориентированный граф, в котором вершины взаимно однозначно соответствуют факторам и в терминах которых описывается предметная область, а дуги отображают взаимовлияния между факторами (рис. 1).

Вес дуги между фактором C_i и фактором C_j может быть положительным, он означает, что увеличение значения фактора C_i приводит к увеличению значения фактора C_j , в то же время уменьшение значения фактора C_i приводит к уменьшению значения фактора C_j . Если вес дуги между фактором C_i и фактором C_j отрицательный, то увеличение значения фактора C_i приводит к уменьшению значения фактора C_j , а уменьшение значения фактора C_i приводит к увеличению значения фактора C_j .

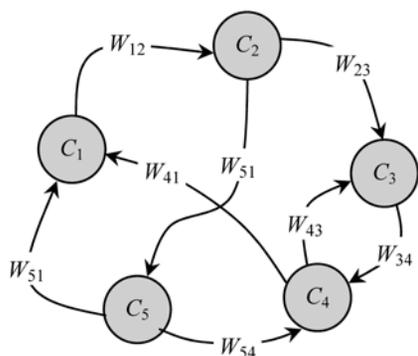


Рис. 1. Пример нечеткой когнитивной карты

В последнее десятилетие наблюдается повышенный интерес исследователей к построению нечетких когнитивных моделей во многих областях [11]. Методология НКК была успешно применена в представлении знаний [12], политических, социальных и социально-экономических исследованиях [13–15], стратегическом планировании [16] и принятии стратегических решений в нечеткой обстановке [17], бизнес-аналитике [18], экономическом прогнозировании [19], системах диспетчерского контроля [20], управлении ИТ-проектами [21], при оценке рисков качества программного обеспечения [22], в системах принятия решений в медицинской информатике [23], экологии [24, 25], интеллектуальном анализе данных [26] и т. д. Отметим, что в работе [27] когнитивная модель применена к моделированию и анализу состояния ИБ организации.

Для когнитивного моделирования управления ИБ электронного правительства необходимо:

- определить факторы, влияющие на состояние ИБ;
- построить матрицу взаимовлияний факторов;
- построить когнитивную модель управления ИБ;
- на разработанной модели отработать возможные стратегии управления ИБ электронного правительства.

Факторы управления ИБ электронного правительства

НКК является одним из способов представления знаний. Для построения НКК должны использоваться знания и опыт экспертов в предметной области [28]. Эксперты определяют те факторы, которые лучше описывают предметную область. Факторами могут быть признаки, состояния или системные переменные. Эксперты идентифицируют, какие факторы являются центральными для моделирования системы, и выявляют, какие факторы влияют друг на друга, и для соответствующих факторов определяют позитивное или негативное влияние одного фактора на другое.

Для определения факторов, влияющих на управление ИБ электронного правительства, были проанализированы национальные стратегии кибербезопасности ряда стран [29], а также модельные стратегии международных организаций [30–33]. Эти стратегии разрабатывались с широким привлечением ведущих экспертов по ИБ и могут рассматриваться как достаточно хорошие источники аккумуляции знаний экспертов. В ходе анализа стратегий кибербезопасности был выделен ряд факторов, влияющих на управление ИБ электронного правительства, полный список которых представлен в табл. 1. Рассмотрим краткое описание этих факторов.

1. Правовые меры — для обеспечения ИБ необходимо создать адекватную нормативно-правовую базу. К нормативно-правовой базе относят планирование и разработку механизмов необходимой политики и регулирования, точное определение ролей, прав и обязанностей заинтересованных сторон, базовые мероприятия и инструкции действий по

Таблица 1

Факторы, влияющие на управление ИБ электронного правительства

Факторы	Наименование фактора	Обозначение
C ₁	Правовые меры	<i>Legal</i>
C ₂	Организационные меры	<i>Org</i>
C ₃	Технические меры	<i>Tech</i>
C ₄	Развитие потенциала	<i>HR</i>
C ₅	Сотрудничество заинтересованных сторон	<i>Coop</i>
C ₆	Развитие угроз ИБ	<i>NewT</i>
C ₇	Уровень ИБ электронного правительства	<i>ISec</i>

обеспечению ИБ и т. д. Предусматриваются также разработка основных механизмов реагирования на нарушения через расследование и судебное преследование за преступления и введение санкций за несоблюдение или нарушение закона.

Нормативно-правовые меры могут быть выработаны на основе правовой базы существующих правовых институтов и структур, занимающихся кибербезопасностью и киберпреступностью. Эта группа состоит из следующих показателей:

C₁₁. Уголовное законодательство — для предупреждения киберпреступности и присоединения к международной борьбе с киберопасностью необходимо развивать соответствующую правовую базу. Законодательство по киберпреступности может быть оценено по следующим уровням: отсутствует; разработано частично; является исчерпывающим.

C₁₂. Регулирование и соответствие требованиям стандартов — регулирование ИБ обозначает законы, касающиеся защиты данных, уведомлений о нарушении и требований сертификации/стандартизации. Законы также могут быть классифицированы по уровням: отсутствует; разработано частично; является исчерпывающим.

2. Организационные меры — в стратегиях кибербезопасности предусматривается построение гибкой организационной структуры управления, направленной на обеспечение ИБ. Создание эффективных организационных структур необходимо для продвижения ИБ, борьбы с киберпреступностью и повышения роли мониторинга, предупреждения и реагирования на инциденты для обеспечения межведомственной, кросс-секториальной и трансграничной координации между новыми и существующими инициативами. Организационные меры можно оценить на основе существования и числа учреждений и стратегий, организующих развитие ИБ на национальном уровне. Подгруппа состоит из следующих показателей:

C₂₁. Политика — официально признанные национальные или по конкретным секторам стратегии ИБ.

C₂₂. Дорожная карта для управления — официально признанные национальные или по конкретным секторам планы управления для ИБ.

C₂₃. Ответственный орган — официально признанные национальные или по конкретным секторам агентства по ИБ.

C₂₄. Национальный бенчмаркинг — официально признанные национальные или по конкретным секторам упражнения бенчмаркинга, используемые для измерения уровня ИБ.

3. Технические меры — технология является первой линией обороны против киберугроз и вредоносных Интернет-агентов. Без адекватных технических мер и потенциала для выявления и реагирования на кибератаки электронное правительство и его субъекты остаются уязвимыми для киберугроз. Поэтому электронное правительство должно быть способно развивать стратегии по установлению принятых минимальных критериев безопасности и схем аккредитации для программного обеспечения и информационных систем.

C₃₁. Система раннего предупреждения — предусматриваются повышение готовности к инцидентам, уменьшение времени реагирования, разработка плана восстановления после аварий и механизмов защиты критической информационной инфраструктуры (например, национальный план действий в особых условиях, правило поведения в киберпространстве, информирование о ситуации).

C₃₂. Стандарты — этот показатель определяет существование уполномоченной правительством структур(ы) для реализации международно признанных стандартов по ИБ в государственном секторе и в критических инфраструктурах.

C₃₃. Сертификация — этот показатель определяет существование утвержденной правительством структур(ы) для сертификации и аккредитации государственных учреждений и специалистов государственного сектора по международно признанным стандартам в области ИБ.

4. Развитие потенциала — развитие человеческого и институционального потенциала существенно для первых трех факторов (правовые, технические и организационные). Понимание технологий, рисков и последствий может помочь в разработке более совершенного законодательства, более эффективных политик и стратегий, а также для лучшей организации различных ролей и обязанностей.

Подгруппа состоит из следующих показателей:

C₄₁. Подготовка кадров — указывается на необходимость новых образовательных программ, уделяющих внимание на образование IT-специалистов и профессионалов по кибербезопасности. В некоторых национальных стратегиях кибербезопасности ставится цель усовершенствования образовательных программ специалистов по кибербезопасности для надежного обеспечения кибербезопасности, а также сертификации специалистов по ИБ.

C₄₂. Осведомленность населения — программы осведомления, предусматривающие обучение пользователей новым моделям поведения и работы в киберпространстве.

С₄₃. Научные исследования и инновации — необходимо проведение комплексных научно-практических исследований, направленных на решение проблем безопасности и устойчивости как существующих, так и будущих систем и сервисов. В ряде стратегий предусматриваются определение ведущих центров в области исследований по кибербезопасности и обеспечение инвестициями.

С₄₄. Разработка стандартов — любые официально признанные национальные или по конкретным секторам программы/проекты исследования и разработка стандартов по ИБ, лучших практик и правил для применения в государственном и частном секторах.

С₄₅. Сертификация государственных органов — этот показатель можно измерить по числу государственных учреждений, сертифицированных в соответствии с международно признанными стандартами.

5. Сотрудничество заинтересованных сторон — для управления ИБ требуется участие всех заинтересованных сторон, поэтому государственные структуры и частный сектор должны работать в тесном сотрудничестве. Международное сотрудничество является жизненно важным, поскольку все зависит от одного киберпространства. Сотрудничество должно реализовываться путем обмена информацией и передовым опытом, знаниями на различных уровнях.

Национальное и международное сотрудничество может быть измерено на основе существования числа партнерств, совместных структур и сетей обмена информацией. Подгруппа состоит из следующих показателей:

С₅₁. Внутригосударственное сотрудничество — официально признанное национальное или по конкретным секторам партнерство для трансграничного совместного использования активов ИБ с другими государствами.

С₅₂. Межведомственное сотрудничество — любые официально признанные национальные или по конкретным секторам программы для обмена активами ИБ (люди, процессы, инструменты) в государственном секторе.

С₅₃. Партнерство государственного и частного секторов — официально признанные национальные или по конкретным секторам программы для обмена активами ИБ между государственным и частным секторами.

С₅₄. Международное сотрудничество — международное сотрудничество может охватывать законодательные меры, реагирование на инциденты, научные исследования, сертификацию аппаратного и программного обеспечения.

6. Развитие угроз ИБ — в этой работе рассматриваются следующие показатели:

С₆₁. Развитие акторов угроз — динамика изменений в акторах угроз (инсайдеры, активисты/хактивисты, криминалы, стратегические конкуренты, враждебные государства).

С₆₂. Появление новых типов атак — разработка и осуществление хорошо скоординированных, целенаправленных атак [34], усовершенствование существующих методов атак, многошаговые, многовекторные атаки, атаки нулевого дня (0-дня), динамические, полиморфные вредные программы и т. д.

С₆₃. Развитие целей атак — развитие новых информационных технологий (критические инфраструктуры, мобильные, "облачные" вычисления, Интернет вещей и т. д.) и электронные услуги.

7. Уровень ИБ электронного правительства — интегральная оценка уровня ИБ электронного правительства, определяется на основе основных показателей рисков ИБ электронного правительства. Рассматриваются следующие три уровня ИБ в зависимости от соответствующего уровня рисков [35]:

- высокий уровень ИБ — соответствует низкому уровню рисков;
- удовлетворительный уровень ИБ — соответствует приемлемому уровню рисков;
- низкий уровень ИБ — соответствует высокому уровню рисков.

Построение матрицы взаимовлияний факторов

При построении НКК наиболее сложной задачей является назначение весов взаимовлияний факторов. В работе [20] приводятся два алгоритма вычисления матрицы взаимовлияний факторов.

В первом алгоритме каждый эксперт оценивает веса взаимовлияний как число из интервала $[-1, 1]$. Далее эти матрицы весов взаимовлияний агрегируются как осредненное значение суммы весов или применяется пороговая функция (например, сигмоидная функция). Так как опыт и знания экспертов об объекте оценки могут быть разными, то каждому эксперту можно назначить неотрицательный числовой вес доверия. С учетом весов доверия экспертов агрегированные значения весов взаимовлияний могут быть вычислены следующей формулой (учитываются только веса одинакового знака):

$$W_{ij} = \frac{\sum_{k=1}^m b_k w_{ij}^k}{m}, \quad (1)$$

где w_{ij}^k — оценка веса взаимовлияния между C_i и C_j k -м экспертом; b_k — вес доверия k -го эксперта; m — число экспертов. Если оценка эксперта отличается от оценок большинства экспертов, то он штрафуются — ему присваивают очень низкий или нулевой вес доверия. Для более детального описания с описанием этого алгоритма читателям рекомендуется работа [20].

Второй алгоритм для построения матрицы взаимовлияний факторов НКК использует нечеткую логику. Эксперты описывают каузальность между факторами с помощью лингвистических переменных. Каждый эксперт определяет влияние одного фактора на другой фактор как "негативное" или "позитивное" и после этого описывает степень влияния

с помощью лингвистических переменных типа "сильный", "слабый" и т. д. [36]. Преимуществом этой методологии является то, что экспертам не нужно присваивать числовые веса каузальным связям, они описывают степень каузальных связей между факторами привычными терминами.

Следуя этой методологии, влияние одного фактора на другой можно интерпретировать как лингвистическая переменная, которая принимает значения в универсальном множестве $[-1, 1]$. Ее множество термов может быть следующим [11]:

$T(\text{влияние}) = \{\text{негативно очень сильное, негативно сильное, негативно среднее, негативно слабое, негативно нулевое, позитивно слабое, позитивно среднее, позитивно сильное, позитивно очень сильное}\}$.

Ниже определяется семантическое правило, и эти термы характеризуются нечеткими множествами, функции принадлежности которых показаны на рис. 2.

- $T(\text{негативно очень сильное}) =$ нечеткое множество для "влияние ниже -75% " с функцией принадлежности μ_{nvs} ;
- $T(\text{негативно сильное}) =$ нечеткое множество для "влияние близко к -75% " с функцией принадлежности μ_{ns} ;
- $T(\text{негативно среднее}) =$ нечеткое множество для "влияние близко к -50% " с функцией принадлежности μ_{nm} ;
- $T(\text{негативно слабое}) =$ нечеткое множество для "влияние близко к -25% " с функцией принадлежности μ_{nw} ;
- $T(\text{негативно нулевое}) =$ нечеткое множество для "влияние близко к 0 " с функцией принадлежности μ_z ;

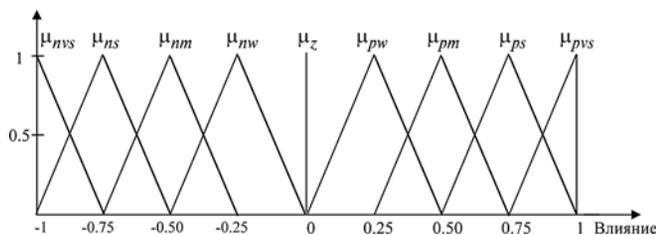


Рис. 2. Термы лингвистической переменной "Влияние"

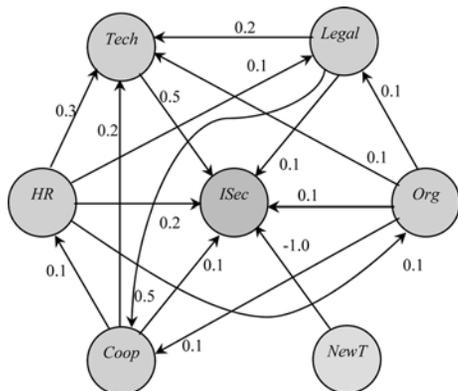


Рис. 3. НКК модель управления ИБ "электронного государства" в виде графа

- $T(\text{позитивно слабое}) =$ нечеткое множество для "влияние близко к 25% " с функцией принадлежности μ_{pw} ;
- $T(\text{позитивно среднее}) =$ нечеткое множество для "влияние близко к 50% " с функцией принадлежности μ_{pm} ;
- $T(\text{позитивно сильное}) =$ нечеткое множество для "влияние близко к 75% " с функцией принадлежности μ_{ps} ;
- $T(\text{позитивно очень сильное}) =$ нечеткое множество для "влияние выше 75% " с функцией принадлежности μ_{pvs} .

Лингвистические переменные, которые описывают все взаимодействия факторов, агрегируются, и общая лингвистическая переменная преобразуется в интервал $[-1, 1]$ с помощью дефаззификации. В этой работе использовался метод центра тяжести для дефаззификации [37].

НКК имеют те же основные недостатки, что и другие нечеткие системы: они не в состоянии обучаться самостоятельно. При доступности соответствующих данных веса взаимовлияний факторов можно улучшить, используя механизмы обучения нейронных сетей. Большинство таких подходов основано на методе обучения Хебба (см., например, [38, 39]), но существуют также подходы с использованием эволюционных вычислений [40].

Для вычисления матрицы взаимовлияний факторов в модели НКК для стратегического управления электронным правительством были привлечены пять экспертов по управлению ИБ. Эксперты оценивали влияние факторов друг на друга в вышеопределенных лингвистических переменных, и полученный результат после агрегации и дефаззификации приведен на рис. 3.

Моделирование динамики НКК

Процессы вывода НКК включают вектор состояний $A_{1 \times n}$, который состоит из n значений факторов, и весовую матрицу $W_{n \times n}$, которая отражает веса w_{ij} взаимовлияния между n -факторами. На значение каждого фактора оказывают влияние значения связанных с ним факторов и их предыдущее значение. Значение активации для каждого фактора вычисляется итеративно следующим правилом:

$$A_i^{(t+1)} = f\left(\sum_{j=1}^n w_{ij} A_j^{(t)}\right), \quad i \neq j, \quad (2)$$

где t — текущее время; A — уровень активации фактора C_i ; A_j — уровень активации фактора C_j ; w_{ij} — вес взаимовлияния между C_i и C_j ; f — пороговая функция.

В качестве пороговых функций использовались бинарные, тривалентные и сигмоидные функции [41]. В этой работе в качестве пороговой функции для НКК используется сигмоидная функция

$$f(x) = \frac{1}{1 + e^{-\lambda x}}, \quad (3)$$

где $\lambda > 0$. Эта функция непрерывна, и ее областью значений является отрезок $[0, 1]$.

Мы предполагаем, что состояния факторов могут быть определены как нечеткие переменные, состоящие из трех нечетких множеств: высокое (high), среднее (medium) и низкое (low).

Результаты вычислительных экспериментов

Отметим, что каждая из концепций C_j может принимать значения в интервале $[0, 1]$, который также называется "уровнем активации". Уровень активации можно интерпретировать как относительное число [25]. Более строго уровень активации может представлять членство в нечетком множестве, описывающем лингвистические меры относительной численности (например, низкий, средний, высокий) [10].

Процесс моделирования НКК инициализируется присвоением значения из интервала $[0, 1]$ уровням активации каждого узла НКК на основе мнений специалистов/заинтересованных сторон для текущего состояния. Значение 0 говорит о том, что данный фактор не присутствует в системе в определенной итерации, в то время как значение 1 указывает, что данный фактор присутствует в максимальной степени. Другие значения соответствуют промежуточным уровням активации.

Рассмотрим моделирование следующих сценариев управления ИБ.

Сценарий А: саморазвитие ситуации $A(0) = (1.0, 1.0, 1.0, 1.0, 1.0, 0.0)$.

Сценарий В: использование только технических мер $A(0) = (0.0, 0.0, 1.0, 0.0, 0.0, 0.0)$.

Сценарий С: сильная активация фактора "развитие угроз ИБ" $A(0) = (0.0, 0.0, 0.0, 0.0, 0.0, 1.0)$.

Сценарий D: сильная активация факторов "Развитие потенциала" и "Развитие угроз ИБ" $A(0) = (0.0, 0.0, 0.0, 1.0, 0.0, 1.0)$.

В вычислительных экспериментах мы использовали сигмоидную функцию с параметром $\lambda = 1$. Как правило, вычисления по формуле (2) сходились менее чем за пять временных шагов моделирования. Все модели закончились в стабильном состоянии, но теоретически они могли бы также перейти в предельный цикл или в хаотический аттрактор [6]. Промежуточные значения факторов при вычислениях по сценарию А приведены в табл. 2.

Как видно из табл. 2, сохранение существующих тенденций факторов приведет к ухудшению уровня информационной безопасности.

Из табл. 3 можно заключить, что использование только технических мер не дает существенного улучшения состояния ИБ (разность между стабильными состояниями сценариев А и В составляет 0,0424).

Табл. 4 показывает, что при активации фактора "Развитие угроз ИБ" состояние ИБ ухудшается значительно, также можно отметить ухудшение начальной тенденции фактора "Технические меры".

Табл. 5 показывает, что при развитии угроз ИБ не удастся обеспечить приемлемый уровень ИБ

Таблица 2

Вычисление стабильного состояния для НКК (сценарий А)

Факторы итерации	Legal	Org	Tech	HR	Coop	NewT	ISec
0	1,00	1,00	1,00	1,00	1,00	1,00	1,00
1	0,5498	0,5250	0,6457	0,5987	0,5498	0,5000	0,5987
2	0,5281	0,5150	0,5871	0,5619	0,5292	0,5000	0,5721
3	0,5269	0,5140	0,5821	0,5570	0,5275	0,5000	0,5633
4	0,5268	0,5139	0,5816	0,5566	0,5274	0,0000	0,5624
5	0,5267	0,5139	0,5816	0,5566	0,5274	0,5000	0,5624
6	0,5267	0,5139	0,5816	0,5566	0,5274	0,5000	0,5624

Таблица 3

Конечные результаты вычислений по сценарию В

Факторы	Начальные значения — сценарий В	Конечные значения — сценарий В	Разность между стабильными состояниями сценариев А и В
Правовые меры	0,00	0,5275	0,0008
Организационные меры	0,00	0,5147	0,0008
Технические меры	1,00	1,00	0,4184
Развитие потенциала	0,00	0,5875	0,0309
Сотрудничество заинтересованных сторон	0,00	0,5378	0,0104
Развитие угроз ИБ	0,00	0,5000	0
Уровень ИБ	0,00	0,6048	0,0424

Таблица 4

Конечные результаты вычислений по сценарию С

Факторы	Начальные значения — сценарий С	Конечные значения — сценарий С	Разность между стабильными состояниями сценариев А и С
Правовые меры	0,00	0,5267	0,00
Организационные меры	0,00	0,5139	0,00
Технические меры	0,00	0,5569	-0,0247
Развитие потенциала	0,00	0,5547	-0,0019
Сотрудничество заинтересованных сторон	0,00	0,5267	-0,0007
Развитие угроз ИБ	1,00	1,00	0,5
Уровень ИБ	0,00	0,4976	-0,0648

Таблица 5

Конечные результаты вычислений по сценарию D

Факторы	Начальные значения — сценарий D	Конечные значения — сценарий D	Разность между стабильными состояниями сценариев А и D
Правовые меры	0,00	0,5381	0,0114
Организационные меры	0,00	0,5250	0,0111
Технические меры	0,00	0,5904	0,0088
Развитие потенциала	0,00	1,0000	0,4434
Сотрудничество заинтересованных сторон	0,00	0,5279	0,0005
Развитие угроз ИБ	1,00	1,00	0,5
Уровень ИБ	0,00	0,5238	-0,0386

только за счет развития потенциала; наряду с развитием соответствующего потенциала требуется найти оптимальное сочетание правовых, технических, организационных мер по информационной безопасности при эффективном сотрудничестве всех заинтересованных сторон.

Заключение

В работе представлен метод построения когнитивной модели для стратегического управления информационной безопасностью электронного правительства. Выделены основные управляющие факторы, и на основе математического аппарата нечетких когнитивных карт построена когнитивная карта, отображающая взаимовлияние факторов. Определены начальные тенденции изменения факторов и проанализированы различные стратегические сценарии развития системы управления информационной безопасностью электронного правительства. Предлагаемое когнитивное моделирование позволяет при первом приближении оценить степень достижимости поставленных стратегических целей по ИБ.

Список литературы

1. Grönlund A., Horan T. A. Introducing e-gov: history, definitions, and issues // Communications of the Association for Information Systems. 2004. V. 15. P. 713–729.
2. Имамвердиев Я. Н. Модель ситуационного управления информационной безопасностью электронного правительства // Информационные технологии. 2014. № 8. С. 24–32.
3. Wimmer M., von Bredow B. E-government: aspects of security on different layers // Proc. of the 12th International Workshop on Database and Expert Systems Applications. 2001. P. 350–355.
4. Chen Y.-S., Chong P. P., Zhang B. Cyber security management and e-government // Electronic Government. 2004. V. 1, N. 3. P. 316–327.
5. Алгулиев Р. М., Имамвердиев Я. Н. Информационная безопасность э-государства: актуальные направления исследований // Проблемы информационного общества. 2010. № 1. С. 3–13.
6. Glykas M. Fuzzy Cognitive Maps: Advances in Theory, Methodologies, Tools and Applications. Studies in Fuzziness and Soft Computing. Springer, 2010. V. 247.
7. Максимов В. И., Корноушенко Е. К. Аналитические основы применения когнитивного подхода при решении слабоструктурированных задач // Труды ИПУ РАН, 199. Т. 2. С. 95–109.
8. Tolman E. Cognitive maps in rats and men // Psychological Review. 1948. N. 55. P. 189–208.
9. Axelrod R. Structure of decision: The cognitive maps of political elites. New Jersey: Princetown University Press, 1976.
10. Kosko B. Fuzzy cognitive maps // International Journal of Man-Machine Studies. 1986. V. 24, N 1. P. 65–75.
11. Papageorgiou E. I. Review study of fuzzy cognitive maps and their applications during the last decade // Studies in Computational Intelligence: Business Process Management. 2013. V. 444. P. 281–298.
12. Taber W. R. Knowledge processing with fuzzy cognitive maps // Expert Systems with Applications. 1991. V. 2, N. 1. P. 83–87.
13. Andreou A. S., Mateou N. H., Zombanakis G. A. Soft computing for crisis management and political decision making: the use of genetically evolved fuzzy cognitive maps // Soft Computing Journal. 2005. V. 9, N. 3. P. 194–210.
14. Carvalho J. P. On the semantics and the use of Fuzzy Cognitive Maps in social sciences // Proc. of the IEEE World Congress on Computational Intelligence (WCCI). 2010. N 5584033.
15. Максимов В. И. Структурно-целевой анализ развития социально-экономических ситуаций // Проблемы управления. 2005. № 3. С. 30–38.
16. Tsadiras A., Margaritis K., Mertzios B. Strategic planning using extended Fuzzy Cognitive Maps // Studies in Informatics and Control. 1995. V. 4, N. 3. P. 237–345.
17. Силов В. Б. Принятие стратегических решений в нечеткой обстановке. М.: ИНПРО-РЕС, 1995. 228 с.
18. Xirogiannis G., Glykas M. Fuzzy Cognitive Maps in business analysis and performance-driven change // IEEE Transactions on Engineering Management. 2004. V. 51, N. 3. P. 334–351.
19. Song H., Miao C., Rael W., Shen Z., Cattthoor F. Implementation of fuzzy cognitive maps based on fuzzy neural network and application in prediction of time series // IEEE Transactions on Fuzzy Systems. 2010. V. 18, N. 2. P. 233–250.
20. Stylios C. D., Groumpos P. P. Fuzzy cognitive map in modeling supervisory control systems // Journal of Intelligent & Fuzzy Systems. Application in Engineering and Technology. 2000. V. 8, N. 2. P. 83–98.
21. Rodriguez-Repiso L., Setchi R., Salmeron J. L. Modelling IT Projects success with Fuzzy Cognitive Maps // Expert Systems with Applications. 2007. V. 32, N. 2. P. 543–559.
22. Bhatia N., Kapoor N. Fuzzy Cognitive Map based approach for software quality risk analysis // ACM SIGSOFT Software Engineering Notes. 2011. V. 36, N. 6. P. 1–9.
23. Papageorgion E. I. A new methodology for decisions in Medical Informatics using Fuzzy Cognitive Maps based on fuzzy rule-extraction techniques // Applied Soft Computing. 2011. V. 11, N. 1. P. 500–513.
24. Özsesmi U., Özsesmi S. L. Ecological models based on people's knowledge: a multi-step fuzzy cognitive mapping approach // Ecological Modelling. 2004. V. 176, N. 1–2. P. 43–64.
25. Hobbs B. F., Ludsin S. A., Knight R. L., Ryan P. A., Biberhofer J., Ciborowski J. J. H. Fuzzy cognitive mapping as a tool to define management objectives for complex ecosystems // Ecological Applications. 2002. V. 12. P. 1548–1565.
26. Hong T., Han I. Knowledge-based data mining of news information on the Internet using cognitive maps and neural networks // Expert Systems with Applications. 2002. V. 23, N. 1. P. 1–8.
27. Камаев В. А., Нартов В. В. Моделирование и анализ состояния информационной безопасности организации // Известия ТулГУ. Технические науки. 2011. № 3. С. 148–155.
28. Schneider M., Shnaider E., Kandel A., Chew G. Automatic construction of FCMs // Fuzzy Sets and Systems. 1998. V. 93, N. 2. P. 161–172.
29. Luijff H., Besseling K., Spoelstra M., de Graaf P. Ten National Cyber Security Strategies: a comparison // Proc. 6th International Conference on Critical Information Infrastructures Security (CRITIS 2011). September 2011.
30. ENISA: National Cyber Security Strategies: Setting the course for national efforts to strengthen security in cyberspace. May 2012. 15 p.
31. The ITU National Cybersecurity Strategy Guide. Geneva, 2012. 122 p.
32. OECD: Non-governmental Perspectives on a New Generation of National Cybersecurity Strategies. OECD Digital Economy Papers. OECD Publishing. 2012. N. 212.
33. Klimburg A. (Ed.) National Cyber Security Framework Manual, Tallinn: NATO CCD COE Publication, 2012. 235 p.
34. Sood A. K., Enbody R. J. Targeted Cyberattacks: A Superset of Advanced Persistent Threats // IEEE Security & Privacy. 2013. V. 11, N. 1. P. 54–61.
35. Brothy W. K. Information security management metrics; A definitive guide to effective security monitoring and measurement. CRC Press. 2012. 200 p.
36. Kosko B. Neural networks and fuzzy systems. Englewood Cliffs. N. J.: Prentice-Hall, 1992.
37. Runkler T. A. Selection of appropriate defuzzification methods using application specific properties // IEEE Transactions on Fuzzy Systems. 1997. V. 5, N. 1. P. 72–79.
38. Papakostas G. A., Polydoros A. S., Koulouriotis D. E., Tournassis V. D. Training Fuzzy Cognitive Maps by using Hebbian learning algorithms: A comparative study // IEEE International Conference on Fuzzy Systems (FUZZ). 2011. P. 851–858.
39. Papageorgiou E. I., Stylios C. D., Groumpos P. P. Active Hebbian learning algorithm to train fuzzy cognitive maps // International Journal of Approximate Reasoning. 2001. V. 37, N. 3. P. 219–249.
40. Papageorgiou E. I., Parsopoulos K. E., Stylios C. D., Groumpos P. P., Vrahatis M. N. Fuzzy Cognitive Maps learning using Particle Swarm Optimization // International Journal of Intelligent Information Systems. 2005. V. 25, N. 1. P. 95–121.
41. Tsadiras A. K. Comparing the inference capabilities of binary, trivalent and sigmoid fuzzy cognitive maps // Information Sciences. 2008. V. 178, N. 20. P. 3880–3894.

A Fuzzy Cognitive Model for the Strategic Management of Information Security of E-Government

The article studies the nature and application of cognitive modeling in the strategic management of information security of e-government. Factors of strategic management of information security are defined and the fuzzy cognitive map for strategic management of e-government information security is built on the basis of expert assessments. Based on the developed cognitive model results of different strategies for e-government information security management are analyzed.

Keywords: information security, information security management, information security strategy, cognitive modeling, fuzzy cognitive maps

References

1. Grönlund A., Horan T. A. Introducing e-gov: history, definitions, and issues. *Communications of the Association for Information Systems*. 2004. V. 15. P. 713–729.
2. Imamverdiyev Ya. N. Model' situacionnogo upravleniya informacionnoj bezopasnost'ju je-gosudarstva. *Informacionnye tehnologii*. 2014. N. 8. P. 24–32.
3. Wimmer M., von Bredow B. E-government: aspects of security on different layers. *Proc. of the 12th International Workshop on Database and Expert Systems Applications*. 2001. P. 350–355.
4. Chen Y.-S., Chong P. P., Zhang B. Gyber security management and e-government. *Electronic Government*. 2004. V. 1, N. 3. P. 316–327.
5. Alguliyev R. M., Imamverdiyev Yu. N. Informacionnaya bezopasnost' e-gosudarstva: aktual'nye napravleniya issledovaniy. *Problemy informacionnogo obshestva*. 2010. N. 1. P. 3–13.
6. Glykas M. *Fuzzy Cognitive Maps: Advances in Theory, Methodologies, Tools and Applications*. *Studies in Fuzziness and Soft Computing*. Springer. 2010. V. 247.
7. Maksimov V. I., Kornoushenko E. K. Analiticheskie osnovy primeneniya kognitivnogo podhoda pri reshenii slabostrukturirovannyh zadach. *Trudy IPU RAN*. 1999. V. 2. P. 95–109.
8. Tolman E. Cognitive maps in rats and men. *Psychological Review*. 1948. N. 55. P. 189–208.
9. Axelrod R. *Structure of decision: The cognitive maps of political elites*. New Jersey: Princetown University Press, 1976.
10. Kosko B. Fuzzy cognitive maps. *International Journal of Man-Machine Studies*. 1986. V. 24, N. 1. P. 65–75.
11. Papageorgiou E. I. Review study of fuzzy cognitive maps and their applications during the last decade. *Studies in Computational Intelligence: Business process Management*. 2013. V. 444. P. 281–298.
12. Taber W. R. Knowledge processing with fuzzy cognitive maps. *Expert Systems with Applications*. 1991. V. 2, N. 1. P. 83–87.
13. Andreou A. S., Mateou N. H., Zombanakis G. A. Soft computing for crisis management and political decision making: the use of genetically evolved fuzzy cognitive maps. *Soft Computing Journal*. 2005. V. 9, N. 3. P. 194–210.
14. Carvalho J. P. On the semantics and the use of Fuzzy Cognitive Maps in social sciences. *Proc. of the IEEE World Congress on Computational Intelligence (WCCI)*. 2010. N. 5584033.
15. Maksimov V. I. Strukturno-celevoj analiz razvitiya social'no-ekonomicheskikh situacij. *Problemy upravleniya*. 2005. N. 3. P. 30–38.
16. Tsadiras A., Margaritis K., Mertziotis B. Strategic planning using extended Fuzzy Cognitive Maps. *Studies in Informatics and Control*. 1995. V. 4, N. 3. P. 237–345.
17. Silov V. B. *Prinjatije strategicheskikh reshenij v nechetkoj obstanovke*. M.: INPRO-RES, 1995. 228 p.
18. Xirogiannis G., Glykas M. Fuzzy Cognitive Maps in business analysis and performance-driven change. *IEEE Transactions on Engineering Management*. 2004. V. 51, N. 3. P. 334–351.
19. Song H., Miao C., Rael W., Shen Z., Cathoor F. Implementation of fuzzy cognitive maps based on fuzzy neural network and application in prediction of time series. *IEEE Transactions on Fuzzy Systems*. 2010. V. 18, N. 2. P. 233–250.
20. Stylios C. D., Groumpos P. P. Fuzzy cognitive map in modeling supervisory control systems. *Journal of Intelligent & Fuzzy Systems. Application in Engineering and Technology*. 2000. V. 8, N. 2. P. 83–98.
21. Rodriguez-Repiso L., Setchi R., Salmeron J. L. Modelling IT Projects success with Fuzzy Cognitive Maps. *Expert Systems with Applications*. 2007. V. 32, N. 2. P. 543–559.
22. Bhatia N., Kapoor N. Fuzzy Cognitive Map based approach for software quality risk analysis. *ACM SIGSOFT Software Engineering Notes*. 2011. V. 36, N. 6. P. 1–9.
23. Papageorgiou E. I. A new methodology for decisions in Medical Informatics using Fuzzy Cognitive Maps based on fuzzy rule-extraction techniques. *Applied Soft Computing*. 2011. V. 11, N. 1. P. 500–513.
24. Özesmi U., Özesmi S. L. Ecological models based on people's knowledge: a multi-step fuzzy cognitive mapping approach. *Ecological Modelling*. 2004. V. 176, N. 1–2. P. 43–64.
25. Hobbs B. F., Ludsin S. A., Knight R. L., Ryan P. A., Biberhofer J., Ciborowski J. J. H. Fuzzy cognitive mapping as a tool to define management objectives for complex ecosystems. *Ecological Applications*. 2002. V. 12. P. 1548–1565.
26. Hong T., Han I. Knowledge-based data mining of news information on the Internet using cognitive maps and neural networks. *Expert Systems with Applications*. 2002. V. 23, N. 1. P. 1–8.
27. Kamaev V. A., Natrov V. V. Modelirovanie i analiz sostojaniya informacionnoj bezopasnosti organizacii. *Izvestija TulGU. Tehnicheskie nauki*. 2011. N. 3. P. 148–155.
28. Schneider M., Shneider E., Kandel A., Chew G. Automatic construction of FCMs. *Fuzzy Sets and Systems*. 1998. V. 93, N. 2. P. 161–172.
29. Luijff H., Besseling K., Spoelstra M., de Graaf P. Ten National Cyber Security Strategies: a comparison. *Proc. 6th International Conference on Critical Information Infrastructures Security (CRITIS 2011)*. September 2011.
30. ENISA: *National Cyber Security Strategies: Setting the course for national efforts to strengthen security in cyberspace*. May 2012. 15 p.
31. *The ITU National Cybersecurity Strategy Guide*. Geneva. 2012. 122 p.
32. OECD: *Non-governmental Perspectives on a New Generation of National Cybersecurity Strategies*. *OECD Digital Economy Papers*. OECD Publishing. 2012. N. 212.
33. Klimburg A. (Ed.) *National Cyber Security Framework Manual*, Tallinn: NATO CCD COE Publication, 2012. 235 p.
34. Sood A. K., Enbody R. J. Targeted Cyberattacks: A Superset of Advanced Persistent Threats. *IEEE Security & Privacy*. 2013. V. 11, N. 1. P. 54–61.
35. Brothly W. K. *Information security management metrics; A definitive guide to effective security monitoring and measurement*. CRC Press. 2012. 200 p.
36. Kosko B. *Neural networks and fuzzy systems*. Englewood Cliffs. N. J.: Prentice-Hall, 1992.
37. Runkler T. A. Selection of appropriate defuzzification methods using application specific properties. *IEEE Transactions on Fuzzy Systems*. 1997. V. 5, N. 1. P. 72–79.
38. Papakostas G. A., Polydoros A. S., Koulouriotis D. E., Tourassis V. D. Training Fuzzy Cognitive Maps by using Hebbian learning algorithms: A comparative study. *IEEE International Conference on Fuzzy Systems (FUZZ)*. 2011. P. 851–858.
39. Papageorgiou E. I., Stylios C. D., Groumpos P. P. Active Hebbian learning algorithm to train fuzzy cognitive maps. *International Journal of Approximate Reasoning*. 2001. V. 37, N. 3. P. 219–249.
40. Papageorgiou E. I., Parsopoulos K. E., Stylios C. D., Groumpos P. P., Vrahatis M. N. Fuzzy Cognitive Maps learning using Particle Swarm Optimization. *International Journal of Intelligent Information Systems*. 2005. V. 25, N. 1. P. 95–121.
41. Tsadiras A. K. Comparing the inference capabilities of binary, trivalent and sigmoid fuzzy cognitive maps. *Information Sciences*. 2008. V. 178, N. 20. P. 3880–3894.

ПРОГРАММНАЯ ИНЖЕНЕРИЯ SOFTWARE ENGINEERING

УДК 004.02

С. И. Сметанин, аспирант, **В. А. Игнатюк**, д-р физ.-мат. наук, проф., e-mail: victor.ignatyuk@vvsu.ru
А. А. Евстифеев, аспирант

Владивостокский государственный университет экономики и сервиса, г. Владивосток

Способ реализации программной веб-части системы спутникового мониторинга

Предлагается подход к созданию программной веб-части системы спутникового мониторинга с использованием БД PostgreSQL, Javascript и C#. Обосновывается необходимость разработки собственного решения в условиях различной картографической информации и информационных данных. Предлагаемый концептуальный подход к созданию системы мониторинга, опирающийся на веб-технологии, позволит существенно упростить разработку подобных систем, направленных на решение задач любого возможного типа.

Ключевые слова: система спутникового мониторинга, картографический движок, структура базы данных, JavaScript и C#, сервер, клиентское приложение

Введение

На данный момент существует множество предложений по продаже мобильных устройств, предназначенных для контроля движущихся объектов или трекеров. В большинстве из них есть функция передачи информации по GPRS на любой заданный веб-адрес через определенный интервал времени.

Системы спутникового мониторинга должны решать следующие задачи:

- 1) выбор карты и ее отображение;
- 2) отображение информации об объектах мониторинга (скорость, угол поворота, расход топлива);
- 3) отображение полигонов, линий, точек;
- 4) отображение информации, связанной с полигонами, линиями, точками (всплывающие подсказки);
- 5) математические действия (подсчет пройденного пути, площади полигона, принадлежности точки полигону).

Большинство систем мониторинга используют карты Google Maps и OpenStreetMap, но поставленные перед системой задачи могут быть различными. Иногда требуется использовать картографическую информацию из собственных источников, и готовые фреймворки, рассчитанные на перечисленные выше сервисы, приходится перерабатывать под требования разрабатываемой системы. Это не всегда возможно и актуально в условиях поставленной задачи.

В данной работе представлена структура авторской системы мониторинга, по образцу которой можно создать универсальную систему, работаю-

щую с данными различных типов и категорий. Она основана на синтезе трех составляющих:

- 1) базы данных PostgreSQL;
- 2) ASP.NET (Active Server Pages) — технологии создания веб-приложений и веб-сервисов от компании Майкрософт;
- 3) собственного фреймворка, реализованного на языке JavaScript.

В работе описаны основные части системы и способы их реализации, а также возможные проблемы, с которыми разработчик может столкнуться при создании собственной системы спутникового мониторинга, и способы их решения.

Картографическая информация и учет ее в базе данных

Система спутникового мониторинга должна основываться на данных векторных карт, чтобы использовать и хранить информацию для интерактивного взаимодействия с географическими объектами: зданиями и помещениями, изображенными в качестве полигонов; дорогами, показанными как линии; отметками карты (например, отметки видеокамер, телефонов, магазинов), представленные точками.

При разработке данной системы мониторинга была использована БД PostgreSQL. На специфические типы данных этой БД приведены подробные объяснения.

Поскольку у любой карты есть различные уровни масштабирования, и, следовательно, различные объекты, отображаемые на разных уровнях, поэтому необходимо создать либо отдельные таблицы для

Label Тип: text	Надпись, отображаемая на карте.
Type Тип: int	Тип объекта. 0, 1, 2 (полигоны, полилинии, точки).
Polygons Тип: polygon	Коллекция точек (x,y) в специальном географическом формате PostgreSQL.
TypeDescription Тип: text	Описание самого типа объекта (дом, завод, озеро, помещение, лаборатория и т.д.)
Square Тип: double	Площадь объекта.

Рис. 1. Список столбцов таблицы MapObjects

каждого из уровней, либо общую таблицу и представления (View), разделяемые по параметру K — коэффициенту масштабирования.

Структура созданных таблиц, каждая из которых имеет идентификатор MapObjects ($K = 1...n$), представлена на рис. 1.

Все объекты необходимо хранить в виде полигонов, поскольку это облегчает дальнейшие математические преобразования и действия с ними. Если в формате карт есть такие объекты, как линии или точки, то их следует преобразовать в полигоны по следующему алгоритму.

Если объект является точкой, то вокруг него описывается прямоугольник, где n — ширина точки:

```
PointF[] points = new PointF[] {
    new PointF(point.X - n, point.Y - n),
    new PointF(point.X + n, point.Y - n),
    new PointF(point.X + n, point.Y + n),
    new PointF(point.X - n, point.Y + n),
    new PointF(point.X - n, point.Y - n)
};
```

Если объект является линией, то прямоугольник образуется с помощью последовательного обхода всей линии точками — от начала к концу, и от конца к началу. При этом значение координаты X , по сравнению с базовой точкой линии не изменяется, а координата Y смещается на $\pm n$. Кроме того, необходимо учитывать угол поворота от текущей точки к следующей точке в массиве, чтобы полностью повторить кривую линии новой фигурой — полигоном.

```
for (int i = 0; i < points.Length; i++)
{ //двигаемся от начала к концу
    PointF p1 = new PointF(points[i].X, points[i].Y - n);
    if (i != points.Length - 1) // если это не последняя точка,
    то найти угол между этой и следующей точкой в массиве
        angle = findAngle(points[i], points[i + 1]);
    p1 = rotatePoint(points[i].X, points[i].Y, p1, angle); // повернуть
    созданную точку на заданный угол и добавить ее в новый массив точек
    полигона
    p.Add(p1);
}
for (int i = points.Length - 1; i != -1; i--)
{ //двигаемся от конца к началу. Действия повторяются.
```

```
PointF p1 = new PointF(points[i].X, points[i].Y + n);
if (i != 0)
    angle = findAngle(points[i - 1], points[i]);
p1 = rotatePoint(points[i].X, points[i].Y, p1, angle);
p.Add(p1);
}
PointF p2 = new PointF(points[0].X, points[0].Y - n);
p2 = rotatePoint(points[0].X, points[0].Y, p2, angle);
p.Add(p2);
```

Функция findAngle основана на нахождении угла между двумя векторами: вектора, образованного точками, переданными в функцию, и единичного вектора с координатами (1, 0). Функция rotatePoint вращает точку создаваемого полигона вокруг первоначальной точки линии с учетом вычисленного угла. Она выглядит следующим образом:

```
private static float findAngle(PointF p1, PointF p2)
{
    System.Windows.Vector vector1 = new System.Windows.Vector(p2.X - p1.X, p2.Y - p1.Y);
    System.Windows.Vector vector2 = new System.Windows.Vector(1, 0);
    float labelAngle = (float)System.Windows.Vector.AngleBetween(vector2, vector1);
    if (labelAngle >= 180/2)
        labelAngle -= 180;
    if (labelAngle < -180/2)
        labelAngle += 180;
    return labelAngle;
}
```

Впоследствии вхождение курсора пользователя в выбранный полигон может быть вычислено запросом к базе данных PostgreSQL. Важно отметить, что данная БД поддерживает функции работы с географическими типами [1], поэтому ее использование предпочтительно при построении собственной системы мониторинга.

Стандартный запрос такого вида представлен ниже, где point — параметр с географическим типом точки, передаваемый при выполнении хранимой процедуры:

```
SELECT Label, Type, TypeDescription,
Polygons FROM MapObjectsK =
= 256 WHERE (select :point @ Polygons);
```

Специальный символ @ задает функцию Contains, при этом проверяется, входит ли указанная точка в границы полигона. Это может использоваться для различных способов взаимодействия с объектами: их передвижения, изменения различных характеристик или вызова динамической подсказки. Иногда возникает необходимость также обозначить область, доступную для взаимодействия с объектами карты. Это может быть как вся карта целиком, так и некоторый отдельный ее участок, чтобы не вызывать обработчиков клика пользователя при выходе за границы рабочей области карты.

В разработанной системе мониторинга такая таблица называется MapLimits и включает столбцы xMin, xMax, yMin, yMax, K. Таким образом, при обработке события клика мыши, в первую очередь проверяется, входят ли координаты курсора в установленную область карты, после чего запускается то событие, которое и должно было быть применено изначально.

Рисование карт и взаимодействие с ними

При создании собственных карт разработчик может использовать различные форматы векторной графики. В разработанной системе мониторинга, в качестве карт использовался так называемый "польский формат карт" — свободно распространяемый формат векторной графики, предоставляющий широкие возможности для прорисовки местности [2].

Несмотря на использование векторного формата, ввиду вопросов, связанных с чрезмерной нагрузкой на сервер, был применен "тайловый подход". Тайлинг — метод создания больших изображений из маленьких фрагментов (тайлов) одинаковых размеров, подобно картине из мозаики.

Вся карта разбивается на строки и столбцы. Допустим, целиком карта занимает 1024 пикселя по ширине, и столько же по высоте. Если принять размер тайла равным 256 пикселей, то карту можно условно разбить на 4 строки, каждая из которых содержит 4 столбца. Пример такого подхода демонстрируется на рис. 2.

Каждый из тайлов заранее отрисовывается и сохраняется в папке на сервере, таким образом, по запросу пользователя достаточно только передать уже существующий тайл, а не рисовать все изображение заново.

Координаты карты также могут быть представлены в разных датумах: например, в датуме WGS-84. Следовательно, для того чтобы прорисовать тайл, необходимо сопоставить его координаты (широту и долготу в случае WGS-84) с пикселями карты. Это делается с помощью аффинного преобразования [3].

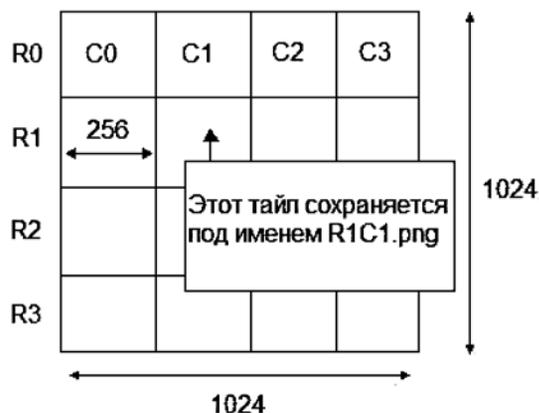


Рис. 2. Разметка карты на тайлы

Взаимосвязь пикселей экрана и географических координат можно представить в виде системы уравнений с неизвестными коэффициентами:

$$x' = a_0 + a_1x + a_2y + a_3x^2 + a_4xy + a_5y^2; \quad (1)$$

$$y' = b_0 + b_1x + b_2y + b_3x^2 + b_4xy + b_5y^2, \quad (2)$$

где x, y — координаты в исходной системе координат; x', y' — координаты в конечной системе координат (пиксели); a_0-a_5, b_0-b_5 — коэффициенты (неизвестны).

Точная привязка проводится по шести точкам, таким образом, требуется шесть пар таких уравнений для нахождения соответствующих коэффициентов. Данные системы уравнений можно решить с использованием уже готовых библиотек. Например, в рассматриваемой работе это реализовано с помощью библиотеки ALGLIB, которая реализует метод LU-декомпозиции на языке C#. Она доступна для скачивания по приведенной ссылке [4].

После предварительного создания карты и ее разбиения на тайлы необходимо средство для вывода этих изображений на экран и взаимодействия с ними. Для этих целей автором статьи был разработан собственный картографический движок, созданный на основе JavaScript, без использования каких-либо дополнительных библиотек или фреймворков. В процессе разработки был использован только плагин jQuery для облегчения работы с объектной моделью документа (DOM).

Отрисовка тайлов происходит с использованием HTML 5 Canvas. Реализация движка — модульная, и состоит из двух основных частей: lottaMap.Мар и lottaMap.Objects. Неполное ее описание приведено в работе [5].

Для начала работы необходимо в *html*-разметке указать элемент `<canvas id="canvas"></canvas>`.

Внутренняя механика

После инициализации объекта необходимо считать и сформировать тайлы, полностью покрывающие выбранный элемент `canvas`. Для этого необходимо знать его ширину и высоту, после чего определяем число тайлов. К ним добавляем тайлы буфера — дополнительные, хранящиеся в памяти изображения за пределами видимой области карты, необходимые для быстрого отображения карты при ее перемещении пользователем.

К объекту `lottaMap.Мар` относится подмножество других объектов, каждый из которых отвечает за те или иные действия пользователя. К перемещению карты относится объект `draggableMap` — он проверяет, передвинул ли пользователь карту на определенное расстояние, после чего меняет значения глобальных переменных `mapX` и `mapY`, отвечающих за координаты верхнего левого угла карты. После этого запускается функция отрисовки карты по заданным координатам.

```

dragMap = function () {
    // перетащить карту
    var dX = startPoint.x - endPoint.x,
        dY = startPoint.y - endPoint.y;
    mapX -= dX;
    mapY -= dY;
    tiles.tilesMX += dX; // смещение тайлов по x. y
    tiles.tilesMY += dY;
    ctx.translate(dX, dY); // контекст основного канваса
    if (useObjects)
        ctxBackground.translate(dX, dY) // канвас зад-
ного фона, для взаимодействия с объектами
    lottamap.Map.drawMap(); // вызывается рисование
карты
    tiles.unvisibleTilesCollector(); // запускается "сборщик
мусора"
};

```

За различные действия с тайлами, а именно за их загрузку, формирование и удаление лишних тайлов, отвечает объект `tiles`. Поскольку все изображения содержатся в массиве, определенном в данном объекте, то от тех тайлов, которые не попадают в область видимости карты, необходимо избавиться, чтобы они не расходовали дополнительную память. Это делается с помощью "сборщика мусора". Он рассчитывает границы текущей области, попадающей в пределы элемента `canvas`. Получившийся таким образом прямоугольник проверяется на пересечение с прямоугольником каждого элемента из массива тайлов `tilesStore` — если пересечения не происходит, то данный тайл необходимо удалить.

```

unvisibleTilesCollector: function () {
    var collectorLimit = 4; // количество тайлов за грани-
цей видимой карты, которые будут учитываться сбор-
щиком
    // сборщик мусора (тайлов, вышедших за границы
зоны просмотра)
    var borderRightWidth = mapX + canvas.width + collec-
torLimit * tileSize, // правая координата X
        borderRightHeight = mapY + canvas.height + collec-
torLimit * tileSize, // правая нижняя координата Y
        borderLeftWidth = mapX - collectorLimit * tileSize, // ле-
вая координата X
        borderLeftHeight = mapY - collectorLimit * tileSize, //
левая нижняя координата Y
        tile, // переменная, используемая для получения
названия тайлов из их массива
        obj1 = [borderLeftWidth, borderLeftHeight, border-
RightWidth, borderRightHeight],
        obj2 = [],
        xStart, xEnd, yStart, yEnd;
    for (tile in tilesStore) { // tilesStore – массив тайлов
        if (tilesStore.hasOwnProperty(tile)) {
            xStart = tilesStore[tile].x; // координаты прямо-
угольника самого тайла
            xEnd = tilesStore[tile].x + tileSize;
            yStart = tilesStore[tile].y;
            yEnd = tilesStore[tile].y + tileSize;
            obj2 = [xStart, yStart, xEnd, yEnd];

```

```

        if (!(((obj2[0] <= obj1[0] && obj1[0] <= obj2[2])
            || (obj1[0] <= obj2[0] && obj2[0] <= obj1[2]))
            && ((obj2[1] <= obj1[1] && obj1[1] <= obj2[3])
            || (obj1[1] <= obj2[1] && obj2[1] <= obj1[3]))) {
            delete tilesStore[tile]; // если тайл не попада-
ет в проверяемую область, удалить его
        }
    }
}

```

Масштабирование

После того как срабатывает событие "mouse-wheel" мыши, выполняются следующие основные действия.

Сначала вычисляется новый уровень карты — это происходит за счет переменной `delta` из события прокрутки мыши. В зависимости от того, в какую сторону крутилось колесо мыши, ее значение будет либо положительным, либо отрицательным. Использовать ее очень удобно, поскольку достаточно просто отнять от текущей переменной уровня значение `delta`.

Затем происходит вычисление коэффициента масштабирования, как отношения между значениями старого и нового уровней. Это используется для вычисления "шагов" — определения, сколько раз в цикле нужно проводить операцию увеличения или отдаления тайла для создания плавного визуального эффекта.

```

if (levels.scales[current] > levels.scales[oldL]) // если но-
вый коэффициент масштабирования больше старого,
то рассчитываем разницу между ними как новый уро-
вень / старый уровень, иначе наоборот
    diff = levels.scales[currentL] / levels.scales[oldL];
else
    diff = levels.scales[oldL] / levels.scales[currentL];
steps = Math.abs(Math.round(Math.log(diff) /
Math.log(zoom))); // вычисляем количество шагов. Zoom
= 2, если колесо крутят вверх, 0,5 — если вниз.

```

До того как начать операцию изменения масштаба, старые тайлы копируются в буфер.

```

tilesCopyStore = [];
for (var tile in tilesStore) {
    tilesCopyStore[tile] = tilesStore[tile];
}

```

После чего средствами `canvas` происходит увеличение или отдаление тайла из копии, сделанной на предыдущем шаге. Для этого вызывается функция масштабирования тайла, причем вызывается то число раз, которое было определено переменной `steps`, через последовательно инкрементируемые задержки времени `i * 80`.

```

for (var i = 0; i < steps; i++) {
    setTimeout(function () {
        tiles.scaleTiles(zoom, x, y);
    }, i * 80);
}

```

Известны координаты места в пикселях, над которым было прокручено колесо мыши. Эти координаты переводятся в координаты карты, после чего умножаются на коэффициент `zoom`. Необходимо также учитывать смещение тайлов при масштабировании `canvas` — за эти значения отвечают переменные `this.tilesMX` и `this.tilesMY`. Кроме того, на данном шаге задается значение для масштабирования скопированных тайлов текущего уровня `scaleT`, также использующихся для создания необходимого визуального эффекта:

```
clearTimeout(timeout); // масштабирование происходит в несколько шагов, по таймеру
ScaleT *= zoom;
center.oldCenter.x = x - this.tilesMX;
center.oldCenter.y = y - this.tilesMY;
center.newCenter.x = Math.round(center.oldCenter.x * zoom);
center.newCenter.y = Math.round(center.oldCenter.y * zoom);
```

Значение точки прокрутки колесика фиксируется, после чего сравнивается со значением этой точки с учетом ее масштабирования. Карта смещается на разницу между ними:

```
ctx.translate(-Math.round(center.newCenter.x - center.oldCenter.x), -Math.round(center.newCenter.y - center.oldCenter.y)); // смещение карты
if (useObjects) // смещение буфера на ту же величину, если используются объекты (подробнее ниже)
  ctxBackground.translate(-Math.round(center.newCenter.x - center.oldCenter.x), -Math.round(center.newCenter.y - center.oldCenter.y));
this.tilesMX += -Math.round(center.newCenter.x - center.oldCenter.x); // вычисления нового значения переменной для смещения тайлов
this.tilesMY += -Math.round(center.newCenter.y - center.oldCenter.y);
mapX -= -Math.round(center.newCenter.x - center.oldCenter.x); // те же самые действия необходимо осуществить для глобальных переменных mapX и mapY, показывающих координаты верхнего левого угла экрана
mapY -= -Math.round(center.newCenter.y - center.oldCenter.y);
drawCopy();
```

Функция `drawCopy` вызывает прорисовку в цикле снятой копии тайлов, помноженной на коэффициент масштабирования:

```
ctx.drawImage(thisTile.img, Math.ceil(thisTile.x * scaleT), Math.ceil(thisTile.y * scaleT), Math.ceil(tileSize * scaleT), Math.ceil(tileSize * scaleT));
```

После этого запускается таймер для начала рисования тайлов нового уровня. Задержка в рисовании установлена на 90 мс, и если все шаги, использующиеся для плавного увеличения копии, еще не выполнены, то это событие отменяется, как видно из кода выше. Таким образом, до того как весь процесс визуального масштабирования тайлов не

завершится, событие таймера по рисованию новых тайлов не будет задействовано:

```
timeout = setTimeout(function () {
  tiles.clearTiles();
  lottaMap.Map.drawMap(true);
}, 90)
```

Взаимодействие с объектами карты

Объект `lottaMap.Objects` отвечает за взаимодействие с различными структурами на карте, начиная от подсказок касательно зданий, дорог и прочих векторных объектов, до управления непосредственно данными трека.

Поскольку `canvas` не поддерживает векторные данные, определение объекта, с которым взаимодействует пользователь, должно быть реализовано самостоятельно.

Один подход заключается в использовании математических формул, например формулы окружности, для того чтобы вызывать событие клика мышью по точке как результат проверки вхождения координат мыши в окружность. Но достаточно сложные фигуры, такие как многоугольники различной формы, тяжело описать подобным образом.

Подход, который использовали авторы статьи, связан с применением второго, буферного `canvas`'а, который остается скрытым для пользователя и хранится лишь в памяти браузера. Каждый объект отрицывается на нем своим собственным, уникальным цветом, а ключ `RGB`, отвечающий за данный цвет, хранится в массиве. Таким образом, общее число возможных ключей $255 \times 255 \times 255 = 16\,777\,216$, чего более чем достаточно в большинстве случаев.

Поскольку сам фон прозрачен, считывание значений цветности пикселя в месте клика мыши укажет, какой объект использовался на данный момент.

Необходимо создать элемент `canvas` и получить с него контекст:

```
var canvasBackground = document.createElement('canvas'), // канвас для проверки клика по точкам трекам
ctxBackground = canvasBackground.getContext('2d');
```

Важно отметить, что над этим элементом также необходимо проводить операции передвижения и масштабирования для того чтобы синхронизировать видимый и фоновый объекты `canvas` между собой.

Объекты, с которыми будет проводиться взаимодействие, необходимо предварительно нарисовать на фоновом `canvas`'е. Если это фигура, заданная определенными координатами, то сделать это не представляет сложности. Проблема возникает, когда необходимо использовать готовые изображения, например значки транспортных средств. В этом случае необходимо проанализировать каждый пиксель изображения и закрасить непрозрачные пиксели выбранным цветом, соответствующим индивидуальному ключу объекта. Это можно сделать с помощью метода `putImageData`, принадлежащего контексту элемента `canvas`, что накладывает следую-

шее ограничение: для каждого нового изображения необходимо создавать свой собственный фоновый canvas. Тем не менее, его не требуется хранить постоянно — после отрисовки изображения его можно сохранить в файл или память браузера с помощью метода canvas'a — `toDataURL("image/png")`. Метод контекста canvas'a, `getImageData` позволяет получить по соответствующим координатам x и y прямоугольник, ширину и высоту которого необходимо установить равными единице. Этот объект содержит информацию о цветности пикселей экрана (0 — красный, 1 — зеленый, 2 — голубой, 3 — альфа-канал). Проверка альфа-канала позволяет установить, содержится ли по выбранным координатам пиксель закрашиваемого изображения, в этом случае его значение будет больше нуля.

Чтобы создать новый пиксель вместо предыдущего с необходимым и установленным цветом, требуется использовать метод `createImageData`. Так же, как и в методе `getImageData`, ширину и высоту области данных изображения следует установить равными единице. Объект, созданный этим методом, содержит в себе массив `data`, в элементы которого `[0]...[3]` необходимо занести значения цвета RGB-ключа, установив значение альфа-канала в состоянии "непрозрачный" — оно должно быть равно 255. Метод `putImageData` помещает данный объект в фоновый canvas, созданный на первом шаге.

Код, приведенный ниже, демонстрирует попиксельный анализ изображения и закрашивание непрозрачных пикселей RGB-цветом индивидуального ключа объекта:

```
var imageCanvas = document.createElement('canvas'),
// создается фоновый canvas
imageCtx = imageCanvas.getContext('2D'); // его контекст
imageCanvas.width = img.width; // размеры canvas'a
устанавливаются равными размерам изображения
imageCanvas.height = img.height;
imageCtx.drawImage(img, 0, 0); // отрисовка изображения
на временном фоновом canvas'e
for (var h = 0; h < img.height; h++) { // перебор пикселей
в цикле
  for (var w = 0; w < img.width; w++) {
    if (imageCtx.getImageData(w, h, 1, 1).data[3]
    > 0) { // проверка альфа-канала
      var imageData = imageCtx.createImageData(1, 1);
      var d = imageData.data;
      d[0] = this.rgb[0];
      d[1] = this.rgb[1];
      d[2] = this.rgb[2];
      d[3] = 255;
      imageCtx.putImageData(imageData, w, h);
    }
  }
}
this.backgroundImage = new Image();
this.backgroundImage.src = imageCanvas.toDataURL(
"image/png"); // преобразование canvas'a в объект
Image
```

Аналогичным образом происходит проверка RGB-ключа при клике мышью по видимому canvas'у. По этим координатам проверяется значение пиксельных данных фонового элемента canvas, после чего найденный RGB-ключ преобразуется в строку и ищется в массиве ключей всех объектов.

```
var isOpaque = pixel[3] === 255; // если пиксель не-
прозрачен, то читаем его параметры цветности
if (isOpaque) {
  var r = pixel[0],
      g = pixel[1],
      b = pixel[2],
      id = r.toString() + g.toString() + b.toString(); //
формируем ключ
  return keys[id];
}
```

Таким образом, возможно реализовать полноценное взаимодействие с объектами без использования векторной графики.

Необходимо подчеркнуть, что все основные действия с интерфейсом пользователя происходят с использованием JavaScript. Сервер занимается только получением необходимых данных из БД и выполняет такие второстепенные задачи, как перекрашивание изображения транспортного средства и сохранение его в папке на сервере.

Тем не менее, чтобы сервер мог осуществить запрос к базе данных, необходимо уведомить его о данной задаче. Наиболее простой способ организовать такое взаимодействие — это AJAX (от англ. *Asynchronous Javascript and XML* — асинхронный JavaScript и XML). Во фреймворке jQuery уже есть объекты для использования AJAX [6], при его применении реализовать процесс взаимодействия с сервером не представляет сложности:

```
$.ajax({
  type: "POST", // POST или GET сообщение
  url: "AjaxData.aspx", // страница, которая примет со-
общение
  data: str, // данные
  success: function (msg) { // функция, которая вызыва-
ется при условном приеме ответных данных с сервера
  }
});
```

Передача данных от сервера к клиенту на примере динамических изображений

В разработанном авторами картографическом движке использованы различные типы объектов — как географические объекты векторной карты, так и объекты, использующиеся для навигации по треку (тип транспортного средства, начало—конец трека, метка "остановки" объекта мониторинга).

В то время как объекты первого типа уже отрисованы на тайлах карты и занесены в базу данных, объектам второго типа требуются готовые изображения, которые необходимо предварительно передать

ObjectType Тип: text	Тип объекта. Автомобиль, грузовой автомобиль, пешеход и т.д.
ImagePath Тип: text	Путь к изображению в «не активированном» виде – при отсутствии наведения на него курсора мыши пользователя
ActiveImagePath Тип: text	Путь к изображению в «активированном» виде
DefaultColor Тип: text	Основной цвет изображения по умолчанию, который будет использоваться при поиске пикселей для перекрашивания

Рис. 3. Список столбцов таблицы ObjectImages



Рис. 4. Изображение для таблицы ObjectImages

клиентскому приложению системы мониторинга — в данном случае, браузеру пользователя.

Рассмотрим следующий пример: пользователю доступна возможность изменения цвета трека наблюдаемого им подвижного объекта. Но будет более наглядно, если и сам трек, и изображение, отвечающее за данный объект, можно перекрашивать в зависимости от выбранного пользователем цвета.

Подобные типы объектов хранятся в таблице ObjectImages. Объекты, изображения которых не изменяются со временем, хранятся в таблице MarkImages.

Структура таблицы ObjectImages продемонстрирована на рис. 3.

Таблица MarkImages отличается отсутствием в ней столбца DefaultColor.

Поскольку изображения таблицы ObjectImages являются динамически изменяемыми, сами изображения также должны создаваться в ходе работы приложения.

Изображения для данных объектов подбираются таким образом, чтобы они имели максимально однородный цвет. За счет этого упрощается процесс поиска пикселей нужного цвета для алгоритма перекрашивания базового изображения.

Пример такого изображения продемонстрирован на рис. 4.

При смене цвета трека, меняется и цвет значка, отображающего текущий объект:

```

bitm = new System.Drawing.Bitmap(serverPath + defaultImagePath); // получаем базовое изображение, которое будет перекрашено
for (int y = 0; y < bitm.Height; y++)
{
    for (int x = 0; x < bitm.Width; x++)
    { // последовательно обходим изображение по пикселям
        Color rgb = bitm.GetPixel(x, y);
        if (rgb.A > 0 && rgb.R == defaultColor.R && rgb.G == defaultColor.G && rgb.B == defaultColor.B) // проверяем пиксели на соответствие стандартному цвету изображения, которые требуется закрасить новым цветом
        {
            bitm.SetPixel(x, y, newColor); // перекрашиваем пиксель
        }
    }
}

```

Новое изображение сохраняется по указанному пути с генерированным временным именем. При следующей смене цвета трека это изображение должно быть удалено и заменено новым.

Для каждого пользователя генерируются таблицы userInfo (рис. 5) и userData.

Колонки LastRepaintingType, LastImageName, LastAImageName и LastRepaintingColor из таблицы userInfo (рис. 5), применяются для условия проверки изменения изображения. Само условие выглядит следующим образом:

```

bool needToPaint = (LastRepaintingType == String.Empty || LastRepaintingColor == String.Empty);
needToPaint = needToPaint || ((LastRepaintingColor != String.Empty ? ColorTranslator.FromHtml("#" + 1 LastRepaintingColor).Name : true) || TransportType != LastRepaintingType);

```

Таким образом, если значения LastRepaintingType и LastRepaintingColor пусты — это значит, что изображение загружается в первый раз, и должно быть перекрашено под установленный пользователем цвет. Если же эти значения уже заполнены, то проверяется, равен ли последний сохраненный цвет LastRepaintingColor — новому цвету Color (если они равны, то необходимость в перерисовке изображения отсутствует), после чего проверяется соответствие столбцов TransportType и LastRepaintingType, поскольку тип объекта мог измениться за прошедшее время. В этом случае его изображение также необходимо изменить.

Данные мониторинга хранятся в таблице userData. Она включает в себя такие столбцы, как:

1) Point, тип point — точки (x, y), в координатах карты (пикселях экрана);

Name Тип: text	Имя объекта мониторинга	LastImageName Тип: text	Имя последнего использованного изображения объекта
Color Тип: text	Цвет трека (и иконки объекта)	LastAImageName Тип: text	Имя последнего активного изображения объекта
TransportType Тип: text	Текущий тип объекта выбранного объекта (может меняться) – автомобиль, грузовой транспорт и т.д.	LastRepaintingColor Тип: text	Последний сохраненный цвет объекта
LastRepaintingType Тип: text	Последний используемый тип объекта.		

Рис. 5. Структура таблицы userInfo

2) GPSpoint, тип point — эти же точки, но в географических координатах (широта и долгота);

3) Date, тип timestamp — дата и время получения данных;

4) Name — имя объекта мониторинга.

Остальные параметры вторичны и используются для информационных отчетов. В зависимости от поставленных задач, в данную таблицу могут быть добавлены дополнительные столбцы, например, для данных скорости, уровня топлива или высоты.

Заключение

В данной статье авторами была рассмотрена реализация следующих основных частей системы спутникового мониторинга.

1. Предпочтительная структура базы данных.

2. Взаимодействие с векторными объектами на карте.

3. Реализация картографического движка на языке JavaScript.

4. Взаимодействие с сервером для получения необходимой информации по треку.

Результатом проделанной работы является программа для системы спутникового мониторинга, способная использовать карты польского формата, адекватно отображать объекты мониторинга и демонстрировать графики различных параметров за указанный промежуток времени — уровень топлива, скорость движения, показания различных датчиков. Хотя в подобных системах часто используют готовые

решения, но они оказываются совершенно бесполезными, когда ситуация требует использовать свои собственные картографические данные или применять различные параметры, которые не предусмотрены в разработанной системе мониторинга.

Информация, указанная в данной статье, послужит для применения в разработке систем мониторинга различной направленности: мониторинга транспорта, физических лиц, событий (пример — уведомление о возгорании в помещении). Приведенные алгоритмы позволяют быстро создать свою собственную систему, не требуя использовать готовые фреймворки или приложения в рамках поставленной задачи.

Список литературы

1. Уорсли Дж., Дрейк Дж. PostgreSQL. Для профессионалов. СПб.: Питер, 2003. 613 с.

2. cGPSmapper User Manual. URL: <http://cgpsmapper.com/download/cGPSmapper-UsrMan-v02.5.pdf> (дата обращения 9.08.2014).

3. Полиномиальные преобразования — математика. URL: <http://gis-lab.info/qa/polynomial-calc.html> (дата обращения 9.08.2014).

4. ALGLIB Free Edition. URL: <http://www.alglib.net/download.php> (дата обращения 9.08.2014).

5. Сметанин С. И., Савченко И. С. Разработка системы мониторинга, использующей веб-интерфейс на базе GPS/ГЛОНАСС // Материал XIV международной научно-практической конференции "Интеллектуальный потенциал вузов — на развитие Дальневосточного региона России и стран АТР". Владивосток, 2012. С. 82—85.

6. Бибо Б., Кац И. jQuery. Подробное руководство по продвинутому JavaScript. М.: Символ-Плюс, 2009. 376 с.

S. I. Smetanin, Graduate Student, V. A. Ignatyuk, Dr. Sci Sciences, prof., e-mail; victor.ignatyuk@vvsu.ru,
A. A. Evstifeev, Graduate Student
Department of Electronics, Vladivostok State University of Economics and Service, Vladivostok

Implementation of the Software Part of the System of Satellite Monitoring

The text shows an approach to the creation of the software part of the satellite monitoring system. Also, it talks about the necessity to develop their own solutions to a wider variety of use cartographic information and other data information. The proposed conceptual approach to the creation of a monitoring system, which is based on web technology, will greatly simplify the development of such systems, aimed at addressing the problems of any possible type. In particular, realization of the main functions which any cartographical cursor — scalings and movements of the card has to realize is given in this article. Also the example of comparison of pixels of the card to geographical coordinates of the objects which are stored in a database that allows to connect them in program functions is given. Code examples are given. The code is given in the C# language as server part and JavaScript as client part.

Keywords: satellite monitoring system, engine mapping, database design, JavaScript, C#, default server, the client application

References

1. Worsley J. C., Drake J. D. *Practical PostgreSQL*. O'Reilly. Sebastopol, CA, 2002. 613 p.

2. cGPSmapper User Manual. URL: <http://cgpsmapper.com/download/cGPSmapper-UsrMan-v02.5.pdf>. 2014.

3. *Polynomial'nye preobrazovaniya — matematika*. URL: <http://gis-lab.info/qa/polynomial-calc.html> (accessed 9.08.2014).

4. ALGLIB Free Edition. URL: <http://www.alglib.net/download.php> (accessed 9.08.2014).

5. Smetanin S. I., Savchenko I. S. Razrabotka sistemy monitoringa, ispol'zuyushchei veb-interfeis na baze GPS/GLONASS. *Materialy XIV mezhdunarodnoj nauchno-prakticheskoi konferentsii: "Intellektual'nyi potentsial vuzov — na razvitie Dal'nevostochnogo regiona Rossii i stran ATR"*. Vladivostok, 2012. P. 82—85 (in Russian).

6. Bibault B., Katz Y. *jQuery in Action*, Manning. Greenwich, CT. 2008. 376 p.

ЦИФРОВАЯ ОБРАБОТКА СИГНАЛОВ И ИЗОБРАЖЕНИЙ DIGITAL PROCESSING OF SIGNALS AND IMAGES

УДК 004.032.24

В. Г. Гетманов¹, д-р техн. наук, гл. науч. сотр., e-mail: v.getmanov@gcras.ru,

Г. И. Борзунов^{2, 3}, д-р техн. наук, проф.,

¹ФГБУН Геофизический центр РАН, г. Москва

²ФГБОУ ВПО Национальный исследовательский ядерный университет "МИФИ", г. Москва

³ФГБОУ ВПО Московский государственный университет дизайна и технологий, г. Москва

Алгоритм параллельных вычислений для задачи спектрально-временного анализа на базисных полигармонических функциях

Предложен алгоритм параллельных вычислений для задачи спектрально-временного анализа (СВАН) на базисных полигармонических функциях (БПФН). Рассмотрены алгоритм оценивания параметров БПФН, метод задания векторов поисковых частот и последовательный алгоритм СВАН. Оценены временная и емкостная сложности последовательного алгоритма СВАН. Сформирована структура алгоритма параллельных вычислений для СВАН. Приведены результаты вычислительного эксперимента с модельной реализацией параллельного алгоритма для СВАН сигнала с амплитудной и частотной модуляцией.

Ключевые слова: алгоритм параллельных вычислений, спектрально-временной анализ, базисные полигармонические функции, аппроксимация

Введение

Алгоритмы параллельных вычислений используются в задачах, которые требуют для своего решения больших затрат машинного времени и объемов памяти. Как правило, чтобы применение параллельных вычислений оказалось возможным, исходные задачи должны быть определенным образом преобразованы. Рассматриваемая задача спектрально-временного анализа на базисных полигармонических функциях решается на основе предложенного в работе алгоритма параллельных вычислений.

Реализация традиционного спектрального анализа для наблюдений на локальных (малых) временных интервалах может основываться на их аппроксимации базисными полигармоническими функциями (БПФН). Решения задач спектрально-временного анализа (СВАН) для нестационарных сигналов, наблюдаемых на исходных больших временных интервалах, в ряде случаев можно получить с помощью решений последовательностей задач традиционного спектрального анализа на системах локальных временных интервалов, выделенных на исходных интервалах.

Предлагаемый в данной работе СВАН на БПФН с оптимизацией их амплитудных и частотных параметров [1–3] имеет ряд преимуществ, которые отсутствуют у вариантов обычных СВАН. Так, для

предлагаемого СВАН по сравнению со СВАН, основанным на скользящих дискретных преобразованиях Фурье (ДПФ) [4, 5], достигается улучшение разрешающей способности по частоте благодаря оптимизации частотных параметров для БПФН (для ДПФ оптимизация частотных параметров отсутствует). Кроме того, для предлагаемого СВАН отсутствуют ложные спектрально-временные оценки, тогда как для СВАН, в котором применяются частотно-временные распределения, например Вигнера [6, 7], имеет место в некоторых случаях их появление. По сравнению со СВАН, в котором используется алгоритм Прони [8], предлагаемый СВАН не накладывает ограничений, связанных с числом наблюдений и числом частотных составляющих в БПФН, что устраняет определенные трудности, возникающие при интерпретации спектрально-временных оценок. Отметим, что БПФН без оптимизации [9–11] или с частичной оптимизацией частотных параметров [12, 13] применяются достаточно широко для различных задач типа СВАН. Ввиду изложенного выше исследование возможностей СВАН с использованием указанных БПФН в задачах анализа многочастотных нестационарных сигналов является актуальным.

Данную статью можно подразделить на две части. Первая часть посвящена собственно предложенному

СВАН — разработке алгоритма оценивания параметров БПФН, разработке метода задания векторов поисковых частот и формированию последовательного алгоритма СВАН. Вторая часть посвящена разработке алгоритма параллельных вычислений для предложенного СВАН — оцениванию временной и емкостной сложности последовательного алгоритма СВАН, определению структуры алгоритма параллельных вычислений для СВАН и реализации вычислительного эксперимента с модельным параллельным алгоритмом СВАН.

1. Алгоритм оценивания параметров базисных полигармонических функций

Рассмотрим задачу оценивания параметров базисных полигармонических функций для локального интервала с точками $i = 0, 1, \dots, I_0 - 1$. Примем, что исходный сигнал является многочастотным и нестационарным. Будем полагать, что БПФН на локальном интервале для такого сигнала может быть представлена в виде модели

$$y_M(c, Ti) = \sum_{l=1}^L (a_l \cos \omega_l Ti + b_l \sin \omega_l Ti), \quad (1)$$

где вектор $c^T = (a^T, b^T, \omega^T) = (a_1, a_2, \dots, a_L, b_1, b_2, \dots, b_L, \omega_1, \omega_2, \dots, \omega_L)$ имеет размерность $(3L, 1)$ и состоит из векторов амплитудных и частотных параметров; Ti — текущее время. На линейные амплитудные параметры БПФН ограничения не накладываются: $-\infty \leq a_l \leq \infty, -\infty \leq b_l \leq \infty, l = 1, 2, \dots, L$. Примем, что исходный сигнал реализован в ограниченной полосе частот с $\omega_{\min}, \omega_{\max}$; для векторов нелинейных частотных параметров БПФН с учетом их упорядочения, допустимое множество значений параметров выглядит следующим образом:

$$\bar{\Omega}_0^L = \{(\omega_1, \dots, \omega_L) : \omega_{\min} \leq \omega_1 < \omega_2 < \dots < \omega_L \leq \omega_{\max}\}, \quad \omega \in \bar{\Omega}_0^L. \quad (2)$$

Задача оценивания параметров базисных полигармонических функций связана с аппроксимацией наблюдений $Y(Ti)$ исходного сигнала модельной функцией $y_M(c, Ti)$ (1) и нахождением оптимальных амплитудных и частотных параметров. С этой целью записывается функционал

$$W(a, b, \omega, Y) = \sum_{i=0}^{I_0-1} (Y(Ti) - y_M(c, Ti))^2 = \sum_{i=0}^{I_0-1} \left(Y(Ti) - \sum_{l=1}^L (a_l \cos \omega_l Ti + b_l \sin \omega_l Ti) \right)^2. \quad (3)$$

Вычисление оптимальных параметров БПФН проводится с помощью решения задачи минимизации введенного функционала:

$$(a^\circ, b^\circ, \omega^\circ) = \arg \left\{ \min_{a, b, \omega \in \bar{\Omega}_0^L} W(a, b, \omega, Y) \right\}. \quad (4)$$

Алгоритм оценивания параметров БПФН на локальном интервале базируется на двухэтапной аппроксимации [1–3]. БПФН является линейной функцией по части параметров и представляется в следующем виде:

$$\sum_{l=1}^L (a_l \cos \omega_l Ti + b_l \sin \omega_l Ti) = \beta^T \varphi(\omega, Ti), \quad i = 0, 1, \dots, N - 1,$$

где вектор линейных параметров $\beta^T = (a_1, \dots, a_L, b_1, \dots, b_L)$ имеет размерность $(2L, 1)$, $\varphi(\omega, Ti)$ — векторная базисная полигармоническая функция,

$$\varphi(\omega, Ti)^T = (\cos(\omega_1 Ti), \dots, \cos(\omega_L Ti), \sin(\omega_1 Ti), \dots, \sin(\omega_L Ti)). \quad (5)$$

Пусть вектор наблюдений Y имеет размерность $(N, 1)$. С использованием $\varphi(\omega, Ti)$ формируется матрица плана сигнала $X(\omega)$ размерности $(N, 2L)$:

$$Y^T = (y(T \cdot 0), y(T \cdot 1), \dots, y(T(N - 1))), \quad X(\omega) = \begin{pmatrix} \varphi(\omega, T \cdot 0) \\ \varphi(\omega, T \cdot 1) \\ \dots \\ \varphi(\omega, T(N - 1)) \end{pmatrix}. \quad (6)$$

Применяя (6), функционал (3) можно записать в векторно-матричном виде:

$$W(\beta, \omega, Y) = (Y - X(\omega)\beta)^T (Y - X(\omega)\beta).$$

На первом этапе аппроксимации для $W(\beta, \omega, Y)$ фиксируются нелинейные частотные параметры $\omega = \text{const}$ с учетом, что $\omega \in \bar{\Omega}_0^L$, и определяются частично оптимальные линейные параметры $\beta^\circ(\omega, Y)$:

$$\beta^\circ(\omega, Y) = \arg \left\{ \min_{\beta, \omega = \text{const}} W(\beta, \omega, Y) \right\},$$

которые вычисляются на основе решения системы линейных уравнений

$$D(\omega) = X^T(\omega)X(\omega), \quad b(\omega, Y) = X^T(\omega)Y, \quad \beta^\circ(\omega, Y) = D(\omega)^{-1}b(\omega, Y), \quad (7)$$

где квадратная матрица $D(\omega)$ имеет размерность $(2L, 2L)$, а вектор коэффициентов Фурье $b(\omega, Y)$ — размерность $(2L, 1)$. На основе $\beta^\circ(\omega, Y)$ формируется функционал частичной остаточной суммы, зависящий только от ω ,

$$W_0(\omega, Y) = W(\beta^\circ(\omega, Y), \omega, Y) = -\beta^{\circ T}(\omega, Y)b(\omega, Y). \quad (8)$$

На втором этапе аппроксимации для $W_0(\omega, Y)$ осуществляется минимизация по $\omega \in \bar{\Omega}_0^L$. Для БПФН определяются оценки частотных параметров ω° и после подстановки ω° в последнее уравнение системы уравнений (7), вычисляются оценки ампли-

тудных параметров β° для локального интервала с точками $i = 0, 1, \dots, I_0 - 1$:

$$\omega^\circ = \omega^\circ(Y) = \arg\{\min_{\omega \in \Omega_0} W_0(\omega, Y)\},$$

$$\beta^\circ = \beta^\circ(\omega^\circ, Y) = D(\omega^\circ)^{-1}b(\omega^\circ, Y). \quad (9)$$

Алгоритм оценивания параметров БПФН реализуется на основе формул (5, 6), проведении вычислений по формулам (7, 8) и минимизации (9) с учетом упорядоченности (2).

2. Метод задания векторов поисковых частот и эффективность частотного поиска

Ввиду того, что функционал $W_0(\omega, Y)$ зависит от многих переменных и является многоэкстремальным [1, 2], его минимизацию целесообразно проводить на основе поисковой процедуры нулевого порядка. Для указанной минимизации и получения оценок амплитудных параметров β° по формуле (9) реализуется конструктивное перечисление векторов поисковых частот. Пусть значения координат векторов поисковых частот выбираются на сетке дискретных значений ω_k в диапазоне $(\omega_{\min}, \omega_{\max})$:

$$\Delta\omega = (\omega_{\max} - \omega_{\min}) / (k_f - 1),$$

$$\omega_k = \omega_{\min} + \Delta\omega(k - 1), \quad k = 1, \dots, k_f,$$

где k_f — число дискретных частот на сетке; $\Delta\omega$ — шаг дискретности. Частоты $\omega_{k_1}, \omega_{k_2}, \dots, \omega_{k_L}$ назначаются из набора дискретных значений $\omega_k, k = 1, \dots, k_f$.

Очевидно, индексы k_1, k_2, \dots, k_L для этих частот с учетом неравенств (2) должны удовлетворять системе из L неравенств

$$1 \leq k_1 \leq k_f - L + 1, \quad k_1 + 1 \leq k_2 \leq k_f - L + 2, \dots,$$

$$k_{L-2} + 1 \leq k_{L-1} \leq k_f - 1, \quad k_{L-1} + 1 \leq k_L \leq k_f.$$

Для $L = 2, 3, 4$ системы неравенств для индексов представляются следующим образом: для $L = 2$ — $1 \leq k_1 \leq k_f - 1, k_1 + 1 \leq k_2 \leq k_f$; для $L = 3$ — $1 \leq k_1 \leq k_f - 2, k_1 + 1 \leq k_2 \leq k_f - 1, k_2 + 1 \leq k_3 \leq k_f$; для $L = 4$ — $1 \leq k_1 \leq k_f - 3, k_1 + 1 \leq k_2 \leq k_f - 2, k_2 + 1 \leq k_3 \leq k_f$.

$\leq k_f - 1, k_3 + 1 \leq k_4 \leq k_f$. Введенные индексы удобно объединить в векторы: $k^T = (k_1, k_2, \dots, k_L)$.

Легко заметить, что в общем случае задача конструктивного перечисления векторов индексов решается на основе известных комбинаторных алгоритмов [14, 15]. Для случая малых значений k_f, L последовательности векторов индексов $k^{s^T} = (k_1^s, k_2^s, \dots, k_L^s), s = 1, \dots, s_f$, достаточно просто могут быть сформированы и пронумерованы на основе вложенных циклов. В таблице индексов для $L = 3, k_f = 7$ приведен пример сформированной по указанной схеме последовательности векторов индексов $k^{s^T} = (k_1^s, k_2^s, k_3^s)$.

Для данного примера число векторов индексов составляет $s_f = 35$.
Метод задания поисковых частот основывается на том, что каждому сформированному вектору индексов k^s может быть поставлен в соответствие вектор поисковых частот ω^s , где s — номер вектора. Координаты $\omega_{k_1}^s, \omega_{k_2}^s, \dots, \omega_{k_L}^s$ для этого вектора частот ω^s можно вычислить по следующим формулам:

Для данного примера число векторов индексов составляет $s_f = 35$.

Метод задания поисковых частот основывается на том, что каждому сформированному вектору индексов k^s может быть поставлен в соответствие вектор поисковых частот ω^s , где s — номер вектора. Координаты $\omega_{k_1}^s, \omega_{k_2}^s, \dots, \omega_{k_L}^s$ для этого вектора частот ω^s можно вычислить по следующим формулам:

$$\omega^s = (\omega_{k_1}^s, \omega_{k_2}^s, \dots, \omega_{k_L}^s), \quad \omega_{k_l}^s = \omega_{\min} + \Delta\omega(k_l^s - 1),$$

$$l = 1, 2, \dots, L, \quad s = 1, \dots, s_f. \quad (10)$$

Отыскание оптимальных частотных и амплитудных параметров БПФН на локальном интервале с Y сводится к последовательным вычислениям: ω^s — по формуле (10); $\varphi(\omega^s, Ti)$ — по формуле (5); $X(\omega^s)$ — по формуле (6); $D(\omega^s), b(\omega^s, Y), \beta^\circ(\omega^s, Y)$ — по формуле (7) и $W_0(\omega^s, Y)$ — по формуле (8), $s = 1, \dots, s_f$, и организации перебора

$$s^\circ = \arg\{\min_{s=1, \dots, s_f} W_0(\omega^s, Y)\}, \quad \omega^\circ = \omega^{s^\circ}, \quad \beta^\circ = \beta^\circ(\omega^{s^\circ}). \quad (11)$$

Вычисления по указанным формулам составляют основу последовательного алгоритма, на котором основывается предлагаемый СВАН.

Таблица индексов

s	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
k_1^s	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	2	2
k_2^s	2	2	2	2	2	3	3	3	3	4	4	4	5	5	6	3	3	3
k_3^s	3	4	5	6	7	4	5	6	7	5	6	7	6	7	7	4	5	6
s	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	
k_1^s	2	2	2	2	2	2	2	3	3	3	3	3	3	4	4	4	5	
k_2^s	3	4	4	4	5	5	6	4	4	4	5	5	6	5	5	6	6	
k_3^s	7	5	6	7	6	7	7	5	6	7	6	7	7	6	7	7	7	

Введение упорядоченности для координат векторов ω из $\bar{\Omega}_0^L$ по (2) существенно снижает количество поисковых вычислений по сравнению с количеством поисковых вычислений без упорядоченности для $\Omega_0^L = \{(\omega_1, \dots, \omega_L): \omega_{\min} \leq \omega_l \leq \omega_{\max}, l = 1, \dots, L\}$. В работе [2] приведены вычисления размеров множеств $\bar{\Omega}_0^L$ и Ω_0^L в виде интегралов $V(\Omega_0^L)$, $V(\bar{\Omega}_0^L)$:

$$V(\Omega_0^L) = \int_{\omega \in \Omega_0^L} d\omega_1, \dots, d\omega_L,$$

$$V(\bar{\Omega}_0^L) = \int_{\omega \in \bar{\Omega}_0^L} d\omega_1, \dots, d\omega_L.$$

Коэффициент $\varepsilon(L)$ оценивает относительное уменьшение размеров

$$\varepsilon(L) = V(\bar{\Omega}_0^L)/V(\Omega_0^L).$$

Для $L = 1, 2, 3, 4$ данный коэффициент принимает значения $\varepsilon(1) = 1$, $\varepsilon(2) = 0,5$, $\varepsilon(3) = 0,166$, $\varepsilon(4) = 0,002$, которые свидетельствуют о существенном снижении временных затрат на частотный поиск вследствие учета упорядоченности.

Необходимо отметить, что обеспечение эффективности частотной поисковой минимизации функционала $W_0(\omega, Y)$ из выражения (9) сопряжено с двумя взаимосвязанными проблемами.

Первая проблема эффективности состоит в обеспечении единственности частотного минимума ω° для $W_0(\omega, Y)$. Рассмотрим достаточно реалистичное предположение, когда исходный многочастотный нестационарный сигнал представляет собой сумму составляющих узкополосных сигналов и его наблюдения на локальном интервале могут быть приближенно представлены в виде суммы синусоидальных сигналов с аддитивным шумом:

$$y(Ti) = \sum_{l=1}^L (a_{0l} \cos \omega_{0l} Ti + b_{0l} \sin \omega_{0l} Ti) + w(Ti),$$

$$i = 0, 1, \dots, N - 1. \quad (12)$$

В работах [1, 2, 16] для случая $L = 1, 2$ и с принятым во внимание указанным предположением, показано, что функционал $W_0(\omega, Y)$ с наблюдениями (12) и ограниченной дисперсией шума σ^2 и локального интервала, на котором уместятся несколько единиц-десятков периодов составляющего узкополосного сигнала с минимальной средней частотой, с учетом упорядоченности частот имеет единственный глубокий глобальный минимум, окруженный системой не очень глубоких локальных минимумов (и максимумов). Подобное заключение, основываясь на работах [1, 2, 16), может быть справедливым и для $L = 3, 4$. Вполне очевидно, что при сформулированных предположениях имеет место единствен-

ность минимума $W_0(\omega, Y)$, что подтверждается вычислительными экспериментами.

Вторая проблема эффективности состоит в учете погрешностей оценивания частотных и амплитудных параметров моделей БПФН [1, 2, 3, 17]. Примем шаг частотного поиска малым, чтобы не учитывать погрешности от этого фактора. Рассматриваемые погрешности зависят от параметров N, T, σ^2 . Поскольку БПФН представляет собой нелинейную функцию и функционал $W_0(\omega, Y)$ является многоэкстремальным, то рассматриваемые погрешности ведут себя необычным образом: при монотонном увеличении σ^2 имеет место скачкообразное нарушение непрерывности функции погрешностей частоты и амплитуд вследствие перемены местами локальных и глобальных экстремумов [16]. Выяснение характера погрешностей оценок частоты и амплитуд будем рассматривать для малых σ^2 в целях обеспечения корректной линеаризации функционала $W_0(\omega, Y)$. В работе [2] с использованием линеаризации были получены оценки математического ожидания погрешностей, которые оказались равными нулю (оценивание без смещений) и оценки дисперсий погрешностей, которые убывали при увеличении длины локальных интервалов.

3. Спектрально-временной анализ на основе базисных полигармонических функций

Традиционный спектральный анализ на основе оценок БПФН для сигнала $Y(Ti)$ на локальном интервале с точками $i = 0, 1, \dots, I_0 - 1$ состоит в вычислении одномерных массивов оценок оптимальных частот и амплитуд, полученных на основе формулы (9):

$$(f_1^\circ, f_2^\circ, \dots, f_L^\circ), f_l^\circ = \omega_l^\circ / 2\pi, (A_1^\circ, A_2^\circ, \dots, A_L^\circ),$$

$$A_l^\circ = (a_l^{\circ 2} + b_l^{\circ 2})^{1/2}, l = 1, 2, \dots, L.$$

Значениям частот ω_k на сетке и индексам k могут быть поставлены в соответствие амплитуды A_k , $k = 1, \dots, k_f$, где $A_k = (a_k^{\circ 2} + b_k^{\circ 2})^{1/2}$, $l = 1, 2, \dots, L$ и $k = k_l^\circ$ — координаты оптимального вектора индексов, для остальных индексов $k \neq k_l^\circ$ будем иметь $A_k = 0$.

Результатом традиционного спектрального анализа на локальном интервале является одномерный массив распределений амплитуд A_k по частоте, где индекс k определяет распределение по частотам ω_k , $k = 1, \dots, k_f$.

СВАН на основе БПФН для исходного нестационарного многочастотного сигнала $Y(Ti)$ на большом временном интервале с точками $i = 0, 1, \dots, I_{f0} - 1$ реализуется на последовательности скользящих локальных интервалов, содержащих по I_0 точек с граничными точками I_{1j}, I_{2j} ; эти интервалы рас-

положены внутри исходного интервала, j — номер скользящего локального интервала, $j = 1, \dots, m$. На каждом из скользящих локальных интервалов сигналу $Y(T_i)$ ставится в соответствие оценка БПФН. Вычисляются двумерные массивы оптимальных амплитуд и частот, полученных на основе решений задач аппроксимации (4) или (9) для последовательности скользящих локальных интервалов

$$(f_{1j}^\circ, f_{2j}^\circ, \dots, f_{Lj}^\circ), f_{lj}^\circ = \omega_{lj}^\circ/2\pi; (A_{1j}^\circ, A_{2j}^\circ, \dots, A_{Lj}^\circ), A_{lj}^\circ = (a_{lj}^{\circ 2} + b_{lj}^{\circ 2})^{1/2}, l = 1, 2, \dots, L, j = 1, \dots, m.$$

Результатом СВАН на основе БПФН для исходного временного интервала будет двумерный массив распределений амплитуд $A_{k,j}$, где индекс k определяет распределение по частотам ω_k , $k = 1, \dots, k_f$, индекс j определяет распределение по времени, $j = 1, \dots, m$.

4. Оценки временной и емкостной сложности последовательного алгоритма СВАН

Эффективность последовательного алгоритма СВАН (5—8, 10, 11) определяется целым рядом параметров, из которых наиболее важными являются параметры временной и емкостной сложности.

Параметр временной сложности последовательного алгоритма, позволяющий оценить временные затраты, необходимые для получения результата, определяется величиной s_f , которая может быть вычислена, как число сочетаний из k_f элементов по L : $s_f = C(k_f, L)$ и $C(k_f, L) = k_f! / (L!(k_f - L)!)$.

Примем время вычисления, определяемое каждым вектором ω^s , за условную временную единицу (базовую операцию). Тогда временная сложность [18] последовательного алгоритма для перебора в (11) может быть принята равной $T_{f1}(k_f, L) = k_f! / (L!(k_f - L)!)$. Используя формулу Стирлинга и переходя к асимптотическим оценкам [19], после очевидных упрощений для $k_f \gg L$, получим оценку временной сложности в виде функции $T_{f2}(k_f, L) = O((k_f/L)^L)$, которая асимптотически ведет себя так же, как и

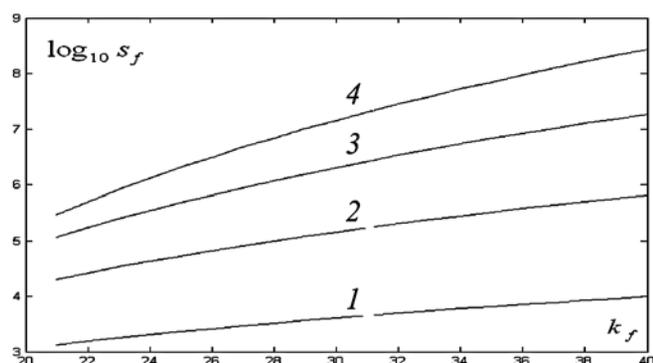


Рис. 1. Графики зависимостей числа частотных векторов $\log_{10} s_f(k_f, L)$

$T_{f1}(k_f, L)$. Таким образом, алгоритм, на котором основывается предлагаемый СВАН, имеет экспоненциальную временную сложность. На рис. 1 представлены графики зависимостей числа векторов поисковых частот $s_f = C(k_f, L)$ в логарифмическом масштабе для параметров $k_f = 21 \dots 40$; параметр L принимал следующие значения: кривая 1 — $L = 3$, кривая 2 — $L = 5$; кривая 3 — $L = 7$; кривая 4 — $L = 9$.

Приведенные на рис. 1 графики экспериментально подтверждают приведенные выше результаты анализа: последовательный алгоритм, на котором основывается предлагаемый СВАН, имеет экспоненциальную временную сложность.

Параметр емкостной сложности последовательного алгоритма (5—8, 10, 11), связанный с необходимым количеством данных, которые должны храниться в памяти, может быть приближенно представлен выражением $V_f = 4(s_f L + s_f L^2) / 10^6$ (Мбайт), где первое слагаемое определяет объем памяти для массива векторов частот, второе слагаемое — объем памяти для квадратных матриц из первого уравнения системы (7). Видно, что предлагаемый последовательный алгоритм СВАН имеет экспоненциальную емкостную сложность.

Можно заключить, что для практического использования предлагаемого СВАН необходима разработка и программная реализация параллельного алгоритма.

5. Структура алгоритма параллельных вычислений для СВАН

Для применения параллельных вычислений требуется, чтобы исходный вариант алгоритма СВАН был определенным образом преобразован.

Структура предлагаемого параллельного алгоритма аналогичная, с определенными допущениями, структуре из работы [20], предполагает наличие главного процесса, связанного посредством интернет-каналов с компьютерными кластерами, расположенными в различных географических точках. В компьютерном кластере реализуется функционирование набора процессов-исполнителей. Главный процесс обеспечивает управление работой параллельного алгоритма. В нем формируются задания для процессов-исполнителей в виде файлов с исполнительными модулями и входными данными. В свою очередь, процессы-исполнители передают результаты своей работы в виде файлов на главный процесс. Будем полагать, что суммарное число процессов-исполнителей составляет величину n_f и что процессы-исполнители имеют одинаковые производительности и объемы оперативной памяти.

Число процессов-исполнителей n_f зависит от объема памяти V_f и объема оперативной памяти для процесса-исполнителя V_{f0} . Используя значения V_f из разд. 4, можно сделать приближенную

оценку n_f для системы параллельных вычислений в зависимости от параметров k_f, L, V_{f0} :

$$n_f = \frac{V_f}{V_{f0}} = \frac{4(L + L^2)s(k_f L)/10^6}{V_{f0}}$$

Известно, что ускорение и эффективность параллельных алгоритмов в большой степени зависит от их балансировки, т. е. от того, насколько равномерно распределяется вычислительная нагрузка между процессами-исполнителями [21]. Балансировка разработанного параллельного алгоритма для СВАН основывается на разбиении последовательности номеров векторов поисковых частот $s = 1, \dots, s_f$ на n_f равных по длине участков. Вводятся параметр $ds_0 = \text{ent}(s_f/n_f)$ и граничные точки s_{1n}, s_{2n} для последовательности индексов $s = 1, \dots, s_f$, которые могут быть вычислены на основе следующих рекуррентных формул:

$$s_{11} = 1, s_{2n} = s_{1n} + ds_0 - 1, s_{1,n+1} = s_{2n} + 1, \\ n = 1, \dots, n_f - 1, s_{2,n_f} = s_f$$

Таким образом, объем вычислений на каждом процессоре пропорционален значению s_f/n_f , что свидетельствует о снижении вычислительной сложности.

Введенное разбиение последовательности номеров векторов индексов на равные участки корректно, поскольку:

- не увеличивает количество вычислений и объем памяти, которые требуются для реализации алгоритма СВАН;

- обеспечивает равномерную загрузку вычислительных узлов за счет разбиения последовательности индексов с номерами $s = 1, \dots, s_f$ на равные участки индексов;

- обеспечивает масштабируемость при увеличении числа вычислительных узлов путем уменьшения длин участков индексов.

Структура алгоритма параллельных вычислений для СВАН основывается на двух шаговых вычислительных процедурах.

1. Выполнение прямых переборов для функционала $W_0(\omega^s, Y)$ по векторам поисковых частот, которые могут быть сведены к переборам по целым индексам $s_{1n} \leq s \leq s_{2n}$. Процессы-исполнители обеспечивают решение задач оптимизации функционалов и нахождения последовательности значений оценок векторов параметров и функционалов $\omega_n^\circ, \beta_n^\circ, W_{0n}, n = 1, \dots, n_f$,

$$s_n^\circ = \arg\left\{ \min_{s_{1n} \leq s \leq s_{2n}} W_0(\omega^s, Y) \right\}, \omega_n^\circ = \omega^{s_n^\circ},$$

$$\beta_n^\circ = \beta^\circ(\omega^{s_n^\circ}), W_{0n} = W_0(\omega^{s_n^\circ}, Y), n = 1, \dots, n_f.$$

2. Осуществление пересылки последовательностей значений оценок $\omega_n^\circ, \beta_n^\circ, W_{0n}, n = 1, \dots, n_f$. Главный процесс обеспечивает нахождение опти-

мальных оценок частотных и амплитудных параметров для БПФН

$$n^\circ = \arg\left\{ \min_{1 \leq n \leq n_f} W_{0n} \right\}, \omega^\circ = \omega_{n^\circ}^\circ, \beta^\circ = \beta_{n^\circ}^\circ.$$

6. Вычислительный эксперимент с модельным параллельным алгоритмом

Вычислительный эксперимент с модельной реализацией параллельного алгоритма СВАН был проведен на основе комплекса MATLAB. Рассматривался упрощенный пример, для которого были сформированы модельные наблюдения одночастотного зашумленного сигнала $Y(Ti), i = 0, 1, \dots, I_{f0} - 1$, с амплитудной и частотной модуляцией:

$$Y(Ti) = E(Ti)\cos(\varphi_1(Ti) + \varphi_2(Ti)) + w(Ti). \quad (13)$$

Амплитудные и фазовые (частотные) модуляционные функции были представлены следующим образом:

$$\varphi_1(Ti) = 2\pi f_0 Ti + \varphi_0, \quad \varphi_2(Ti) = \beta(Ti)^2/2, \\ E(Ti) = E_0 \exp(-\alpha(Ti - N_0 T)^2). \quad (14)$$

В выражении (13) $w(Ti)$ — случайные нормально-распределенные числа с нулевым математическим ожиданием и дисперсией σ^2 имитировали шум в наблюдениях.

Параметры модели принимали следующие значения: $I_{f0} = 768, T = 60$ с, $E_0 = 1,0, f_0 = 0,2 \cdot 10^{-3}$ Гц ($T_0 = 1/f_0 = 5 \cdot 10^3$ с), $\varphi_0 = 0,2, \beta = 0,75 \cdot 10^{-8}$ Гц/с, $\alpha = 1 \cdot 10^{-8}$ с⁻², $N_0 = 350$. Для модели (13, 14) частота сигнала $f(Ti) = \dot{\varphi}_1(Ti) + \dot{\varphi}_2(Ti) = f_0 + \beta Ti$ изменялась по линейному закону, амплитуда — по гауссовскому закону. Частота модельного сигнала изменялась от значения $f(T \cdot 0) = 0,2 \cdot 10^{-3}$ Гц до $f(T \cdot (N_f - 1)) = 0,345 \cdot 10^{-3}$ Гц. Длительность сигнала (13) была равна $t_f = I_{f0} T = 768 \cdot 60 = 46\,080$ с (12,8 ч). Модельный сигнал для $\sigma = 0$ изображен на рис. 2.

Локальный интервал принимали состоящим из $N = 128$ точек и его длительность равнялась $TI_0 = 7,68 \cdot 10^3$ с (2,13 ч); в среднем, на локальных интервалах умещалось $\sim 1,5$ периода исходного сиг-

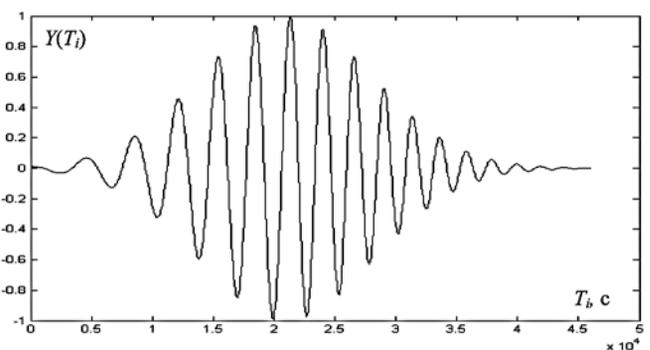


Рис. 2. Модельный одночастотный сигнал с амплитудной и частотной модуляцией

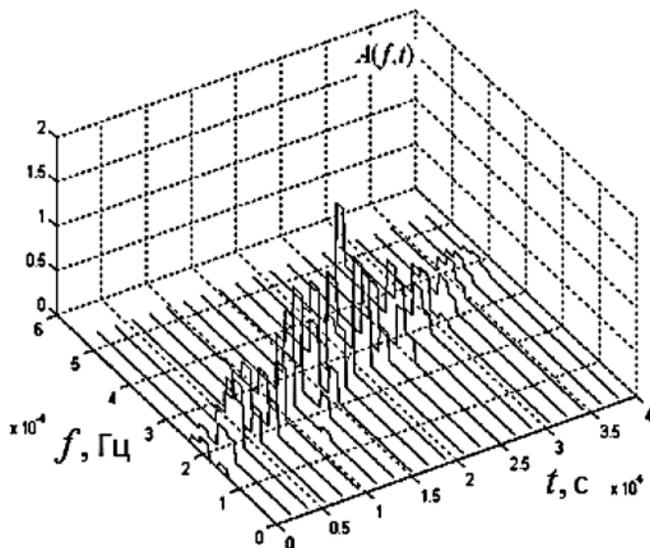


Рис. 3. СВАН-диаграмма модельного одночастотного сигнала с амплитудной и частотной модуляцией

нала. Параметр скольжения выбирали равным $dI_0 = I_0/4 = 32$, при этом число локальных интервалов оказывалось равным $m_f = 21$. Были заданы следующие параметры для формирования векторов поисковых частот: $k_f = 21$, $L = 3$, число векторов составило $s_f = 165$. Были назначены границы поискового диапазона по частоте $f_{\min} = 0,1 \cdot 10^{-3}$ Гц, $f_{\max} = 0,6 \cdot 10^{-3}$ Гц, шаг дискретности по частоте принимал значение $\Delta f = 0,25 \cdot 10^{-4}$ Гц. Для вычислительного эксперимента принимали $n_f = 5$.

На рис. 3 представлена СВАН-диаграмма значений распределений амплитуд $A(f, t)$ в зависимости от частоты и времени.

Общая длительность функций-оценок на СВАН-диаграмме составила $(m_f - 1)TdI_0 = 38\,400$ с. Из представленной СВАН-диаграммы видно, что при данных параметрах сигнала и принятых параметрах алгоритма осуществляется работа алгоритма СВАН с удовлетворительной точностью: оценки частоты сигнала реализуются в виде линейной функции времени, оценки амплитуды — в виде гауссовой кривой.

Заключение

Предложенный параллельный алгоритм может быть использован в задачах спектрально-временного анализа на базисных полигармонических функциях для многочастотных нестационарных сигналов.

Анализ последовательного алгоритма для СВАН показал, что по временной и емкостной сложности он относится к классу экспоненциальных. Разработанный параллельный алгоритм для СВАН обеспечил снижение временной и емкостной сложности.

Разработанный метод формирования последовательности частотных векторов, учитывающий упо-

рядочность для координат частотных параметров, обеспечил снижение временных затрат на поиск при построении БПФН.

Вычислительный эксперимент, реализованный на модельном сигнале, продемонстрировал работоспособность и возможность обеспечения удовлетворительной точности предложенного параллельного алгоритма СВАН.

Предложенный алгоритм параллельных вычислений может быть обобщен и использован для распараллеливания оптимизационных задач, основанных на поисковых процедурах нулевого порядка — прямых переборах в многомерных параллелепипедах параметров.

Работа выполнена при поддержке "Программы фундаментальных исследований Президиума РАН № 1 по стратегическим направлениям развития науки на 2014 г. "Фундаментальные проблемы математического моделирования".

Список литературы

1. Гетманов В. Г. Об алгоритме поиска по частоте в задаче оценивания параметров моделей полигармонических сигналов // Автометрия. 2009. № 3. С. 83–89.
2. Гетманов В. Г. Цифровая обработка нестационарных колебательных сигналов на основе локальных и сплайновых моделей. М.: Изд-во НИЯУ МИФИ. 2011. 298 с.
3. Гетманов В. Г. Технология спектрально-временного анализа нестационарных сигналов на основе локальных и сплайновых аппроксимационных моделей // Труды 11-й международной конференции "Цифровая обработка сигналов и ее приложения". М.: РНТОРЭС им. А. С. Попова. 2009. Т. 1. С. 144–147.
4. Лайонс Р. Цифровая обработка сигналов. М.: Бином-Пресс, 2006. 656 с.
5. MATLAB Digital Signal Processing Toolbox./www.mathworks.com.
6. Cohen L. Time-Frequency Analysis. Theory and Application // Prentice Hall Signal Processing Series. 1995. 299 p.
7. Hlawatch F., Auger F. Time-Frequency Analysis: Concepts and Methods. London (UK): ISTE and Wiley, 2008. 440 p.
8. Марпл мл. С. Л. Цифровой спектральный анализ и его приложения / Пер. с англ. М.: Мир. 1990. 584 с.
9. Барабанов А. Б. Идентификация параметров полигармонической модели речевого сигнала // Материалы XII Всероссийского совещания по проблемам управления (ВСПУ—2014). Москва, 16–19.06.2014. С. 3038–3049.
10. Ramos P. M., Silva M. F., Martins R. C., Serra A. C. Simulation and experimental results of multiharmonic least-squares fitting algorithms applied to periodic signals // IEEE Transactions on Instrumentations and Measurements. 2007. V. 56, N 4. P. 1412–1418.
11. Root D. E., Verspecht J., Sharrit D., Wood J. Broad-band poly-harmonic distortion behavioral models from fast automated simulation and large-signal vertical network measurements // IEEE Transactions on Microwave Theory and Techniques. 2005. V. 53, Is. 11. P. 3656–3664.
12. Якушев Д. И. Геоинформационное моделирование пространственно-временных геофизических процессов с полигармонической структурой. Автореферат диссертации на соискание уч. степ. д.т.н. / ЛЭТИ им. В. И. Ульянова (Ленина). Санкт-Петербург, 2008.
13. Salinas J. R., Garcia-Lagos F., Joya G., Sandoval F. Sine Fitting Multiharmonic Algorithms Implemented Neural Network // Computational and Ambient Intelligence. Springer. 2007. P. 622–629.
14. Тимашевская Н. Е. О нумерации перестановок и сочетание для организации параллельных вычислений в задачах проектирования управляющих систем // Известия Томского политехнического университета. 2004. Т. 307, № 6. С. 18–20.

15. **Липский В.** Комбинаторика для программистов. М.: Мир. 1988. 200 с.

16. **Гетманов В. Г., Кузнецов П. А., Скворцов О. Б.** Измерение нестационарных амплитуд и частот узкополосных сигналов // *Метрология*. 1997. № 11. С. 30–39.

17. **Гетманов В. Г.** О частотном подпоиске в задаче оценивания параметров кусочно-синусоидальных функций // *Автометрия*. 1992. № 2. С. 93–98.

18. **Макконел Дж.** Анализ алгоритмов. Вводный курс. М.: Техносфера, 2002. 304 с.

19. **Новиков Ф. А.** Дискретная математика: учебник для вузов. Стандарт третьего поколения. СПб.: Питер, 2011. 384 с.

20. **Алешин И. М., Мишин Д. Ю., Жижин М. Н.** и др. Применение распределенных вычислительных систем при определении параметров сейсмической анизотропной коры и верхней мантии // *Геофизические исследования*. 2009. Т. 10, № 4. С. 34–47.

21. **Борзунов Г. И., Петрова Т. В., Сучкова Е. А.** Повышение эффективности масштабирования набора задач при поиске экстремальных разбиений // *Безопасность информационных технологий*. 2011. № 3. С. 116–120.

V. G. Getmanov^{1, 2}, Professor, Chief Researcher, **G. I. Borzunov^{2, 3}**, Professor,

¹The Geophysical Center of the Russian Academy of Sciences

²National Research Nuclear University MEPHI (Moscow Engineering Physics Institute)

³Moscow state University of design and technology

The Algorithm of Parallel Calculations for a Problem of the Spectral-Time Analysis on Basic Polyharmonic Functions

The algorithm of parallel calculations for a problem of the spectral-time analysis (STAN) on basic polyharmonic functions is offered. The analysis of consecutive algorithm for STAN showed that on temporary complexity it belongs to the class of exponential algorithms, it formed the basis for development of parallel algorithm. The method of a task of vectors of search frequencies is created. It is known that the speedup and efficiency of parallel algorithms heavily depends on its balancing, i. e. how evenly distributed computational load between processes executing. Balancing the developed parallel algorithm is based on partitioning the sequence of numbers of the vectors of the search frequencies at equal length sections. Introduced a splitting sequence of vectors of indices into equal parts correctly because: does not increase the amount of computation and memory required for the implementation of the algorithm; ensures uniform loading of the compute nodes by partitioning the sequence of index numbers as equal parts of the index; provides scalability when increasing the number of compute nodes by reducing the lengths of the parts of the index. In the offered parallel algorithm effective frequency search is used. Results of computing experiment with model realization of parallel algorithm for signal STAN with amplitude and frequency modulation are given.

Keywords: parallel calculations, spectral time analysis, polyharmonic functions, approximation, estimation

References

1. **Getmanov V. G.** Ob algoritme poiska po chastote v zadache ot-senivaniya parametrov modelej poligarmonicheskikh signalov. *Avto-metriya*. 2009. N. 3. P. 83–89.

2. **Getmanov V. G.** *Tsifrovaya obrabotka nestatsionarnykh kole-batel'nykh signalov na osnove lokal'nykh i splajnovykh modelej*. M.: Izd-vo NIYAU MIFI. 2011. 298 p.

3. **Getmanov V. G.** Tekhnologiya spektral'no-vremennogo analiza nestatsionarnykh signalov na osnove lokal'nykh i splajnovykh ap-proksimatbionnykh modelej. *Trudy 11-oy mezhdunarodnoj konferentsii "Tsifrovaya obrabotka signalov i ego prilozheniya"*. M.: RNTOREHS im. A. S. Popova. 2009. Vol. 1. P. 144–147.

4. **Lajons R.** *Tsifrovaya obrabotka signalov*. M.: Binom-Press. 2006. 656 p.

5. **MATLAB** Digital Signal Processing Toolbox. www.math-works.com.

6. **Cohen L.** *Time-Frequency Analysis. Theory and Application*. Prentice Hall Signal Processing Series. 1995. 299 p.

7. **Hlawatch F., Auger F.** *Time-Frequency Analysis: Concepts and Methods*. London (UK): ISTE and Wiley, 2008. 440 p.

8. **Marpl ml. S. L.** *Tsifrovoi spektral'nyi analiz i ego prilozheniya*. Per. s angl. M.: Mir, 1990. 584 p.

9. **Barabanov A. B.** Identifikatsiya parametrov poligarmonicheskoy modeli rechevogo signala. Materialy XII Vserossiyskogo soveshchaniya po problemam upravleniya (VSPU–2014). Moskva. 16–19.06.2014. P. 3038–3049.

10. **Ramos P. M., Silva M. F., Martins R. C., Serra A. C.** Simulation and experimental results of multiharmonic least-squares fitting algorithms applied to periodic signals. *IEEE Transactions on Instru-mentations and Measurements*. 2007. V. 56, N. 4. P. 1412–1418.

11. **Root D. E., Verspecht J., Sharrit D., Wood J.** Broad-band poly-harmonic distortion behavioral models from fast automated

simulation and large-signal vertical network measurements. *IEEE Transactions on Microwave Theory and Techniques*. 2005. V. 53, Iss. 11. P. 3656–3664.

12. **Iakushev D. I.** *Geoinformacionnoe modelirovanie prostranstven-no-vremennykh geofizicheskikh processov s poligarmonicheskoy strukturoy*. Avtoreferat dissertatsii na soiskanie uch. step. d.t.n. LETI im. V. I. Ulianova (Lenina), Sankt-Peterburg. 2008.

13. **Salinas J. R., Garcia-Lagos F., Joya G., Sandoval F.** Sine Fitting Multiharmonic Algorithms Implemented Neural Network. *Computational and Ambient Intelligence*. Springer. 2007. P. 622–629.

14. **Timashevskaya N. E.** O numeratsii perestанovok i sochetani dlia organizatsii paralelnykh vychisleni v zadachah proektirovaniia upravliaiushchikh sistem. *Izvestiia Tomskogo politehnicheskogo universiteta*. 2004. Vol. 307, N. 6. P. 18–20.

15. **Lipski V.** *Kombinatorika dlia programmistov*. M.: Mir, 1988. 200 p.

16. **Getmanov V. G., Kuznecov P. A., Skvorcov O. B.** Izmerenie nestacionarnykh amplitud i chastot uzkopolosnykh signalov. *Metrologiia*. 1997. N. 11. P. 30–39.

17. **Getmanov V. G.** O chastotnom podpoiske v zadache ocenivaniia parametrov kusochno-sinusoidalnykh funkci. *Avtometriia*. 1992. N. 2. P. 93–98.

18. **Макконел Дж.** *Анализ алгоритмов. Вводный курс*. М.: Техносфера. 2002. 304 p.

19. **Новиков Ф. А.** *Дискретная математика: Учебник для вузов. Стандарт третьего поколения*. СПб.: Питер. 2011. 384 p.

20. **Алшин И. М., Мишин Д. Ю., Жижин М. Н.** и др. Применение распределенных вычислительных систем при определении параметров сейсмической анизотропной коры и верхней мантии. *Геофизические исследования*. 2009. Т. 10, N. 4. P. 34–47.

21. **Борзунов Г. И., Петрова Т. В., Сучкова Е. А.** Povyshenie effektivnosti masshtabirovaniia nabora zadach pri poiske ekstremalnykh razbieni. *Bezopasnost informatsionnykh tehnologi*. 2011. N. 3. P. 116–120.

Ю. Г. Табаков, аспирант, e-mail: vurik204@rambler.ru,
Воронежская государственная лесотехническая академия, г. Воронеж

Модель и алгоритм обработки низкочастотного сигнала для тренажера на основе вейвлет-преобразований

Рассмотрены основные задачи при обработке низкочастотных сигналов, снятых с коры головного мозга, и возможные способы их решения. Предложены математическая модель и алгоритм, построенные на модифицированных вейвлет-преобразованиях Добеши и Морле, позволяющие обрабатывать низкочастотные сигналы в реальном времени и анализировать полученные данные. Представлены структурная схема помодульной обработки сигнала и блок-схема алгоритма. Результаты исследований могут быть применены в специализированных медицинских учреждениях по восстановлению опорно-двигательной системы человека.

Ключевые слова: анализ сигналов, обработка сигналов, низкочастотные сигналы, программирование, интеллектуальный тренажер, вейвлет Добеши, вейвлет Морле, вейвлет-преобразование, математическая модель

Введение

С каждым годом информационные технологии развиваются с огромной скоростью, а вместе с ними появляются различные проблемы создания, хранения, управления и обработки данных. Так, в последнее время стали использовать методы управления различными высокотехническими устройствами с помощью низкочастотных сигналов, снятых с поверхности коры головного мозга человека с помощью датчиков электроэнцефалограммы (ЭЭГ). Эти методы находят все более широкое применение в видеоиграх, управлении инвалидными колясками, радиоуправляемыми самолетами и т. п.

При обработке низкочастотных сигналов в целях формирования управляющих сигналов возникает ряд проблем, которые часто усложняют анализ полученных данных. Для этого необходимо решить следующие задачи:

- определение амплитудных и частотных характеристик низкочастотных сигналов;
- помодульное выполнение процесса обработки для получения информационного сигнала;
- анализ полученных данных в целях выявления управляющих сигналов;
- проведение мониторинга данных в реальном времени.

При решении перечисленных задач следует учитывать высокоамплитудные остроконечные формы сигналов. Эти волны сигналов хаотичны и не постоянны, что может привести к различным всплескам на фоне общей активности коры головного мозга человека [1]. В связи с этим было разработано специальное математическое и программное обеспечение на основе вейвлет-преобразований Добеши и Морле, позволяющее обрабатывать данные в реальном времени.

Таким образом, решив поставленные задачи, можно приступить к созданию интеллектуального тренажера на основе низкочастотных сигналов с обучающейся информационной базой. Эти тренажеры предназначены для полного или частичного восстановления опорно-двигательной системы человека через стимуляцию мышц, т. е. тренажеры подают электрические разряды с определенной частотой и силой, что приводит к сокращению и расслаблению мышц конечностей рук или ног [2].

Например, существуют такие тренажеры, как миостимуляторы: компактный прибор для миостимуляции Butterfly XFT1002, стимулятор мышц M18, тренажер для живота электронный Abtronic X2, пояс-тренажер для пресса жимфлекс, устройство для миостимуляции мышц Sport-Elec Multi-Sport, миостимулятор Спорт-Элек SP-2 и т. п., не обладающие специальным программным обеспечением и обучающей информационной системой.

Подобные тренажеры не позволяют полностью автоматизировать процесс восстановительного лечения опорно-двигательной системы человека с помощью низкочастотных сигналов. Автоматизированный процесс, управляемый низкочастотными сигналами, позволяет проводить специализированные тренировки, которые не только подготавливают мышцы к работе, но и восстанавливают фантомные связи между корой головного мозга и конечностями рук или ног [3].

Цель статьи заключается в разработке математической модели и алгоритма в виде подключаемых модулей для программного обеспечения, которые позволят решить поставленные задачи при анализе и обработке низкочастотных сигналов, снятых с поверхности коры головного мозга, а также определить статистические зависимости этих отклонений от естественного нормального распространения сигналов.

Способ снятия и обработки низкочастотных сигналов

При снятии низкочастотных сигналов с поверхности коры головного мозга человека на первых этапах обработки необходимо применять различные критерии в разработанных моделях [4], благодаря чему можно достичь желаемых результатов. На рис. 1 отображена структурная схема снятия и обработки низкочастотных сигналов.

Для достижения максимального результата низкочастотный сигнал обрабатывается в режиме реального времени с помощью разработанного специального математического и программного обеспечения САЗСМЧ, которое построено на основе вейвлет-преобразования Добеши и Морле [5]. Загрузочный логотип программы представлен на рис. 2.

Функции сигналов по времени и частоте связаны друг с другом вейвлет-преобразованием. За основу функций вейвлет-преобразования в обработке низкочастотных сигналов будут использоваться разработанные модифицированные алгоритмы Добеши [6, 7] и Морле [8]. Зависимости амплитуд низкочастотных (НЧ) сигналов представлены на рис. 3 и рис. 4 соответственно для каждого метода.

Алгоритм обработки НЧ сигналов на основе методов вейвлет-преобразований

При решении поставленных задач следует учитывать некоторые правила, которые должна включать в себя математическая модель:

- среднее значение функции на всем промежутке должно быть равно 0;
- функция быстро убывает при $t = \infty$.

Исходя из этих правил, можно описать основную функцию вейвлет-преобразования, которая имеет следующий вид (1) [9]:

$$W(a, b) = \frac{1}{a} \int_{-\infty}^{\infty} \psi\left(\frac{t-b}{a}\right) f(t) dt, \quad (1)$$

где $W(a, b)$ — вейвлет-коэффициенты; ψ — вейвлет; a — момент времени; b — параметр, обратный частоте; $f(t)$ — анализируемые данные низкочастотного сигнала, зависящие от времени t .

В процессе снятия НЧ сигнала могут возникать различные ошибки, что негативно влияет на обработку. Поэтому на первоначальном этапе необходимо учитывать каждый временной отрезок сигнала (рис. 5). Подобный метод приводит к снижению погрешностей и улучшает обработку низкочастотного сигнала [10].

Следует учитывать основные критерии функций вейвлет-преобразования, называемые "функциональные координаты". Эти базисы позволят смещать целевую функцию (1) на заданный временной диапазон в реальном времени [11].

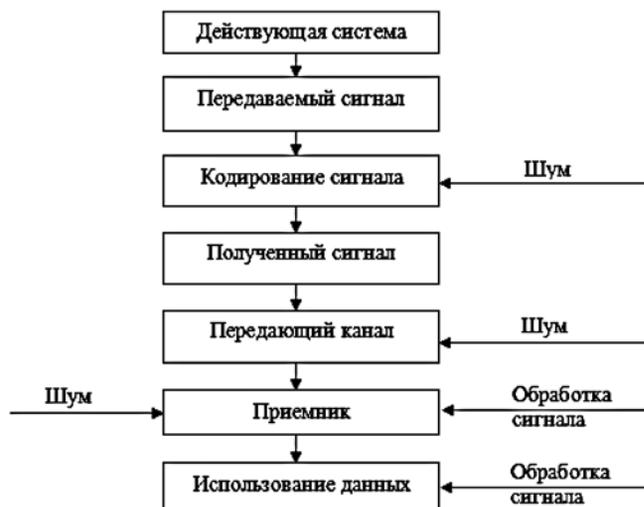


Рис. 1. Структурная схема снятия и обработки сигналов

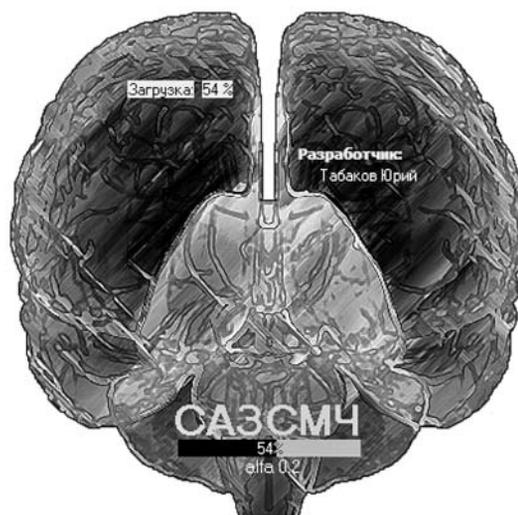


Рис. 2. Загрузочный логотип программы САЗСМЧ

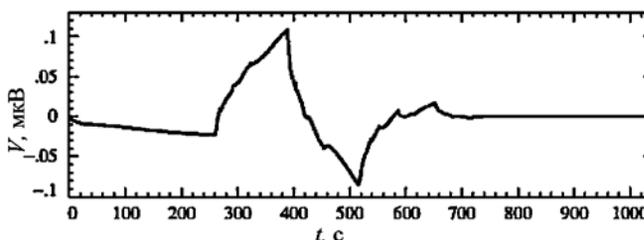


Рис. 3. Временная зависимость амплитуды НЧ сигнала на основе метода вейвлета Добеши

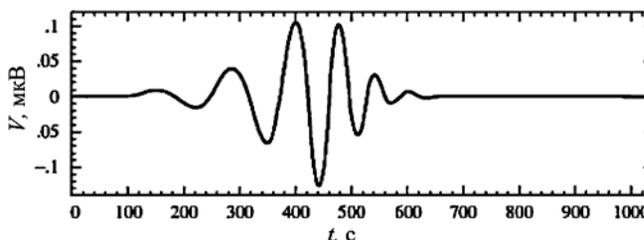


Рис. 4. Временная зависимость амплитуды НЧ сигнала на основе метода вейвлета Морле

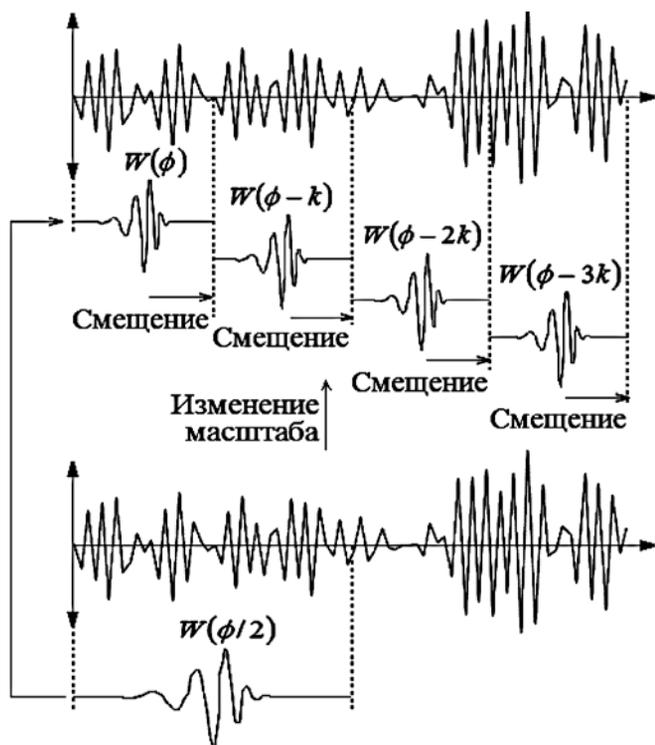


Рис. 5. Временные отрезки сигнала для его помодульной обработки

Разрабатываемый алгоритм основан на модифицированных функциях вейвлет-преобразований Добеши [6, 7] и Морле [8], предназначенных для обработки НЧ сигналов амплитудой от 5 до 15 мкВ и частотой в диапазоне 10...50 Гц. Таким образом, математическая модель получается гибкой и функциональной:

$$\varphi(x) = \sqrt{2} \sum_{k=0}^{2M-1} h_k \varphi(2x - k), \quad (2)$$

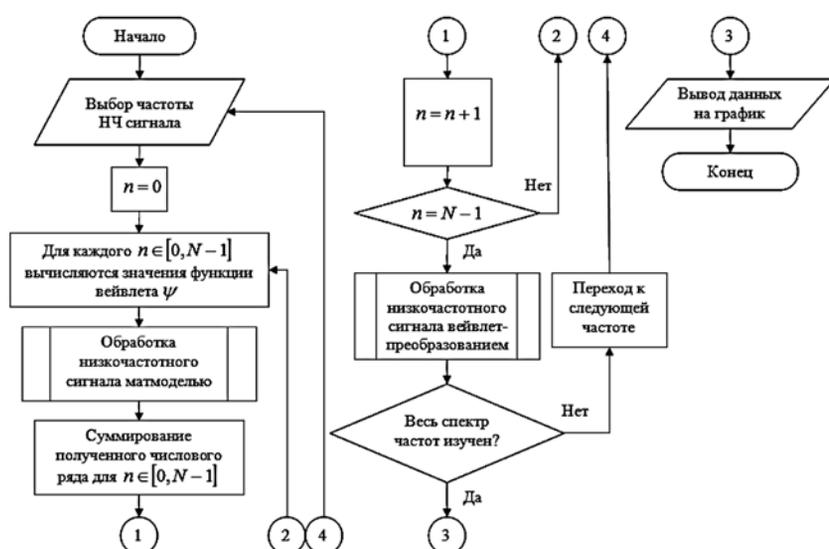


Рис. 6. Алгоритм обработки низкочастотного сигнала

где M — определяет число коэффициентов вейвлета; k — целочисленные трансляции; $h_k = \sqrt{2} \int \varphi(x) \overline{\varphi(2x - k)} dx$ — длина области вейвлета; φ — растяжение/сжатие вейвлета.

Функция вейвлет-преобразования Добеши имеет вид

$$\psi(x) = \sqrt{2} \sum_{k=0}^{2M-1} h_k \varphi(2x - k).$$

Функция вейвлет-преобразования Морле имеет вид

$$\psi(x) = \exp\left(ik_b r - \frac{k^2}{2}\right),$$

где r — среднеквадратическое отклонение; i — начальное отклонение; $b = 2M - 1$ — максимальный коэффициент.

Многомасштабный вейвлет-анализ основан на разложении сигналов на всем промежутке обработки с заданным масштабом j_n :

$$f(x) = \sum_{k=0}^{2M-1} s_{j_n, k} \phi_{j_n, k} + \sum_{j=j_n}^{j_{\max}} \sum_{k=0}^{2M-1} d_{j, k} \psi_{j, k}$$

где $\psi_{j, k}$ — смещенные версии масштабного "материнского" вейвлета ψ ; $\phi_{j_n, k}$ — масштабированные функции ϕ ; $s_{j_n, k}$ — коэффициенты аппроксимации; $d_{j, k}$ — детализирующие коэффициенты.

В математическую модель (2) был внедрен "отцовский вейвлет", позволяющий повысить обработку низкочастотного сигнала:

$$\phi(t) = \sqrt{2} \sum_{k=0}^{2M-1} h_k \phi(2t - k).$$

НЧ сигнал, снятый с поверхности коры головного мозга, необходимо преобразовывать для хранения и дальнейшей обработки в целях формирования управляющих сигналов. Функции преобразования [12] сигнала записываются следующим образом:

- 8-битное преобразование:

$$(\text{long})(\text{unsigned char})pHdr \rightarrow \rightarrow lpData[n] - 128;$$

- 16-битное преобразование:

$$(\text{long})(\text{signed short})pHdr \rightarrow \rightarrow lpData[n].$$

Алгоритм обработки низкочастотного сигнала представлен на рис. 6.

Заключение

По итогам работ, рассмотренных в данной статье, был разработан алгоритм с использованием вейвлет-преобразований, которые позволили решить поставленные задачи по формированию информационных сигналов, предназначенных для управления интеллектуальными тренажерами.

С помощью разработанной методики были обработаны и проанализированы низкочастотные сигналы коры головного мозга с датчиков электроэнцефалограммы. По итогам вычислительных экспериментов были выявлены и сопоставлены альфа- (α -), бета- (β -), гамма- (γ -) и мю- (μ -) сигналы, а также определена основа для реализации метода обработки управляющих сигналов коры головного мозга с датчиков электроэнцефалограммы [13, 14], связанная со снижением или уменьшением бета- (β -) сигналов с параметрами амплитуды менее 5...15 мкВ с частотой диапазона 10...50 Гц.

Список литературы

1. Лавлинский В. В., Иванова О. Г. Формирование моделей и методов взаимодействия информационных процессов // Приборы и системы. Управление, контроль, диагностика. 2014. № 5. С. 39—50.
2. Бибииков Д. В., Буров Р. Б., Лавлинский В. В., Табаков Ю. Г. Исследование подходов для создания информационной составляющей при проектировании интеллектуального тренажера на основе сигналов коры головного мозга // Моделирование систем и процессов. 2012. № 4. С. 52—56.
3. Бибииков Д. В., Буров Р. Б., Лавлинский В. В., Табаков Ю. Г. Метод проектирования схем для считывания НЧ-сигналов с коры головного мозга // Моделирование систем и процессов. 2013. № 2. С. 11—14.
4. Табаков Ю. Г., Лавлинский В. В. Рационализация выбора математических алгоритмов для управляющих НЧ сигналов // Моделирование систем и процессов. 2014. № 3. С. 39—41.
5. Табаков Ю. Г., Лавлинский В. В., Бибииков Д. В. Оптимизация алгоритмов вейвлет-преобразования при моделировании НЧ сигналов // Моделирование систем и процессов. 2014. № 3. С. 47—49.
6. Бибииков Д. В., Буров Р. Б., Лавлинский В. В., Табаков Ю. Г. Вейвлет-преобразование Добеши для низкочастотных сигналов, снятых с коры головного мозга человека // Моделирование систем и процессов. 2013. № 2. С. 8—11.
7. Табаков Ю. Г., Лавлинский В. В., Бибииков Д. В. Метод и алгоритм обработки НЧ сигналов с помощью вейвлета Добеши // Моделирование систем и процессов. 2014. № 3. С. 42—44.
8. Бибииков Д. В., Лавлинский В. В., Табаков Ю. Г. Модифицированный алгоритм вейвлет-преобразования Морле для анализа НЧ сигналов // Моделирование систем и процессов. 2013. № 3. С. 12—14.
9. Лавлинский В. В., Табаков Ю. Г. Анализ вейвлет-преобразований Добеши и Морле на малейшие изменения в НЧ сигнале // Научный вестник Воронежского государственного архитектурно-строительного университета. Сер. "Информационные технологии в строительных, социальных и экономических системах". 2014. № 2. С. 56—59.
10. Табаков Ю. Г., Бибииков Д. В. Анализ вейвлет-преобразования Морле для снятия и обработки НЧ сигналов // Системы управления и информационные технологии. 2014. № 3.2(57). С. 272—275.
11. Лавлинский В. В. Теоретические предпосылки решения проблем формирования моделей и методов взаимодействия информационных процессов // Моделирование систем и процессов. 2013. № 2. С. 30—36.
12. Лавлинский В. В. Теоретические основы моделирования компонентов для систем автоматизации проектирования электронной базы на основе синтеза виртуальной реальности // Моделирование систем и процессов. 2013. № 3. С. 16—20.
13. Бибииков Д. В., Лавлинский В. В. Применение автокорреляционных методов анализа сигналов с датчиков электроэнцефалограммы для разработки интеллектуального тренажера по восстановлению опорно-двигательных навыков // Моделирование систем и процессов. 2012. № 2. С. 22—26.
14. Табаков Ю. Г., Лавлинский В. В., Бибииков Д. В. Обработка НЧ сигналов для интеллектуальных тренажеров с применением программных линейных фильтров с дискретным временем // Моделирование систем и процессов. 2014. № 3. С. 45—47.

Yu. G. Tabakov, Graduate Student, e-mail: vurik204@rambler.ru
Voronezh State Academy of Forestry Engineering, Voronezh

Problems Processing LF Signals in of Intellectual Information Systems

The article describes and shows the basic problems in the processing of low-frequency signals removed from the cerebral cortex and possible solutions. Proposed a specially designed mathematical models and algorithms for processing low-frequency signals. These model and algorithms constructed on a specially designed and modified Daubechies and Morlaix wavelet-transform, which allow processing low-frequency signals in real time and analyze the data obtained. Present detailed block diagram of the low-frequency signal processing module, which demonstrates the work of the mathematical model. Present a block diagram of the developed mathematical algorithm demonstrating phased operation of processing from low-frequency signal captured from human cerebral cortex. Obtained in the course research results, as well as the developed mathematical models and algorithms can be used in specialized medical institutions dealing with problems rehabilitation of the musculoskeletal system of the person. The purpose of such research — process and analyze low frequency signals taken from the human cerebral cortex, and the data obtained reveal the control signals for intellectual simulators to restore the musculoskeletal system of the person.

Keywords: signal analysis, signal processing, low-frequency signals, programming, intellectual simulator, Daubechies wavelet, Morlaix wavelet, wavelet transform, mathematical model

References

1. **Lavlinskiy V. V., Ivanov O. G.** Formirovanie modelej i metodov vzaimodejstviya informacionnyh processov. *Pribory i sistemy. Upravlenir, kontrol', diagnostika*. 2014. N 5. P. 39—50.
2. **Bibikov D. V., Burov R. B., Lavlinskiy V. V., Tabakov Yu. G.** Issledovanie podhodov dlja sozdaniya informacionnoj sostavljajushhej pri proektirovanii intellektual'nogo trenazhera na osnove signalov kory golovnogogo mozga. *Modelirovanie sistem i processov*. 2012. N. 4. P. 52—56.
3. **Bibikov D. V., Burov R. B., Lavlinskiy V. V., Tabakov Yu. G.** Metod proektirovanija shem dlja schityvanija NCh-signalov s kory golovnogogo mozga. *Modelirovanie sistem i processov*. 2013. N. 2. P. 11—14.
4. **Tabakov Yu. G., Lavlinskiy V. V.** Racionalizacija vybora matematicheskikh algoritmov dlja upravliaiushhih NCh signalov. *Modelirovanie sistem i processov*. 2014. N. 3. P. 39—41.
5. **Tabakov Yu. G., Lavlinskiy V. V., Bibikov D. V.** Optimizacija algoritmov vejljet-preobrazovanija pri modelirovanii NCh signalov. *Modelirovanie sistem i processov*. 2014. N. 3. P. 47—49.
6. **Bibikov D. V., Burov R. B., Lavlinskiy V. V., Tabakov Yu. G.** Vejljet-preobrazovanie Dobeshi dlja nizkochastotnyh signalov, snjatyh s kory golovnogogo mozga cheloveka. *Modelirovanie sistem i processov*. 2013. N. 2. P. 8—11.
7. **Tabakov Yu. G., Lavlinskiy V. V., Bibikov D. V.** Metod i algoritm obrabotki NCh signalov s pomoshh'ju vejljeta Dobeshi. *Modelirovanie sistem i processov*. 2014. N. 3. P. 42—44.
8. **Bibikov D. V., Lavlinskiy V. V., Tabakov Yu. G.** Modificirovannyj algoritm vejljet-preobrazovanija Morle dlja analiza NCh signalov. *Modelirovanie sistem i processov*. 2013. N. 3. P. 12—14.
9. **Lavlinskiy V. V., Tabakov Yu. G.** Analiz vejljet-preobrazovanij Dobeshi i Morle na malejschie izmenenija v NCh signale. *Nauchnyj vestnik Voronezhskogo gosudarstvennogo arhitekturno-stroitel'nogo universiteta. Ser. "Informacionnye tehnologii v stroitel'nyh, social'nyh i jeekonomicheskikh sistemah"*. 2014. N. 2. P. 56—59.
10. **Tabakov Yu. C., Bibikov D. V.** Analiz vejljet-preobrazovanija Morle dlja snjatija i obrabotki NCh signalov. *Sistemy upravlenija i informacionnye tehnologii*. 2014. N. 3.2(57). P. 272—275.
11. **Lavlinskiy V. V.** Teoreticheskie predposylki reshenja problem formirovanija modelej i metodov vzaimodejstviya informacionnyh processov. *Modelirovanie sistem i processov*. 2013. N. 2. P. 30—36.
12. **Lavlinskiy V. V.** Teoreticheskie osnovy modelirovanija komponentov dlja sistem avtomatizacii proektirovanija jelektronnoj bazy na osnove sinteza virtual'noi real'nosti. *Modelirovanie sistem i processov*. 2013. N. 3. P. 16—20.
13. **Bibikov D. V., Lavlinskiy V. V.** Primenenie avtokorreljacionnyh metodov analiza signalov s datchikov jelektrojencefalogrammy dlja razrabotki intellektual'nogo trenazhera po vosstanovlenju oporno-dvigatel'nyh navykov. *Modelirovanie sistem i processov*. 2012. N. 2. P. 22—26.
14. **Tabakov Yu. G., Lavlinskiy V. V., Bibikov D. V.** Obrabotka NCh signalov dlja intellektual'nyh trenazherov s primeneniem programmyh linejnyh fil'trov s diskretnym vremenem. *Modelirovanie sistem i processov*. 2014. N. 3. P. 45—47.

ИНФОРМАЦИЯ

Институт Математики им. С. Л. Соболева СО РАН
Российский Фонд Фундаментальных Исследований
Российская Ассоциация Распознавания Образов и Анализа Изображений

V Всероссийская конференция
"ЗНАНИЯ — ОНТОЛОГИИ — ТЕОРИИ"

с международным участием

6—8 октября 2015 г.
Новосибирск

Конференция посвящена математическим методам представления данных, извлечения знаний и построения теорий предметных областей, анализу формальных понятий, а также методам извлечения информации из текстов естественного языка. Лучшие по итогам конференции доклады будут опубликованы в журналах «Информационные технологии» и «Программная инженерия» (из списка ВАК).

Тематика Конференции отражает основные стадии процесса познания:

— **Обнаружение закономерностей и извлечение знаний**, скрытых в структурированных и неструктурированных данных. Машинное обучение. Распознавание образов. Прогнозирование. Индуктивный вывод.

— **Систематизация знаний**. Инженерия знаний. Управление знаниями. Разработка онтологий предметных областей: технологии создания и применения онтологий.

— **Построение теорий предметных областей**. Анализ формальных понятий. Логическая семантика естественного языка. Нечеткие логики.

Работа конференции планируется в виде пленарных, секционных и стендовых докладов и круглых столов по тематике конференции. Рабочие языки конференции — русский и английский.

Контактные данные для переписки:

zont@math.nsc.ru

Сайт конференции:

<http://math.nsc.ru/conference/zont/15>

ЖУРНАЛ В ЖУРНАЛЕ



**НЕЙРОСЕТЕВЫЕ
ТЕХНОЛОГИИ**

№ 6

ИЮНЬ

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.

Аникин В. И., Карманова А. А.

Кластеризация и классификация многомерных данных
клеточной нейронной сетью Кохонена 470

Усков А. А., Жукова А. Г.

Нейросетевая оценка сложности IDEF-диаграмм 478

В. И. Аникин, д-р техн. наук, проф., e-mail: anikin_vi@mail.ru,
Поволжский государственный университет сервиса, г. Тольятти,
А. А. Карманова, инженер-программист, e-mail: turaeva.alexandra@mail.com,
ООО "НетКрэкер", г. Тольятти

Кластеризация и классификация многомерных данных клеточной нейронной сетью Кохонена

Показана перспективность и высокая временная эффективность кластеризации и классификации многомерных выборок данных нейронной сетью Кохонена, обучаемой клеточным автоматом. Продемонстрировано полезное применение краевого эффекта и многосвязных самоорганизующихся карт Кохонена для решения проблемы "мертвых" нейронов и надежного выделения границ группировки кластеров в пространстве с линейно и/или нелинейно разделимыми классами учебных образцов.

Ключевые слова: нейронная сеть Кохонена, клеточный автомат, Excel, классификация многомерных данных, визуализация, U -матрица, P -матрица

Введение

Ввиду огромного количества информации, которая, согласно оценкам аналитиков, удваивается каждые 2—3 года, лишь очень малая ее часть будет когда-либо увидена человеческим глазом. Единственная возможность понять и найти что-то полезное в ней — использовать методы *Data Mining (Knowledge Discovery in Data — KDD)*. Технология KDD позволяет решать задачи многомерной кластеризации, регрессии, поиска ассоциаций, классификации и используется во многих областях с большим объемом данных — астрономии, биологии, медицине, телекоммуникациях, банковском деле, промышленном производстве и т. д. Важная и быстрорастущая часть KDD — анализ связей между данными, имеющий приложения в биоинформатике, поисковых системах и др.

Существующие парадигмы исследовательского анализа многомерных данных базируются на трех аспектах: 1) предобработка в целях приведения данных к форме, удобной для последующего анализа; 2) кластеризация данных, т. е. их группировка в физически осмысленные наборы/множества; 3) визуализация полученных результатов таким образом, чтобы сходство и различие данных было хорошо видно.

Достоинством самоорганизующихся карт Кохонена (*Self-Organizing Map — SOM*) является то, что они позволяют решать задачу анализа данных в едином процессе, объединяющем все три перечисленных аспекта [1]. Действительно, самоорганизующаяся карта Кохонена выполняет нелинейное отображение многомерных данных на одно-, двух- или трехмерную решетку и поэтому представляет собой эффективный инструмент визуализации связей между данными. Метод SOM применим к любым данным, которые можно представить векторами их свойств.

В ранних работах по методу SOM использовали для классификации небольшие одномерные и дву-

мерные нейронные сети (НС) Кохонена в режиме "один нейрон на один кластер/класс" (*k-means SOM*) [1]. Если число узлов SOM отличается от ожидаемого числа классов, то требуются группировка или разделение кластеров в целях выделения необходимого числа классов.

Важным методом группирования кластеров в методе SOM является визуализация пространственного распределения нейронов с помощью U -матрицы Ултша (визуализация, основанная на расстояниях) вида, показанного в табл. 1 [2], где используются четыре типа расстояний от центра нейрона u_{ij} до его ближайших соседей: $dx(i, j) = d(u_{ij}, u_{i+1, j})$, $dy(i, j) = d(u_{ij}, u_{i, j+1})$, $dxy(i, j) = d(u_{ij}, u_{i+1, j+1})$, $dyx(i, j) = d(u_{i+1, j}, u_{i, j+1})$ и $dz(i, j) = 0,5(dxy(i, j) + dyx(i, j))$. Элементы $du(i, j)$ матрицы могут быть произвольными. В нашем исследовании значения этих последних элементов равны числу учебных образцов, попавших в окрестность Вороного нейрона u_{ij} ; в совокупности эти элементы образуют так называемую H -матрицу. Достоинством табличной реализации UH -матрицы является то, что инструмент условного форматирования Excel дает возможность визуализировать границы между классами учебных образцов интерактивно, путем варьирования значения порогового расстояния, отделяющего соседние классы друг от друга.

Другим методом группирования кластеров, также предложенным Ултшем [3], является метод P -матрицы. P -матрица имеет размерность U -матрицы, а значения ее элементов равны числу учебных образцов внутри гиперсфер заданного радиуса, проведенных вокруг нейронов и промежуточных аппроксимирующих точек. В дополнение к информации о расстояниях, предоставляемой U -матрицей, P -матрица содержит информацию о плотности распределения учебных образцов во входном пространстве (визуализация, основанная на плотности).

Таблица 1

U -матрица	...	$2j - 1$	$2j$	$2j + 1$...
...
$2i - 1$...	$dz(i - 1, j - 1)$	$dy(i - 1, j)$	$dz(i - 1, j)$...
$2i$...	$dx(i, j - 1)$	$du(i, j)$	$dx(i, j)$...
$2i + 1$...	$dz(i, j - 1)$	$dy(i, j)$	$dz(i, j)$...
...

Таблица 2

P -матрица	...	$2j - 1$	$2j$	$2j + 1$...
...
$2i - 1$...	$w_{cp}(i - 1, j - 1)$	$w_{cp}(i - 1, j)$	$w_{cp}(i - 1, j)$...
$2i$...	$w_{cp}(i, j - 1)$	$w(i, j)$	$w_{cp}(i, j)$...
$2i + 1$...	$w_{cp}(i, j - 1)$	$w_{cp}(i, j)$	$w_{cp}(i, j)$...
...

В нашем исследовании координаты аппроксимирующих точек для P -матрицы определялись как показано в табл. 2, где $w_{cp}(i, j \pm 1)$ — среднее значение одноименных координат соседних нейронов по горизонтали SOM; $w_{cp}(i \pm 1, j)$ — по вертикали; $w_{cp}(i \pm 1, j \pm 1)$ — по диагоналям.

Цель нашего исследования — повышение эффективности алгоритма обучения и качества классификации, достигаемые за счет использования клеточной НС Кохонена с многосвязной SOM. Клеточная нейронная сеть Кохонена всегда обучается на максимальной скорости в пакетном режиме, что позволяет сократить время обучения в несколько десятков раз по сравнению с классическим алгоритмом обучения [1]. О насущной необходимости повышения временной эффективности алгоритма обучения НС Кохонена классическим алгоритмом говорят, например, условия эксперимента по классификации генов, выполненных в эффективной работе [4]: размерность данных — 79, число генов в выборке — 2460, число итераций обучения — $3,5 \cdot 10^6$, время обучения на рабочей станции с процессором Alpha — 19 ч, число реализаций опыта — практически неограниченно.

Обоснование и постановка задачи исследования

Важной особенностью данного исследования в сравнении с работой [5] является то, что здесь оптимизирован алгоритм адаптивного саморазворачивания нейронной сети, выполняемый в первых эпохах обучения. Концептуально главная идея описанного в работе [5] алгоритма саморазворачивания сети — постепенное втягивание активными нейронами неактивных нейронов в нормированный гиперкуб пространства учебных образцов, — осталась неизменной, но использованный в этой работе алгоритм саморазворачивания претерпел три изменения.

1. В первой эпохе обучения в качестве нейрона-победителя выбирается центральный, а не угловой

нейрон SOM, благодаря чему число эпох i_p саморазворачивания сети по сравнению с алгоритмом в работе [5] сократилось вдвое и равно $i_p = \lceil (L - 1)/2 \rceil$, где L — длина большей стороны прямоугольной SOM, скобки $\lceil \rceil$ означают округление вверх до ближайшего целого.

2. Мультипликативная поправка (dw_x, dw_y) к весам соседей нейрона-победителя вычисляется по формуле $(dw_x, dw_y) = 1 - \alpha(i, j)$, учитывающей структуру обучающего клеточного автомата (КА) Мура, где (i, j) — смещение вновь активизируемых нейронов относительно их активных соседей в плоскости саморазворачивания сети xu ; $\alpha = 10^{-3} \dots 10^{-4}$ — масштабный коэффициент. Относительные смещения (i, j) активизируемых нейронов определяются из простых соображений: в первой эпохе обучения (рис. 1) центральный нейрон-победитель с координатами $(0, 0)$ втянет 8 нейронов из окружения радиуса $r = 1$ в точки плоскости xu с относительными смещениями $(-1, -1), (0, -1), (1, -1), (-1, 0), (1, 0), (-1, 1), (0, 1), (1, 1)$, во второй эпохе обучения эти 8 нейронов, в свою очередь, втянут в нормированный гиперкуб пространства образцов своих соседей из окружения радиуса $r = 2$ и поместят их в точки со смещениями $(-2, -2), (-2, -1), \dots, (2, 1), (2, 2)$ и т. д.

3. К двум статическим типам связей между нейронами (\oplus — связь включена, \ominus — связь выключена), определенным в работе [5], добавлен еще один, адаптивный тип связи [6]: \boxplus — связь включена в эпохах саморазворачивания сети и автоматически отключается в последующих эпохах обучения. Благодаря этому типу связей саморазворачивание сети всегда выполняется клеточным автоматом Мура, причем во всем входном пространстве учебных образцов, рассматриваемом как единое целое, кроме того упрощается процедура начальной инициализации клеточных НС Кохонена с многосвязными SOM [5].

Выбор плоскости xu , в которой происходит саморазворачивание сети, неоднозначен (всего имеется $d(d - 1)/2$ таких плоскостей, где d — размерность пространства учебных образцов) и оказывает влияние на результат кластеризации данных методом SOM. Очевидно, что оптимальной, но не необходимой, является плоскость двух главных компонент [7].



Рис. 1. Адаптивная активация нейронов за три эпохи саморазворачивания клеточной НС Кохонена

В описываемых ниже экспериментах по кластеризации и классификации данных клеточной НС Кохонена использовались тестовые моделируемые компьютером и взятые из архива *UCI Machine Learning Repository* экспериментальные выборки многомерных данных:

1. Моделируемая выборка данных объемом 2000 элементов с равномерным случайным распределением учебных образцов в двух соприкасающихся углами кубах; трехмерный вариант выборки, предложенной Ултшем [3] для обоснования необходимости введения P -матрицы.

2. Моделируемая трехмерная случайная выборка данных Chainlink объемом 1000 элементов, предложена Ултшем [2] для обоснования необходимости введения U -матрицы.

3. 13-мерная экспериментальная выборка данных *Wine recognition* объемом 178 элементов для классификации производителей вина: <http://mlr.cs.umass.edu/ml/datasets/Wine>.

4. Трехмерная экспериментальная выборка данных Haberman's Survival объемом 306 элементов для классификации послеоперационной выживаемости больных раком: URL: <https://archive.ics.uci.edu/ml/datasets/Haberman's+Survival>.

Кластеризацию элементов тестовых выборок данных и их классификацию методом группирования проводили посредством созданной в Excel табличной модели клеточной НС Кохонена. Визуализацию результатов кластеризации выполняли с использованием встроенных средств форматирования Excel, она включала автоматическое построение по результатам кластеризации интерактивных UH - и P -матриц, координатных C -карт, карты классов учебных образцов (при обучении с учителем).

Метод решения задачи исследования

Для решения поставленной задачи исследования использовали две клеточные НС Кохонена, реализованные в виде итерационных табличных моделей Excel [8].

1. Число нейронов в сети — 49, размер прямоугольной SOM — 7×7 .

2. Число нейронов в сети — 400, размер прямоугольной SOM — 20×20 .

Число нейронов в каждой сети и размеры SOM можно произвольно изменять в сторону уменьшения путем деактивации некоторых нейронов.

Заметим, что использование табличных моделей НС Кохонена двух размеров не является принципиально необходимым, но позволяет уменьшить время, затрачиваемое на кластеризацию многомерных выборок данных в Excel. Поэтому на обучающих выборках данных небольшого размера, таких как выборки Фишера [5] и Хабермана, использовали табличную модель НС Кохонена размером 49 нейронов, а на выборках большого размера (объемом в тысячи образцов) — размером 400 нейронов (размер сети выбирается из соображений,

чтобы каждый кластер в среднем содержал единицы—десятки учебных образцов).

Обе табличные модели клеточной НС Кохонена были реализованы одинаково на пяти рабочих листах Excel: "Выборки", "Инициализация", "Обучение", "Кластеризация", "Визуализация".

Рабочий лист "Выборки" содержит приведенные выше многомерные выборки данных, нормированные к интервалу $[-1, 1]$ по каждой числовой координате, а также интерфейс активации, используемой в данный момент выборки.

Рабочий лист "Инициализация" предназначен для задания начальной конфигурации клеточной НС Кохонена, позволяя пользователю определить геометрию нейронной сети и поперечные размеры прямоугольной SOM, явно указать нейрон, который станет победителем в первой эпохе обучения. На этом же рабочем листе выполняются расчеты мультипликативных поправок (dw_x, dw_y) к весам нейронов в эпохах саморазворачивания сети и вычисляются координаты случайной точки на гиперсфере начальной инициализации весов нейронов [5].

На рабочем листе "Обучение" табличными средствами реализован описанный в работе [5] оригинальный алгоритм обучения клеточной НС Кохонена. Пользовательский интерфейс этого листа позволяет в визуальном режиме определять параметры обучения (число эпох саморазворачивания, взаимодействия, дообучения по алгоритму WTA), разрывать/активировать связи между нейронами, запускать/останавливать итерации обучения.

На рабочем листе "Кластеризация" реализован алгоритм выполнения одной эпохи подачи учебных образцов на входы обученной НС в целях их подразделения на кластеры и вычисления элементов P -матрицы для заданного радиуса гиперсфер, окружающих нейроны и промежуточные точки.

Наконец, рабочий лист "Визуализация" содержит интерактивные средства визуализации результатов кластеризации. Используя встроенные в Excel формулы и средства условного форматирования, удается эффективно визуализировать UH - и P -матрицы, координатные C -карты, карты классов образцов (при обучении с учителем), а также интерактивно выполнять разнообразную пост-обработку полученных результатов.

С нашей точки зрения, именно погруженность клеточной НС Кохонена в мощную вычислительную и аналитическую среду Excel является главным достоинством данного экспериментального исследования, открывая перед исследователем потрясающие возможности для креативного творчества и визуального экспериментирования с алгоритмом вычислений, входными, выходными и промежуточными данными.

Работа с табличной моделью клеточной НС Кохонена включает следующие шаги (рис. 2).

1. В зависимости от обучающей выборки данных выбирается один из двух файлов Excel: для вы-

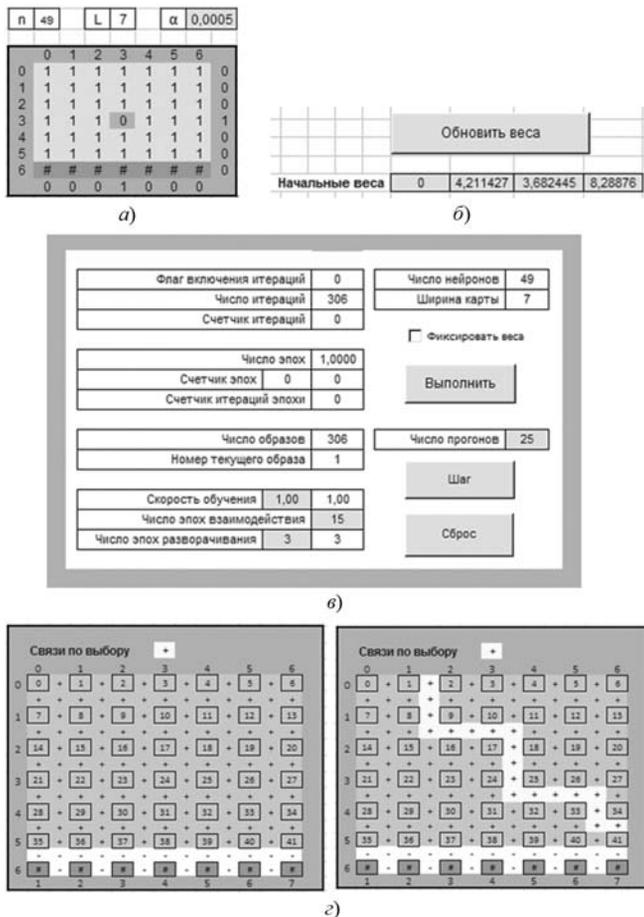


Рис. 2. Пользовательский интерфейс табличной модели НС Кохонена с прямоугольной SOM размером 6 × 7 нейронов: а — задание структуры сети; б — выбор случайной точки на гиперсфере инициализации весов нейронов; в — пользовательский интерфейс алгоритма обучения НС; г — типы связей между нейронами SOM: слева — конфигурация Мура, справа — двухсвязная конфигурация

борок объемом до 500—1000 образцов — табличная модель клеточной НС Кохонена размером 7 × 7 нейронов, для выборок большего объема — размером 20 × 20 нейронов.

2. С помощью пользовательского интерфейса, показанного на рис. 2, а, где активные нейроны помечены символом 1, нейрон-победитель — символом 0, недействующие нейроны — символом #, определяется структура клеточной НС Кохонена, указывается нейрон-победитель в первой эпохе обучения, задается значение масштабного коэффициента α .

3. Щелчком на кнопке Обновить веса выбирается случайная начальная точка на гиперсфере инициализации начальных весов нейронов (рис. 2, б).

4. На рабочем листе Обучение задаются параметры обучения: скорость обучения; число эпох саморазворачивания сети; взаимодействия, обучения/прогонов (рис. 2, в).

5. Указываются типы связей между нейронами НС Кохонена в эпохах взаимодействия (рис. 2, г).

6. Запускается процесс обучения сети (кнопка Выполнить на рис. 2, в).

7. На рабочем листе Кластеризация выполняется эпоха кластеризации элементов выборки данных обученной НС Кохонена.

8. На рабочем листе Визуализация выполняется интерактивный анализ результатов кластеризации, в случае необходимости изменяются структура и параметры нейронной сети, и шаги 3—8 повторяются заново.

Результаты экспериментального исследования

Действенность описанного алгоритма адаптивного саморазворачивания клеточной НС Кохонена продемонстрируем на примере двумерной случайной выборки данных с круговой симметрией объемом 4000 учебных образцов (рис. 3).

Видно, что вновь активизируемые нейроны втягиваются в нормированное пространство учебных образцов латерально, без скручивания сети. Краевой эффект, обычно считающийся паразитным, теперь играет положительную роль, оставляя на периферии пространства учебных образцов свободное место для добавления следующих активизируемых нейронов.

Эксперименты по кластеризации и классификации элементов тестовых выборок данных состояли в следующем. Сначала НС Кохонена с однослойной SOM обучалась клеточным автоматом Мура (рис. 4, а), выполнялась кластеризация элементов обучающей выборки и делалась попытка группировки полученных кластеров. Для этого, варьируя порог группирования, посредством UH - и P -матриц наблюдали изменения вида границы между клас-

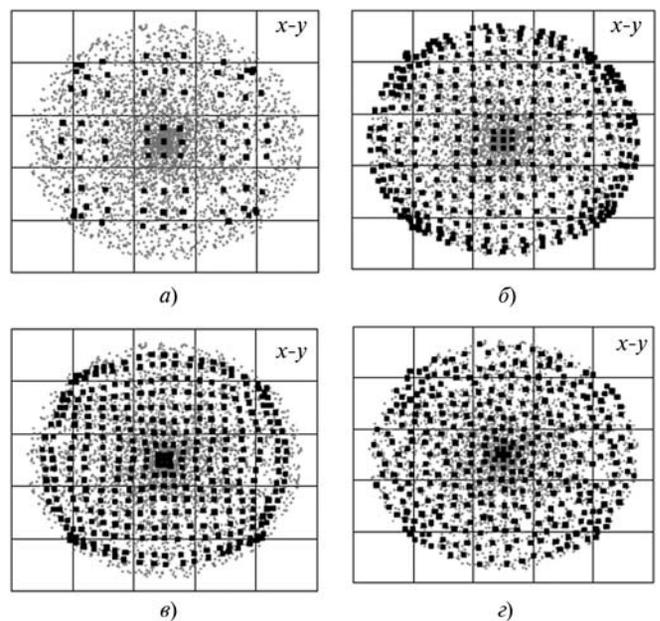


Рис. 3. Обучение нейронной сети Кохонена с прямоугольной SOM размером 19 × 19 нейронов: а — 4 эпохи саморазворачивания; б — 9 эпох саморазворачивания сети; в — 15 эпох взаимодействия; г — 20 эпох обучения

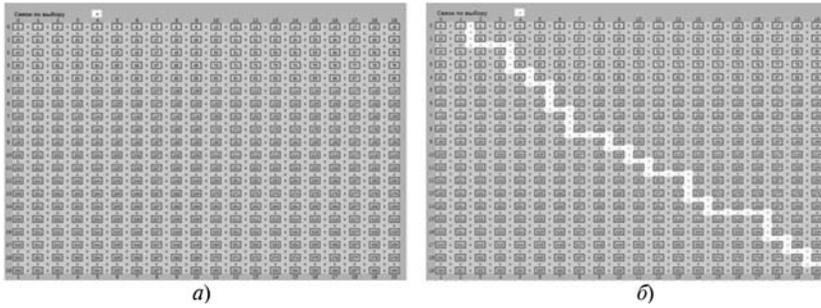


Рис. 4. Обучающие клеточные автоматы для случайной выборки данных 1

сами (обычно она выглядит "размытой", нечетко выраженной).

Затем на базе полученных результатов группирования, односвязная SOM разрезалась на несколько областей вдоль предполагаемых границ классов (рис. 4, б), и НС Кохонена с полученной многосвязной прямоугольной SOM обучалась вторично. Для достижения более качественной группировки, операцию разрезания карты SOM иногда приходилось выполнять несколько раз. После завершения процесса обучения снова выполнялись кластеризация элементов обучающей выборки и группировка кластеров.

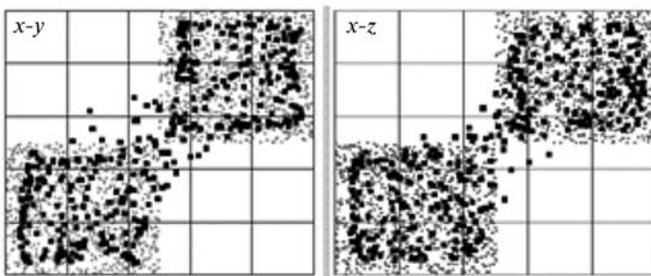
Физические закономерности кластеризации и классификации многомерных данных исследовались на приведенных выше тестовых выборках данных 1 и 2.

Примечательной особенностью тестовой обучающей выборки данных 1 является то, что два класса ее образцов пространственно линейно разделимы, но у соприкасающихся вершин кубов образцы, принадлежащие разным классам, отстоят друг от друга на небольшом расстоянии, и при кластеризации могут быть ошибочно отнесены к другому классу.

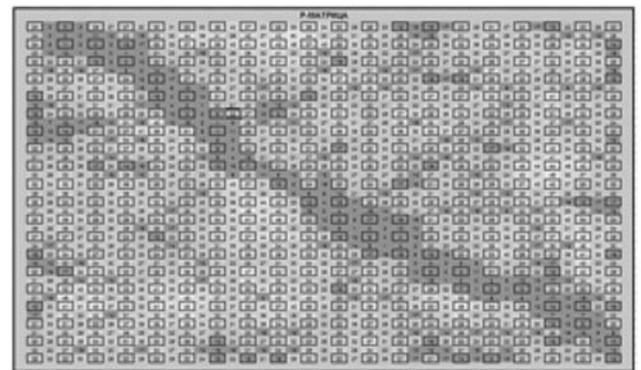
Параметры обучения НС Кохонена выборкой данных 1: число нейронов в сети — 400; размер SOM — 20×20 , начальные веса нейронов — $(0; 7,754; -6,273; 0,727)$, нейрон-победитель в первой эпохе обучения — 189; начальные веса нейрона-победителя — $(0; 0; 0; 0)$, число эпох саморазворачивания сети — 10; число эпох взаимодействия — 30; всего эпох обучения — 40.

Результаты классификации элементов выборки 1 при заданных параметрах обучения позволяют сделать ряд важных выводов (рис. 5, б).

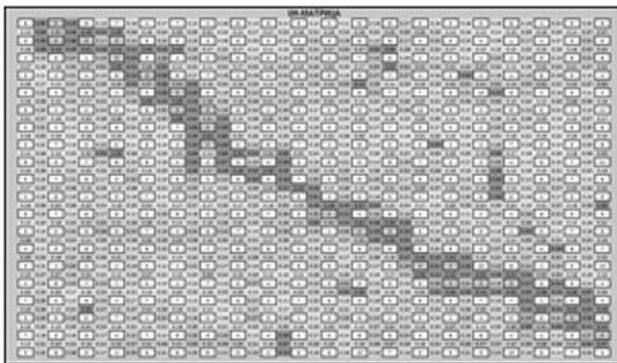
При классификации элементов выборки данных клеточной НС Кохонена с односвязной SOM наблюдаются около 20 "мертвых" нейронов, не победивших ни разу (рис. 5, а слева). Появление "мертвых" нейронов обусловлено тем, что смежные с ними нейроны относятся к разным классам, поэтому в эпохах взаимодействия эти "мертвые" нейроны, сами находясь вне области концентрации учебных образцов, благодаря связям со своими со-



а)



б)



в)

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0	2.00	0.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
1	2.00	0.00	0.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
2	2.00	2.00	2.00	0.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
3	2.00	2.00	2.00	2.00	0.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
4	2.00	2.00	2.00	2.00	2.00	0.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
5	2.00	2.00	2.00	2.00	2.00	2.00	0.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
6	2.00	2.00	2.00	2.00	2.00	2.00	2.00	0.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
7	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	0.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
8	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	1.80	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
9	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	0.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
10	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	0.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
11	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	0.00	1.00	1.00	1.00	1.00	1.00	1.00
12	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	0.00	1.00	1.00	1.00	1.00	1.00
13	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	0.00	1.00	1.00	1.00	1.00
14	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	0.00	1.00	1.00	1.00
15	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	0.00	1.00	1.00
16	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	0.00	1.00
17	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	0.00
18	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00
19	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00

г)

Рис. 5. Результаты кластеризации элементов выборки данных 1 после 30 эпох обучения, односвязная SOM: а — 30 эпох обучения; б — UH-матрица: порог = 0,32; в — P-матрица: $R = 0,2$, порог = 20; г — карта классов

седами поочередно подтягиваются к победившим образцам то одного, то другого класса. При использовании двухсвязной SOM связи между нейронами, принадлежащими разным классам, принудительно разрываются, и появление "мертвых" нейронов исключается (рис. 6, а).

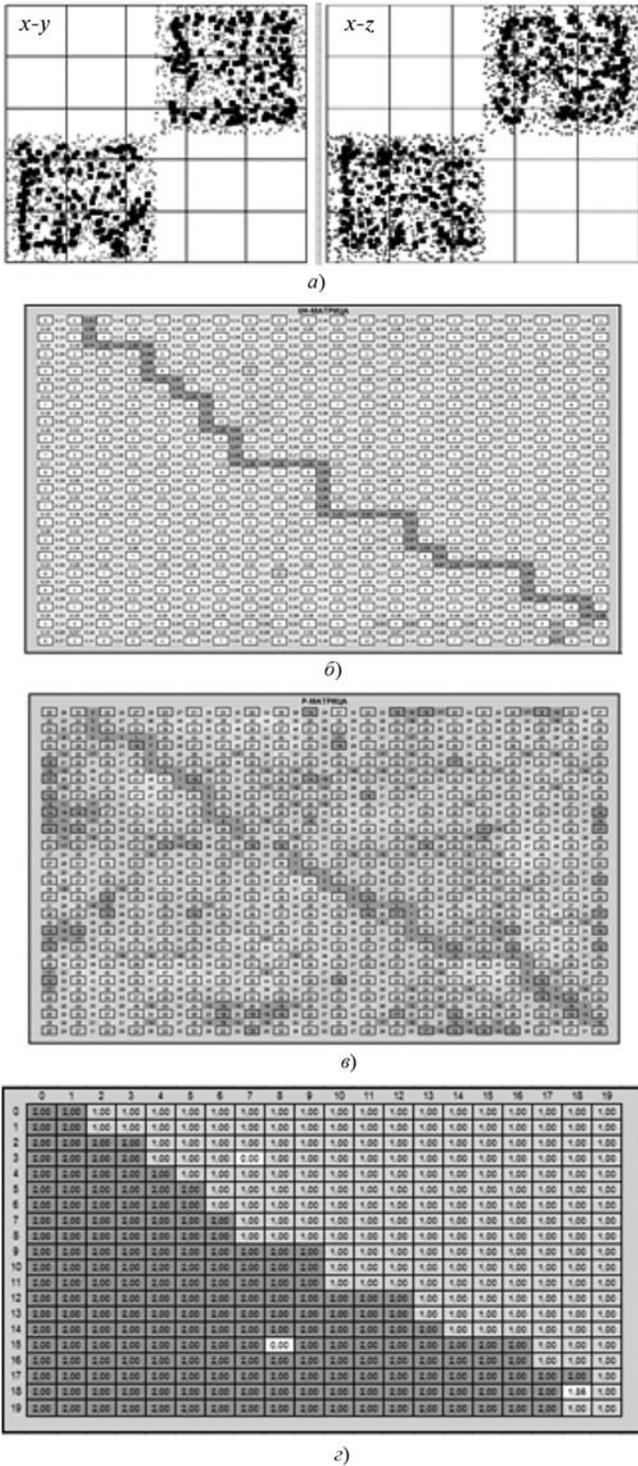


Рис. 6. Результаты кластеризации элементов выборки данных 1 после 30 эпох обучения, двухсвязная SOM: а — 30 эпох обучения; б — UH -матрица: порог = 0,62; в — P -матрица: $R = 0,2$, порог = 20; г — карта классов

Краевой эффект, с которым обычно борются путем задания периодических граничных условий на краях SOM, наблюдается и у границ пространственно разделимых классов образцов. На самом деле этот эффект внутренне присущ самому методу SOM, и избавиться от него невозможно, пока существуют латеральные локальные связи между нейронами. Единственный корректный механизм борьбы с ним — это дообучение (после завершения эпох взаимодействия) нейронной сети Кохонена по алгоритму WTA.

В случае многосвязной SOM улучшается разделение учебных образцов на классы посредством UH -матрицы, граница между классами становится четко определенной (рис. 6, б), повышается порог группирования, что равносильно увеличению расстояния между граничными элементами разных классов. Положительный эффект от использования многосвязной SOM в случае P -матрицы менее заметен (рис. 5, в, б, в).

Аналогичные закономерности кластеризации и классификации данных клеточной НС Кохонена выявляют эксперименты с тестовой обучающей выборкой 2. Элементы этой выборки распределены в трехмерном пространстве так, что они образуют два тонких сцепленных друг с другом подобно олимпийским кольцам тора, один из которых лежит в плоскости xy , второй — в плоскости xz (рис. 7, а).

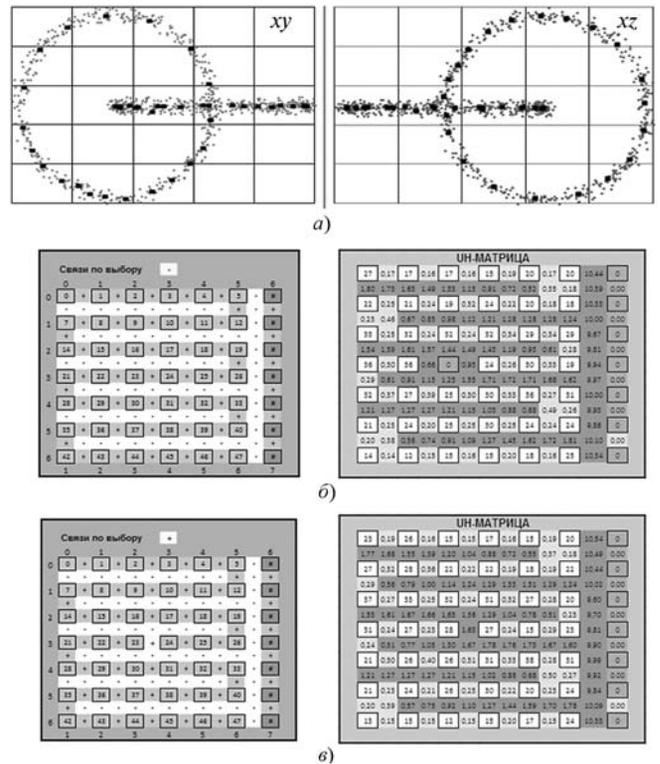


Рис. 7. Результаты кластеризации элементов тестовой выборки данных 2 после 40 эпох обучения: а — проекции координат образцов на плоскости xy и xz ; б — односвязная линейная SOM и UH -матрица при пороге 0,50; в — двухсвязная линейная SOM и UH -матрица при пороге 0,50

Отличительной особенностью данной выборки является то, что ее элементы линейно не разделимы в пространстве, хотя и образуют два явно выраженных класса.

Попытка классифицировать элементы выборки данных двухклеточной НС Кохонена с двумерной SOM к успеху не привела, и нам пришлось воспользоваться НС Кохонена с линейной SOM (рис. 7, б). Это означает, что структура нейронной сети, применяемой для классификации многомерных данных, зависит от их пространственного распределения, поэтому в общем случае успешное практическое применение метода SOM предполагает наличие предварительных знаний об этом распределении.

Следует также отметить, что число эпох саморазворачивания НС Кохонена с линейной SOM довольно большое и равно половине длины последней. Это позволяет наглядно наблюдать, как постепен-

но, по мере выполнения эпох саморазворачивания, сеть начинает все лучше понимать пространственную структуру данных. По завершении процесса обучения нейроны распределяются в пространстве образов так, как показано на рис. 7, а.

Как и в экспериментах с выборкой данных 1, разрыв связи между нейронами, расположенными у границы двух классов, устраняет причину появления "мертвых" нейронов, граница между классами на *UH*-матрице становится четко выделенной, заметно повышается порог группирования (рис. 7, в).

Такие же закономерности наблюдаются и при классификации образов экспериментальной выборки данных *Wine recognition*, только в этом случае прямоугольную SOM приходится разрезать на три области по числу ожидаемых классов (рис. 8).

Форма контура разрезания определяется из вида *UH*- и *P*-матриц, а также карты классов. Метки на карте классов показывают, образцы каких классов содержатся в каждом кластере, например, метка 0/2/3 означает, что в соответствующем кластере имеются учебные образцы второго и третьего классов.

Заметим также, что, как показывает более внимательный анализ *UH*-матрицы, образцы класса 2 делятся по средней вертикальной линии (рис. 8, б) на два вложенных субкласса. О существовании этих субклассов в описании экспериментальной выборки данных *Wine Recognition* ничего не говорится.

Экспериментальная выборка данных Хабермана является особой и отличается от предыдущих тестовых выборок тем, что классифицировать ее элементы методом группирования кластеров не удастся. Действительно, варьирование значений порогов группирования для *UH*- и *P*-матриц этой выборки не позволяет выявить четкую границу между классами больных, выживших и умерших от рака. Причиной этому является то, что элементы выборки данных Хабермана не являются пространственно разделимыми (ни линейно, ни нелинейно).

Решить задачу классификации элементов этой выборки данных клеточной НС Кохонена можно, тем не менее, методом обучения с учителем. На рис. 9 показана карта классов, полученная после обучения этой выборкой клеточной НС Кохонена с прямоугольной SOM размером 6 × 7 нейронов при следующих параметрах: начальные веса нейронов — (0; 1,547; 6,117; 7,512); координаты нейрона-победителя 24 — (0; 0; 0); эпох разворачивания сети — 3; эпох взаимодействия — 15; эпох обучения — 25.

Светлой заливкой на карте выделены нейроны с долей больных, которые умерли от рака в течение пяти лет после операции, не превышающей 25 %.

Заключение

Проведенные экспериментальные исследования показали перспективность и высокую временную эффективность авторского алгоритма кластеризации

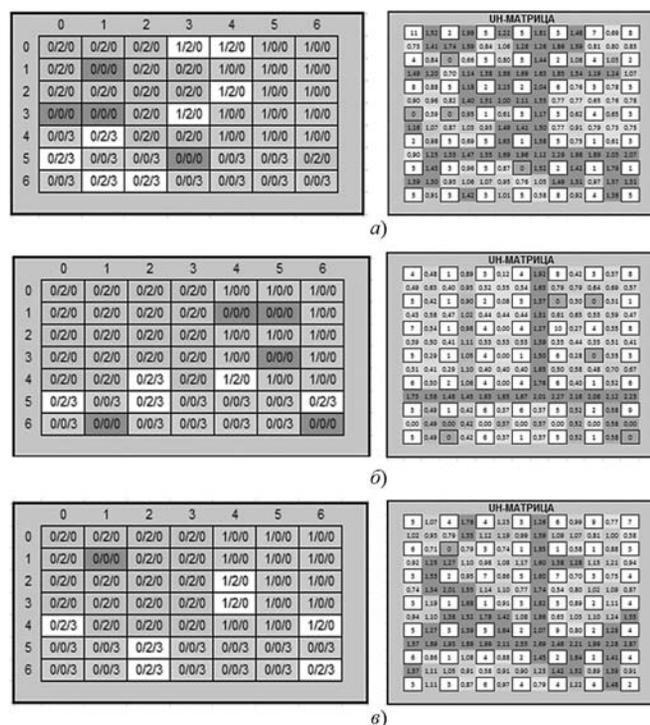


Рис. 8. Результаты кластеризации элементов экспериментальной выборки данных *Wine recognition*: а — односвязная SOM: 40 эпох обучения, порог = 1,25; б — трехсвязная SOM, 30 эпох взаимодействия, порог = 1,25; в — трехсвязная SOM: 40 эпох обучения, порог = 1,25

С	0	1	2	3	4	5	6
0	1,42	1,11	1,50	1,23	1,09	1,20	1,50
1	1,33	1,11	1,00	1,00	1,00	1,33	2,00
2	1,33	1,17	1,11	1,25	1,00	1,67	1,57
3	1,43	1,07	1,57	1,20	1,21	1,67	1,50
4	1,33	1,67	1,50	1,11	1,50	1,33	1,50
5	1,09	1,00	1,11	1,38	1,25	1,20	1,50
6	0	0	0	0	0	0	0

Рис. 9. Карта классов выборки данных Хабермана: порог выделения — 1,25

и классификации многомерных выборок данных клеточной НС Кохонена с многосвязной SOM. Исследования проводили с использованием итерационной табличной модели этой сети, реализованной в Excel чисто табличными средствами, без написания программного кода VBA. Временная сложность алгоритма обучения сети равняется $T = O(n \cdot N \cdot d \cdot i)$, где n — число нейронов в сети; N — объем обучающей выборки; d — размерность входного пространства; i — число эпох обучения. Экспериментальное время выполнения одной эпохи обучения для $n = 400$, $N = 8000$, $d = 4$ составило $t \approx 100$ с, т. е. алгоритм обучения клеточной НС Кохонена работает в Excel приблизительно в 2 раза медленнее, чем классический алгоритм обучения НС Кохонена в программе SOM_PAK [1]. Это незначительное возрастание времени обучения клеточной НС Кохонена, очевидно обусловленное низким быстродействием электронных таблиц, многократно окупается наглядностью, гибкостью анализа и эффективной визуализацией результатов кластеризации и классификации многомерных данных в Excel.

Показано полезное применение краевого эффекта и многосвязных SOM для надежного выде-

ления границ группирования кластеров в линейно и нелинейно делимых пространствах учебных образцов и для решения проблемы "мертвых" нейронов, расположенных вблизи этих границ.

Список литературы

1. **Кохонен Т.** Самоорганизующиеся карты: пер. 3-го англ. изд. М.: БИНОМ. Лаборатория знаний, 2008. С. 158—337.
2. **Ultsch A., Siemon H. P.** Kohonen's Self Organizing Feature Maps for Exploratory Data Analysis // Proc. INNC'90 Neural Networks Conf. 1990. P. 305—308.
3. **Ultsch A.** U*-Matrix: A Tool to Visualize Clusters in High Dimensional Data // Department of Computer Science, University of Marburg, Technical Report. 2003. N. 36. P. 1—12.
4. **Nikkilä J., Törönen P., Kaski S., Venna J., Castrén E., Wong G.** Analysis and Visualization of Gene Expression Data Using Self-Organizing Maps // Neural Networks. 2002. N. 15. P. 953—966.
5. **Аникин В. И., Карманова А. А.** Обучение искусственной нейронной сети Кохонена клеточным автоматом // Информационные технологии. 2014. № 11. С. 73—80.
6. **Shah-Hosseini H., Safabakhsh H.** A TASOM-based Algorithm for Active Contour Modeling // Pattern Recognition Letters. 2003. V. 24. P. 1361—1373.
7. **Marghescu D.** Multidimensional Data Visualization Techniques for Financial Performance Data: A Review // TUCS Technical Report N. 810, 2007. P. 1—32.
8. **Аникин В. И., Аникина О. В.** Визуальное табличное моделирование клеточных автоматов в Microsoft Excel. Тольятти: Изд-во ПВГУС, 2013. 321 с.

V. I. Anikin, Professor, e-mail: anikin_vi@mail.ru, Volga State University of Service, Tolyatti,
A. A. Karmanova, Software Engineer, e-mail: turaeva.alexandra@mail.com, LLC "NetCracker", Tolyatti

Clustering and Classification of Multidimensional Data by Kohonen's Cellular Neural Network

The paper presents experimental results of multidimensional data clustering and classification by help of Kohonen's cellular neural network (CNN). An important feature of our study is that Kohonen's CNN and visualization tools (self-organizing map (SOM), U-, H- and P-matrixes, coordinate maps, map of data classes) have been implemented in Microsoft Excel spreadsheet, without programming in VBA. The user interface of this spreadsheet model makes it easy to change the configurable parameters and visually observe the neural network learning process and data classification results using cluster grouping method.

The experimental studies have shown high temporal efficiency of Kohonen's CNN learning algorithm. Having the time complexity of the algorithm is $T = O(n \cdot N \cdot d \cdot i)$, where n — the number of neurons, N — the size of training sample, d — the input space dimension, i — the number of training epochs, the experimental run time of one training epoch was $t \approx 100$ s for $n = 400$, $N = 8000$, $d = 4$. That is, the learning algorithm of Kohonen's CNN works in Excel about 2 times slower than the classic learning algorithm in the SOM_PAK program. This slight training time increase in Excel is repeatedly paid by clarity, analysis flexibility and effective visualization of multi-dimensional data clustering patterns.

A useful application of the edge effect and multilinked SOMs for reliable identification of clusters grouping boundaries in linear and nonlinear separable input spaces and the possibility of solving a well-known problem of "dead" neurons are shown.

Keywords: Kohonen's neural network, cellular automata, Excel, multi-dimensional data classification, visualization, U-matrix, P-matrix

References

1. **Kohonen T.** Samoorganizujuschiecija karti. Moskva: BINOM. Laboratoriya znaniy. 2008. P. 159—337.
2. **Ultsch A., Siemon H. P.** Kohonen's Self Organizing Feature Maps for Exploratory Data Analysis. Proc. INNC'90 Neural Networks Conf. 1990. P. 305—308.
3. **Ultsch A.** U*-Matrix: A Tool to Visualize Clusters in High Dimensional Data. Department of Computer Science, University of Marburg, Technical Report. 2003. N. 36. P. 1—12.
4. **Nikkilä J., Törönen P., Kaski S., Venna J., Castrén E., Wong G.** Analysis and Visualization of Gene Expression Data Using Self-Organizing Maps. Neural Networks. 2002. N. 15. P. 953—966.
5. **Anikin V. I., Karmanova A. A.** Obuchenie iskusstvennoi neuronnoi seti Kohonena kletochnim avtomatom. Informaciiionnie tehnologii. 2014. N. 11. P. 73—80.
6. **Shah-Hosseini H., Safabakhsh H.** A TASOM-based Algorithm for Active Contour Modeling. Pattern Recognition Letters. 2003. V. 24. P. 1361—1373.
7. **Marghescu D.** Multidimensional Data Visualization Techniques for Financial Performance Data: A Review. TUCS Technical Report N. 810, 2007. P. 1—32.
8. **Anikin V. I., Anikina O. V.** Vizualnoe tablichnoe modelirovanie kletochnih avtomatov v Microsoft Excel. Tolyatti: PVGUS, 2013. 321 p.

Нейросетевая оценка сложности IDEF-диаграмм

Рассмотрена нейросетевая система для оценки сложности восприятия IDEF-моделей человеком, отличающаяся применением экспертного метода на основе парных сравнений для получения данных, используемых для обучения многослойного персептрона, что позволяет для произвольной IDEF-диаграммы прогнозировать ее коэффициент сложности восприятия, по которому можно проводить сравнение и оптимизацию IDEF-моделей. Результаты, изложенные в статье, могут быть полезны при построении IDEF-моделей, а также проектировании CASE-средств на их основе.

Ключевые слова: CASE-средства, искусственная нейронная сеть, коэффициент сложности восприятия, экспертный метод, метод парных сравнений

Введение

Визуальные модели, выполненные согласно группе стандартов IDEF (Integrated Computer-Aided Manufacturing) (далее IDEF-модели), широко используются в практике моделирования сложных систем, CASE+-средствах и при реинжиниринге бизнес-процессов [1–4]. Среди IDEF-моделей наибольшее распространение имеют: SADT (IDEF0) — функциональные модели, DFD — диаграммы потоков данных, IDEF3 — модели процессов. IDEF-модели представляют собой набор диаграмм, связанных между собой по иерархическому принципу. Каждая из диаграмм содержит от двух до восьми блоков, а также как внешние связи, так и между блоками.

При построении IDEF-моделей важной проблемой является представление диаграмм таким образом, чтобы обеспечить наилучшее понимание модели пользователем-человеком. Предлагается характеризовать сложность для понимания диаграммы человеком скалярным положительным параметром — коэффициентом сложности восприятия (КСВ). Чем больше КСВ для данной диаграммы, тем сложнее она воспринимается человеком.

Для прогнозирования КСВ исходя из структуры диаграммы предлагается использовать нейросетевую систему на основе многослойного персептрона. Для получения обучающей выборки используется экспертный метод и метод парных сравнений, так как экспертам достаточно тяжело в абсолютной шкале точно оценить КСВ для большого числа диаграмм.

В статье далее подробно рассмотрены процесс получения обучающей выборки, а также разработка искусственной нейронной сети на основе пакета расширения MATLAB Neural Network Toolbox [5].

Формирование обучающей выборки

В соответствии со стандартами IDEF-диаграммы содержат не более восьми блоков. Структура диаграммы представляется вектором $\mathbf{s} = [n_1, n_2, \dots, n_8]$, где n_j — число связей j -го блока ($j = 1, 2, \dots, 8$). Блок обязательно должен содержать связи таким образом, если на диаграмме имеется k блоков, то $n_j = 0$ при $j > k$.

Процесс формирования обучающей выборки состоит из следующих шагов.

Шаг 1. Генерируются N IDEF-диаграмм со случайным числом блоков (от 2 до 8) и случайными внешними и межблочными связями (от 2 до 20 на один блок). Указанным диаграммам сопоставляются векторы \mathbf{S}_i ($i = 1, 2, \dots, N$).

Шаг 2. В соответствии с методом парных сравнений экспертам последовательно представляются диаграммы с номерами l и m , после чего предлагается оценить, во сколько раз КСВ l -й диаграммы Q_l больше, чем m -й — Q_m , в виде коэффициента $a_{lm} = Q_l/Q_m$ (a_{lm} выбирается из множества значений $\{1, 2, 3, \dots, 9\}$) [6]. Всего экспертам необходимо сделать оценку $M = C_N^2$ раз, C_N^2 — число сочетаний из N по 2. Например, при $N = 200$ экспертам необходимо ответить на $M = 19\,900$ вопросов, что занимает порядка 100 человеко-часов.

Шаг 3. Строится обратная симметричная матрица парных сравнений [6]:

$$\mathbf{A} = \begin{pmatrix} 1 & a_{12} & \dots & a_{1n} \\ 1/a_{12} & 1 & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ 1/a_{1n} & 1/a_{2n} & \dots & 1 \end{pmatrix}. \quad (1)$$

Для нахождения КСВ согласно методу парных сравнений необходимо найти главный собственный вектор $\mathbf{q} = (q_1, q_2, \dots, q_N)^T$ матрицы \mathbf{A} (см. (1)), который определяется уравнением [6]:

$$\mathbf{A}\mathbf{q} = \lambda_{\max}\mathbf{q}, \quad (2)$$

где λ_{\max} — максимальное собственное значение матрицы \mathbf{A} .

В силу того, что матрица \mathbf{A} имеет большую размерность, нахождение ее главного собственного вектора представляет определенные сложности. Для решения уравнения (2) использовался степенной метод (счет на установление) — метод простых итераций с нормированием на каждом шаге [7].

В качестве начального приближения выбирается вектор

$$\mathbf{q}_0 = \frac{1}{N}(1, 1, \dots, 1)^T.$$

Итерационная процедура описывается формулами

$$\begin{cases} \mathbf{y}_{k+1} = \mathbf{A}\mathbf{q}_k, \\ \mathbf{q}_{k+1} = \mathbf{y}_{k+1} / \|\mathbf{y}_{k+1}\|_1, \end{cases} \quad (3)$$

где $\|\mathbf{y}\|_1 = \sum_{n=1}^N |y_{(n)}|$ — l_1 -норма вектора; $y_{(n)}$ — n -я компонента вектора \mathbf{y} .

Отметим, что даже проведение одной итерации процедуры (3) может дать приемлемый по точности результат [6].

Для получения результата с заданной точностью ε необходима реализация итерационной процедуры (3) до выполнения неравенства $\|\mathbf{q}_{k+1}\| - \|\mathbf{q}_k\| < \varepsilon$ [7].

Для оценки того, насколько согласованно мнение экспертов при ответе на вопросы, необходимо найти наибольшее собственное значение матрицы \mathbf{A} [6]

$$|\lambda_{\max}| = \|\mathbf{q}_{k+1}\| / \|\mathbf{q}_k\|$$

и индекс согласованности

$$I_S = \frac{\lambda_{\max} - N}{N - 1}.$$

Мнение экспертов считается согласованным при выполнении соотношения

$$I_S / I_{SL} < 0,1,$$

где I_{SL} — случайный индекс (средний индекс согласованности сгенерированных случайным образом по шкале от 1 до 9 обратно симметричных матриц размерности N).

В результате описанных выше действий получена обучающая выборка

$$\langle \mathbf{s}_i, Q_i \rangle, \quad i = 1, 2, \dots, N, \quad (4)$$

где \mathbf{s} — вектор структуры i -й диаграммы; Q_i — КСВ i -й диаграммы.

Проектирование искусственной нейронной сети

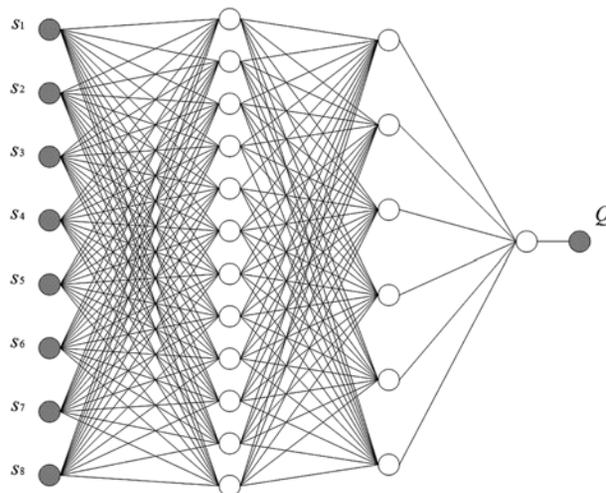
В качестве искусственной нейронной сети выберем многослойный персептрон с сигмоидальными функциями принадлежности [5]. Данный тип нейронной сети хорошо показал себя при решении задач аппроксимации сложных зависимостей, если имеется обучающая выборка достаточного объема. Кроме того, имеется большое число пакетов программ, в которых реализована эмуляция многослойного персептрона.

Для выбора структуры нейронной сети существует несколько эмпирических формул, воспользуемся следующими из них [5]:

$$\begin{cases} \frac{mN}{1 + \log_2 N} \leq L_w \leq m \left(\frac{N}{n} + 1 \right) (n + m + 1) + m; \\ L = \frac{L_w}{m + n}; \end{cases} \quad (5)$$

$$\frac{n}{2} \leq L_S \leq 3n, \quad (6)$$

где n — размерность входного вектора; m — размерность выходного вектора; N — объем обучаю-



Структура искусственной нейронной сети

щей выборки; L_w — число весов нейронной сети; L — число нейронов сети; L_S — число нейронов в скрытых слоях.

Подставляя в формулы (5) и (6) числовые значения $N = 200$, $n = 9$, $m = 1$, получим: $3 \leq L \leq 29$, $4 \leq L_S \leq 24$.

Выберем трехслойный персептрон со структурой 12—6—1. Число нейронов в первом скрытом слое — 12, во втором скрытом слое — 6, в выходном слое — 1, общее число нейронов — 19. Структура искусственной нейронной сети приведена на рисунке. Закрашенными кружками показаны входы и выход сети, незакрашенными кружками — нейроны.

Данная сеть эмулировалась с применением пакета MATLAB Neural Network Toolbox и обучалась на основе данных выборки (4) [5].

Обученная нейронная сеть может использоваться для прогнозирования КСВ IDEF-диаграмм на основе их структуры.

Заключение

Предложенная в статье нейросетевая система для оценки коэффициента сложности восприятия IDEF-диаграмм [8] может найти применение при разработке CASE-средств. В частности, предполагается разработать алгоритмы и реализующие их программные приложения, которые позволят в процессе взаимодействия с пользователем в диалоговом режиме строить оптимальные с точки зрения сложности восприятия IDEF-модели, что улучшит технологии разработки программного обеспечения.

Список литературы

1. Stahlknecht P., Hasenkamp U. Einführung in die Wirtschaftsinformatik. Berlin—Heidelberg—New York: Springer-Verlag, 2012. 327 p.
2. Sommerville I. Software Engineering (9th Edition). Boston—Columbus—Indianapolis—New York: Addison—Wesley, 2011. 790 p.
3. Вендров А. М. Проектирование программного обеспечения экономических информационных систем. 2-е изд., перераб. и доп. М.: Финансы и статистика, 2006. 544 с.

4. Черемных С. В., Семенов И. О., Ручкин В. С. Моделирование и анализ систем. IDEF-технологии: практикум. М.: Финансы и статистика, 2006. 192 с.
5. Дьяконов В. П., Круглов В. В. MATLAB 6.5 SP1/7/7 SP1/7 SP22 + Simulink 5/6. Инструменты искусственного интеллекта и биоинформатики. М.: Солон-Пресс, 2006. 456 с.
6. Саати Т. Принятие решений. Метод анализа иерархий. М.: Радио и связь, 1993. 278 с.
7. Калиткин Н. Н. Численные методы. М.: Наука, 1978. 512 с.

8. Усков А. А., Жукова А. Г. Матричное представление иерархических визуальных моделей для CASE-средств // Информационные технологии. 2014. № 11. С. 36–39.
9. Усков А. А., Жукова А. Г., Кондратова Н. В. Программный модуль "Оценка коэффициента сложности восприятия визуальных моделей". Свидетельство о государственной регистрации программы для ЭВМ в Федеральной службе по интеллектуальной собственности (РОСПАТЕНТ) № 2014618234 от 13.08.2014.

A. A. Uskov, Professor, e-mail: prof.uskov@gmail.com, A. G. Zhukova, Russian University of Cooperation

Neural Network Assessment Challenges IDEF-Charts

Considered neural network system for estimating the complexity of perception IDEF-human model, characterized by the use of expert-based method of paired comparisons for the data used for training multilayer perceptron, which allows for arbitrary IDEF-charts to predict its factor of perception, where you can make a comparison and optimization IDEF-models. The results presented in this paper may be useful in the construction of IDEF-models as well as the design of CASE-tools based on them.

When constructing models IDEF-important issue is to present the diagrams so as to ensure the best understanding of the model a human user. It is proposed to characterize the complexity of human-to-understand diagrams positive scalar parameters — degree of difficulty of perception. Than it is for this chart is, the more difficult it is perceived by man.

In order to predict the perception of degree of difficulty, based on the structure of the diagram, it is proposed to use the neural network system based on multilayer perceptron. For the training sample is used expert method and the method of paired comparisons, as the experts is rather difficult to accurately estimate the absolute scale factor of the perception of a large number of diagrams. The article discussed in detail the process of obtaining the training sample, as well as the development of an artificial neural network based on the expansion pack MATLAB Neural Network Toolbox.

The proposed article neural network system for evaluating factor of perception IDEF-diagrams can be used in the development of CASE-tools. In particular, it is proposed to develop algorithms and implement their software applications that allow a process of interaction with the user interactively construct optimal from the point of view of the complexity of perception IDEF-model, which will improve the software development technology.

Keywords: CASE-tools, artificial neural network, factor of perception, expert method, the method of paired comparisons

References

1. Stahlknecht P., Hasenkamp U. *Einführung in die Wirtschaftsinformatik*. Berlin—Heidelberg—New York: Springer-Verlag, 2012. 327 p.
2. Sommerville I. *Software Engineering* (9th Edition). Boston—Columbus—Indianapolis—New York: Addison—Wesley, 2011. 790 p.
3. Vendrov A. M. *Proektirovanie programmnogo obespechenja ekonomicheskikh informacionnyh sistem*. 2-e izd., pererab. i dop. M.: Finansy i statistika, 2006. 544 p.
4. Cheremnyh S. V., Semenov I. O., Ruchkin V. S. *Modelirovanie i analiz sistem. IDEF-tehnologii: praktikum*. M.: Finansy i statistika, 2006. 192 p.

5. D'jakonov V. P., Kругlov V. V. *MATLAB 6.5 SP1/7/7 SP1/7 SP2 + Simulink 5/6. Instrumenty iskusstvennogo intellekta i bioinformatiki*. M.: Solon-Press, 2006. 456 p.
6. Saati T. *Prinjatje reshenij. Metod analiza ierarhij*. M.: Radio i svjaz', 1993. 278 p.
7. Kalitkin N. N. *Chislennye metody*. M.: Nauka, 1978. 512 p.
8. Uskov A. A., Zhukova A. G. *Matrichnoe predstavlenie ierarchicheskikh vizual'nyh modelej dlja CASE-sredstv. Informacionnye tehnologii*. 2014. N. 11. P. 36–39.
9. Uskov A. A., Zhukova A. G., Kondratova N. V. *Programmnyj modul' "Ocenka koefefficienta slozhnosti vosprijatija vizual'nyh modelej"*. Svidetel'stvo o gosudarstvennoj registracii programmy dlja JeVM v Federal'noj sluzhbe po intellektual'noj sobstvennosti (ROSPATENT) N 2014618234 ot 13.08.2014.

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 08.04.2015. Подписано в печать 25.05.2015. Формат 60×88 1/8. Бумага офсетная.

Усл. печ. л. 8,86. Заказ ИТ615. 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 солюшнз". Отпечатано в ООО "Авансд солюшнз".

119071, г. Москва, Ленинский пр-т, д. 19, стр. 1.