

# ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Том 22  
2016  
№ 12

ТЕОРЕТИЧЕСКИЙ И ПРИКЛАДНОЙ НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ

Издается с ноября 1995 г.

УЧРЕДИТЕЛЬ  
Издательство "Новые технологии"

## СОДЕРЖАНИЕ

### ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

- Разоренов А. А., Фомичев В. А. Компактная формализация входных и промежуточных данных алгоритмов семантического анализа предписаний . . . . . 883
- Домников А. С., Белоус В. В. Синтез рационального упорядочения учебных модулей методом Бержа . . . . . 892
- Малыгин И. Г., Комашинский В. И. Информационные технологии и искусственный интеллект — основные двигатели четвертой индустриальной революции (Industrie 4.0) . . . . . 899

### МОДЕЛИРОВАНИЕ И ОПТИМИЗАЦИЯ

- Львович И. Я., Львович Я. Е., Преображенский А. П., Чопоров О. Н. Особенности методов вычислительной гидродинамики для моделирования турбулентности . . . . . 905
- Бронштейн Е. М., Давлетбаев А. А. Задача маршрутизации с учетом зависимости стоимости транспортировки от погодных условий . . . . . 913
- Ефлов В. Б., Питухин Е. А., Стафеев Ф. В. Вопросы производительности при выполнении запросов в реляционной СУБД H2. . . . . 918
- Орехов Э. Ю., Орехов Ю. В. Сравнение и оценка качества эвристических алгоритмов на конечной массовой задаче в условиях неравновероятной генерации тестовых примеров. . . . . 925

### ЦИФРОВАЯ ОБРАБОТКА СИГНАЛОВ И ИЗОБРАЖЕНИЙ

- Гаврилов С. В., Гуров С. И., Жукова Т. Д., Рыжова Д. И. Применение теории кодирования для повышения помехозащищенности комбинационных схем . . . 931

### ПРОГРАММНАЯ ИНЖЕНЕРИЯ

- Решетова Е. Н. Универсальные компетенции студента образовательного направления "Программная инженерия" . . . . . 938

### СИСТЕМЫ СВЯЗИ И ТЕЛЕКОММУНИКАЦИИ

- Соловьев Б. И., Жусов Д. Л., Просолупов С. А. Определение мест размещения сетевых ресурсов на квазиоднородной топологической структуре инфокоммуникационной системы . . . . . 946

### ПРИКЛАДНЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

- Тарасов В. Н., Липилина Л. В., Бахарева Н. Ф. Автоматизация расчета характеристик систем массового обслуживания для широкого диапазона изменения их параметров . . . . . 952
- Указатель статей, опубликованных в журнале "Информационные технологии" в 2016 году. . . . . 958

Главный редактор:  
СТЕМПКОВСКИЙ А. Л.,  
акад. РАН, д. т. н., проф.

Зам. главного редактора:  
ИВАННИКОВ А. Д., д. т. н., проф.  
ФИЛИМОНОВ Н. Б., д. т. н., с.н.с.

Редакционный совет:  
БЫЧКОВ И. В., акад. РАН, д. т. н.  
ЖУРАВЛЕВ Ю. И.,  
акад. РАН, д. ф.-м. н., проф.  
КУЛЕШОВ А. П.,  
акад. РАН, д. т. н., проф.  
ПОПКОВ Ю. С.,  
акад. 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.,  
акад. 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.,  
акад. 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. В., д. т. н., проф.  
ЛЬВОВИЧ Я. Е., д. т. н., проф.  
МАРТЫНОВ В. В., 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.  
САКСОНОВ Е. А., 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.  
ШИЛОВ В. В., к. т. н., доц.

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

Информация о журнале доступна по сети Internet по адресу <http://novtex.ru/IT>.  
Журнал включен в систему Российского индекса научного цитирования.  
Журнал входит в Перечень научных журналов, в которых по рекомендации ВАК РФ должны быть опубликованы научные результаты диссертаций на соискание ученой степени доктора и кандидата наук.

# INFORMATION TECHNOLOGIES

# ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Vol. 22

2016

No. 12

THEORETICAL AND APPLIED SCIENTIFIC AND TECHNICAL JOURNAL

Published since November 1995

ISSN 1684-6400

## CONTENTS

### INTELLIGENT SYSTEMS AND TECHNOLOGIES

- Razorenov A. A., Fomichov V. A.** Compact Formalization of Instructions Semantic Parsing Algorithms' Input and Intermediary Data Structures. . . . . 883
- Domnikov A. S., Belous V. V.** Synthesis of Rational Taught Modules Ordering by Berzha's Method . . . . . 892
- Malygin I. G., Komashinskiy V. I.** Information Technology and Artificial Intelligence — the Main Engines of the Fourth Industrial Revolution (Industrie 4.0) . . . . . 899

### MODELING AND OPTIMIZATION

- Lvovich I. Ya., Lvovich Ya. E., Preobrazhensky A. P., Choporov O. N.** The Features of the Methods of Computational Fluid Dynamics for Turbulence Modeling . . . . . 905
- Bronshtein E. M., Davletbaev A. A.** Vehicle Routing Problem with Transport Cost Depending on Weather . . . . . 913
- Yeflov V. B., Pitukhin E. A., Stafeev F. V.** Performance Issues of Query Execution in Relational H2 Database . . . . . 918
- Orekhov E. Yu., Orekhov Yu. V.** Comparing and Estimating Quality of Heuristic Algorithms on a Finite Problem under Non-Equiprobable Generation of Test Instances . . . . . 925

### DIGITAL PROCESSING OF SIGNALS AND IMAGES

- Gavrilov S. V., Gurov S. I., Zhukova T. D., Ryzhova D. I.** Application of Coding Theory to Improve the Noise Immunity of Combinational Circuits . . . . . 931

### SOFTWARE ENGINEERING

- Reshetova E. N.** Student Universal Competences of Educational Direction "Software Engineering" . . . . . 938

### CONNECTIONS AND TELECOMMUNICATIONS SYSTEMS

- Soloviev B. I., Zhusov D. L., Prosolupov S. A.** Determining the Placement of Network Resources on a Quasi-Homogeneous Topological Structure of Information and Communication Systems. . . . . 946

### APPLIED INFORMATION TECHNOLOGIES

- Tarasov V. N., Bahareva N. F., Lipilina L. V.** Automation for Calculating Characteristics Queuing System for a Wide Range Changing Their Parameters . . . . . 952
- Index** of articles published in the journal "Information Technology" in 2016 . . . . . 958

#### Editor-in-Chief:

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

#### Deputy Editor-in-Chief:

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

#### Chairman:

Bychkov I. V., Member of RAS,  
Dr. Sci. (Tech.), Prof.  
Zhuravljov Yu. I., Member of RAS,  
Dr. Sci. (Phys.-Math.), Prof.  
Kuleshov A. P., Member of RAS,  
Dr. Sci. (Tech.), Prof.  
Popkov Yu. S., 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., 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., 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. V., Dr. Sci. (Tech.), Prof.  
Lvovich Ya. E., Dr. Sci. (Tech.), Prof.  
Martynov V. V., 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.  
Saksonov E. A., 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.  
Shilov V. V., Cand. Sci. (Tech.), Ass. Prof.

#### Editors:

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

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

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

УДК 004.512; 004.89

**А. А. Разоренов**, аспирант, e-mail: arazorenov@hse.ru,

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

Национальный исследовательский университет "Высшая школа экономики", Москва

### Компактная формализация входных и промежуточных данных алгоритмов семантического анализа предписаний

*Предлагаются новые формальные средства для разработки алгоритмов семантического анализа (парсинга) естественно-языковых предписаний. В качестве методологической основы исследования выбрана теория K-представлений (концептуальных представлений) — единственная сегодня теория, удобная для построения семантических представлений произвольно сложных предписаний. Изложены следующие теоретические результаты: (а) построена новая формальная модель лингвистической базы данных (ЛБД), эта модель значительно более компактна по сравнению с двумя моделями ЛБД, рассматриваемыми теорией K-представлений; (б) предложена новая формализация понятия "семантико-синтаксическая структура предписания". Предложенные формальные средства были использованы в проектировании практически полезного естественно-языкового интерфейса файлового менеджера.*

**Ключевые слова:** *естественный язык, семантический анализ предписаний, семантическое представление, теория K-представлений, СК-язык, морфологический базис, шаблон K-представления, словарь лингвистических фреймов, лингвистическая база данных, граф семантико-синтаксических связей, естественно-языковое управление файловой системой, Natural Language Commander*

#### Введение

В начальный период исследований по разработке систем общения на естественном языке (ЕЯ) с прикладными интеллектуальными системами одной из центральных была идея использования семантического языка-посредника (или концептуального языка) для формального представления содержания (смысла) входного текста на ограниченном естественном языке (ЕЯ-текста). В частности, в монографии [1] с этой целью использовался оригинальный язык концептуальных диаграмм.

Благодаря стремительному прогрессу сети Интернет пользователи из разных стран получили технический доступ к огромному массиву текстов на английском, русском и многих других языках. Возникла проблема разработки компьютерных средств, облегчающих поиск таких текстов, которые могут представлять интерес для пользователя. Поэтому в последние два десятилетия основное внимание исследователей уделялось созданию статистических методов обработки ЕЯ-текстов.

Однако с начала 2000-х годов, особенно в последнее десятилетие, можно было наблюдать быстрое возрождение интереса к идее семантической обработки ЕЯ-текстов с использованием семантического языка-посредника. В большой степени эта

тенденция связана со значительным ростом интереса в течение последних 5—7 лет к естественно-языковым интерфейсам (ЕЯИ), способным выполнять семантический анализ предписаний. Основная часть публикаций рассматривает различные аспекты взаимодействия с роботами [2—7] (работа [4] базируется на статье [8]), управления персонажами видеоигр, навигации в двух- и трехмерном пространстве [9] и программирования на ЕЯ [10, 11].

Кроме указанных применений ЕЯИ одной из актуальных тем является управление файловой системой и в целом компьютером или мобильным устройством (смартфоном, планшетом). К практическим результатам, полученным в этом направлении, в частности, относятся:

- система речевого управления мобильными устройствами Apple Siri [12];
- ее русскоязычный аналог Собеседник HD [13];
- Braina Project — программное обеспечение для управления компьютером в целом [14];
- NLUI Server — серверное программное обеспечение для построения сценариев взаимодействия с программами на ЕЯИ встраивания поддержки ЕЯИ в приложения [15].

Задача семантического анализа предписаний смещает акценты со статистических методов обработки ЕЯ-текстов к разработке методов извлечения

смысла из текста. Формальные методы проектирования таких ЕЯИ разработаны еще недостаточно. Основная причина заключается в том, что наиболее популярные средства представления смысла текстов базируются на логике первого порядка или лямбда-исчислении. Оба подхода ориентированы на изучение утверждений, множества их выражений (формул) не включают формальных аналогов предписаний (императивов).

По этой причине перечисленные выше ЕЯИ для обработки предписаний ориентируются на заранее подготовленные шаблоны. Для данных подходов характерны:

- доминирующая ориентация на наборы простых (одиночных) предписаний, т.е. предписаний, образованных одним глаголом с зависимыми словами;
- необходимость предусмотреть полный спектр поддерживаемых предписаний.

Анализ публикаций по формальной семантике ЕЯ показывает, что существует по меньшей мере одна теория, предлагающая выразительные механизмы для формального описания структурированных значений, или семантической структуры, произвольно сложных предписаний на ЕЯ. Это теория К-представлений (концептуальных представлений); на первой стадии разработки она называлась теорией К-исчислений и К-языков. Это оригинальная теория проектирования семантико-синтаксических анализаторов ЕЯ-текстов с широким использованием формальных средств для представления входных, промежуточных и выходных данных. Теория К-представлений (ТКП) представлена в большой серии публикаций на русском и английском языках [16—30]. ТКП является центральной частью Интегральной формальной семантики ЕЯ — научного направления на стыке математической информатики, математической лингвистики и компьютерной лингвистики [17, 27].

Данная статья продолжает серию публикаций [31, 32], описывающих методологию проектирования файлового менеджера с ЕЯИ NLC-1 (Natural Language Commander — Version 1) в рамках ТКП. Цель статьи заключается в разработке новых формальных средств для проектирования алгоритмов семантического анализа сложных предписаний. В разделе 1 рассматриваются основные существующие подходы к семантическому анализу предписаний. В разделах 2—4 описываются новая математическая модель лингвистической базы данных и новый подход к формальному представлению семантико-синтаксической структуры ЕЯ-текстов. Раздел 5 кратко описывает применение разработанных формальных средств к проектированию файлового менеджера с ЕЯИ NLC-2 (Natural Language Commander — Version 2).

## 1. Выбор формы семантических представлений предписаний

Проектные решения в процессе разработки алгоритмов семантико-синтаксического анализа в значительной степени зависят от используемых подходов к представлению структурированных значений (СЗ), или смысловой структуры, предписаний на ЕЯ. Используя другую терминологию, часто говорят о подходах к конструированию семантических представлений (СП) предписаний. Анализ научной литературы показывает, что основными подходами к этой проблеме, используемыми на практике, являются лямбда-исчисление [2], логика предикатов первого порядка [7], абстрактное представление смыслов (АПС) [4, 8] и теория К-представлений [16—30].

Например, в работе [2] средствами лямбда-исчисления предписанию "Move to the chair in the third intersection" поставлено в соответствие СП вида

$$La.move(a) \wedge to(a, Jx.chair1(x)) \wedge \wedge intersect(order(Ly.junction(y), frontdist, 3), x),$$

где  $L$  и  $J$  — соответственно строчные буквы греческого алфавита лямбда и йота. Выражение  $La.move(a)$  является частным случаем лямбда-оператора (или оператора абстракции), это выражение задает понятие "перемещение" (move). Выражение  $Jx.chair1(x)$  — это частный случай йота-оператора; данное выражение используется для обозначения конкретного объекта с меткой  $x$ , являющегося стулом.

Смысловая структура предписания "Take the book on the table" представляется средствами АПС в [4] следующим образом:

$$(t/take — Taking: Theme (b /book); Source (t1/table); Location (o/n; trajet(b))).$$

Одним из существенных недостатков использования СП такого рода для отображения СЗ ЕЯ-текстов является потеря выразительности. Другими словами, если текст относительно большой, его СП не дает возможности восстановить исходный текст (например, для контроля этого процесса пользователем).

СП предписаний, сформированные средствами лямбда-исчисления или АПС, обычно выглядят по-разному. Тем не менее, их важной общей чертой является возможность построения описания различных сущностей (действий и предметов), упоминаемых в предписании.

Подходы лямбда-исчисления и АПС довольно удобны для построения СП простых предписаний: с одним глаголом и зависимыми словами, без связки ИЛИ. Однако предписания, возникающие в реальных приложениях, могут быть существенно более сложными: включать обозначения нескольких действий, соединенные связками И или ИЛИ, указывать последовательность действий, временные интервалы между действиями, условия выполнения действий, упоминать составные обозначения

групп объектов как операнды действий, включать в себя модальные слова "нужно", "следует" и т.д.

Анализ показывает, что выразительные возможности лямбда-исчисления и АПС недостаточны для эффективной работы со сложными инструкциями такого рода. Тем не менее, с середины 1990-х годов имеется подход, свободный от перечисленных ограничений: теория К-представлений (концептуальных представлений), разработанная В. А. Фомичевым [16—30]. Уже ранняя версия этой теории (теория ограниченных К-исчислений и ограниченных стандартных К-языков) [18—19] ввела класс формальных языков с выразительными возможностями АПС, за семнадцать лет до изложения методологии АПС в работе [8]. Часть преимуществ ограниченных стандартных К-языков и СК-языков (стандартных концептуальных языков) указана в заключительной части данного раздела.

Первая часть теории К-представлений — математическая модель (Модель 1) системы первичных единиц концептуального уровня, используемой прикладной интеллектуальной системой. Эта модель определяет, в частности, новый класс формальных объектов, называемых *концептуальными базисами* (к.б.). Построение произвольного к.б.  $B$  равносильно определению некоторой конечной последовательности формальных объектов  $Tuple(B)$ . Рассмотрим интерпретацию ее выделенных компонентов  $St, X, V, F, tp$ . Компонент  $St$  — конечное множество символов, называемых *сортами* и интерпретируемых как наиболее общие понятия, используемые в рассматриваемой предметной области: *физический-объект, интеллектуальная-система, организация, значение-расстояния, значение-стоимости* и т.п. Счетное множество  $V$  содержит *переменные*. Счетное множество  $X$  включает в качестве подмножества множество сортов  $St$ , а также содержит другие символы, интерпретируемые как первичные информационные (или концептуальные) единицы. Множество  $X$  называется *первичным информационным универсумом* к.б.  $B$ . Конечное подмножество  $F$  множества  $X$  содержит обозначения функций.

Компонент  $tp$  последовательности  $Tuple(B)$  — это отображение объединения  $X$  и  $V$  в счетное множество строк  $Types(B)$ , включающее в себя множество сортов  $St$ . Элементы этого множества называются *типами* и интерпретируются как структурированные характеристики (ярлыки) сущностей, обозначаемых элементами множеств  $X$  и  $V$ . Отображение  $tp$  дает более тонкое (более мелкочаечное) структурирование предметной области, чем это позволяет сделать логика предикатов первого порядка.

**Пример.** К.б.  $B$  может удовлетворять следующим условиям: (а)  $St$  включает в себя элементы (сорта) *дин.физ.об* (динамический физический объект), *интс* (интеллектуальная система), *орг* (организация), *инф.об* (информационный объект); (б)  $X$  включает

в себя элементы *Лев-Толстой, Война-и-Мир, человек, тур-группа, Поставщики, Авторство*, и

$$tp(\text{человек}) = \uparrow \text{интс} * \text{дин.физ.об}, tp(\text{Лев-Толстой}) = \\ = \text{интс} * \text{дин.физ.об},$$

$$tp(\text{Война-и-Мир}) = \text{инф.об}, tp(\text{Авторство}) = \\ = \{\{\text{интс}, \text{инф.об}\}\},$$

$$tp(\text{тур-группа}) = \uparrow \{\text{интс} * \text{дин.физ.об}\}, \\ tp(\text{Поставщики}) = \{\{\text{орг}, \{\text{орг}\}\}\}.$$

В данном случае символ  $\uparrow$  указывает на тип понятия; *Поставщики* — наименование функции, ставящей в соответствие организации множество всех ее поставщиков. Символ  $*$  используется для указания наличия двух или более различных семантических характеристик. Например, конкретный человек может в одном контексте рассматриваться как интеллектуальная система (в примере сорт *интс*), а в другом контексте — как динамический физический объект (в примере сорт *дин.физ.об*).

На множестве типов  $Types(B)$  определено отношение частичного порядка  $|-$ , называемое *отношением конкретизации* (здесь символ  $|-$  используется нестандартно, т.е. не так, как в математической логике). Например, могут выполняться следующие соотношения:

$$\text{физ.об} \text{ } |- \text{ } \text{дин.физ.об}, \text{ физ.об} \text{ } |- \text{ } \text{интс} * \text{дин.физ.об}, \\ \text{интс} \text{ } |- \text{ } \text{интс} * \text{дин.физ.об}, \text{ интс} \text{ } |- \text{ } \text{интс} * \\ \text{дин.физ.об}, \{\text{физ.об}\} \text{ } |- \text{ } \{\text{интс} * \text{дин.физ.об}\}.$$

Вторая часть теории К-представлений определяет математическую модель (Модель 2) системы, состоящей из десяти частичных операций на концептуальных структурах. Модель 2 задает, в частности, новый класс формальных языков, называемых СК-языками (стандартными концептуальными языками). Есть веские основания предполагать, что СК-языки являются удобным формальным инструментом для построения СП сколь угодно сложных ЕЯ-текстов (предложений и дискурсов), относящихся к массовым сферам профессиональной деятельности (технологии, медицина, бизнес, спорт и т.д.). Термин "К-представление" (КП) используется для обозначения семантических представлений ЕЯ-текстов, являющихся выражениями СК-языка в каком-то концептуальном базисе.

Выражения СК-языков строятся из первичных информационных единиц, задаваемых к.б., и нескольких служебных символов с помощью индуктивного применения оригинальных правил (утверждений)  $P[0], P[1], \dots, P[10]$ . Формальный язык, порожденный концептуальным базисом  $B$  с помощью этих правил, обозначается через  $Ls(B)$ .

Правило  $P[0]$  говорит о том, что элементы первичного информационного универсума  $X(B)$  и переменные из  $V(B)$  входят в множество  $Ls(B)$ , другими словами, являются К-цепочками. В частности, единица *файл1* является К-цепочкой. Правила  $P[1], \dots, P[10]$  совместно определяют систему, со-

стоящую из десяти частичных операций на концептуальных структурах (см. [20, 21]).

**Пример.** Команда "Передвинься к стулу на третьем перекрестке" может иметь КП *Semrepr1* вида

*Выполнение.Действия(#сейчас#, перемещение1 \* (Целевой-простр-объект, нек стул1 \* (Локализация, нек перекресток1 \* (Номер, 3))))).*

СК-языки позволяют строить СП как простых, так и сложных предписаний с сохранением выразительности ЕЯ и без потери интуитивно понятной связи с исходным текстом предписания на ЕЯ. Выразительные средства СК-языков превосходят средства других рассмотренных выше подходов к формальному представлению структурированных значений предписаний на ЕЯ. Кроме того, нужно подчеркнуть дополнительные возможности СК-языков:

- использование единого формата для представления семантики предписаний и фрагментов знаний из онтологий;
- возможность представления целей и соединения составных целей активных систем с помощью логических связей;
- возможность конструирования формальных СП составных описаний множеств и описаний понятий;
- возможность отражения временных и иных отношений между отдельными целями, являющимися компонентами составной цели.

По отношению к проблеме отображения семантики предписаний перечисленные возможности составляют основную часть преимуществ класса СК-языков, определенного в 2005 г. в монографии [24], по сравнению с формализмом абстрактного представления смыслов, предложенным в 2013 г. 10 исследователями из Великобритании и США [8]. Более того, те же возможности предоставляет класс ограниченных стандартных К-языков, введенный в 1996 г. в статье [18].

## 2. Новое определение морфологического базиса

Начнем описывать центральные идеи нового подхода к преобразованию ЕЯ-предписаний в КП. Главная задача заключается в том, чтобы найти семантико-синтаксические связи между лексическими единицами обрабатываемого текста и на этой основе сформировать одно или несколько КП. С этой целью будут определены несколько новых формальных объектов: морфологический базис, множество шаблонов К-представлений, словарь лексических фреймов и лингвистическая база данных (раздел 3), граф семантико-синтаксических связей (раздел 4).

Сначала рассмотрим основные идеи определения нового класса формальных объектов, называемых *морфологическими базисами*. Более точно, будет дано новое определение морфологического базиса по

сравнению с определениями морфологического базиса, данными в статье [22] и монографиях [24, 27].

Пусть  $W$  — конечное множество элементарных значащих единиц текста входного языка, которое является объединением непересекающихся множеств символов

$$W_d, W_{sep}, W_{conn}, W_{prep},$$

где  $W_d$  — множество базовых лексических единиц языка (слов и коротких словосочетаний), несущих смысловую нагрузку;  $W_{sep}$  — множество разделителей: запятая, точка и т.п.;  $W_{conn}$  — множество слов-связок: "и", "или" и др.;  $W_{prep}$  — множество предлогов и, возможно, других лексических единиц, используемых для связи слов между собой.

Если  $Z$  — произвольное конечное множество, то булеаном множества  $Z$  называется множество  $Boolean(Z)$ , элементами которого являются все подмножества множества  $Z$ , в том числе пустое множество  $\emptyset$  и само множество  $Z$ .

Пусть  $Mprop$  — конечное множество символов, интерпретируемых как значения морфологических признаков, соответствующих лексическим единицам входного языка: наименования частей речи, подклассов частей речи, грамматических падежей и чисел, залога и т.д. Пусть  $M$  — некоторое подмножество множества  $Boolean(Mprop)$  (не исключается, что  $M$  совпадает с булеаном  $Boolean(Mprop)$ ), и  $< =:$  — отношение частичного порядка на  $M$  (т. е.  $< =:$  — это бинарное отношение на  $M$ , являющееся рефлексивным, транзитивным и антисимметричным).

**Пример.** Пусть  $Mprop = \{глагол, сущ, прош.вр., наст.вр., им.над., вин.над., иск.имя\}$ , где *иск.имя* — одно из возможных значений введенного в данной работе обобщенного морфологического признака "класс элементарной единицы текста". Такое значение признака имеют выражения в кавычках или апострофах — искусственные имена различных объектов ("Война и мир", 'Moscow News' и т.п.). Тогда  $M$  может содержать следующие подмножества  $Mprop$ :  $\{глагол\}$ ,  $\{глагол, прош.вр.\}$ ,  $\{глагол, наст.вр.\}$ ,  $\{сущ\}$ ,  $\{сущ, им.над.\}$ ,  $\{сущ, вин.над.\}$ , пустое множество  $\emptyset$ . Отношение  $< =:$  может быть в этом случае задано следующим образом:  $\{глагол\} < =: \{глагол, прош.вр.\}$ ,  $\{глагол\} < =: \{глагол, наст.вр.\}$ ,  $\{сущ\} < =: \{сущ, им.над.\}$ ,  $\{сущ\} < =: \{сущ, вин.над.\}$ ,  $\emptyset < =: Y$ , где  $Y$  — произвольный элемент  $M$  подмножества  $Mprop$ .

Пусть  $B$  — произвольный концептуальный базис (см. пояснение в разделе 1 и определение в работах [20, 24, 27]). Пусть также заданы следующие отображения:

- $rm: W \rightarrow Boolean(M)$  — отображение, ставящее в соответствие лексической единице входного языка множество возможных наборов ее морфологических признаков (пустое множество не исключается);

- $rc: W \rightarrow W$  — отображение, ставящее в соответствие элементарной единице текста его базовую морфологическую форму;
- $ru: W \rightarrow Boolean(X(B))$  — отображение, ставящее в соответствие элементарной значащей единице текста подмножество соответствующих ей элементов первичного информационного универсума. При этом для каждого  $d$  из  $W_{conn}$  множество  $ru(d)$  является подмножеством множества  $\{\wedge, \vee\}$ , т.е. множества, состоящего из логических связок конъюнкция и дизъюнкция.

Например, для некоторого подмножества русского языка, описывающего операции с файловой системой, выполняется соотношение  $rm(\text{папку}) = \{\{\text{сущ}, \text{вин.над.}\}\}$ . В свою очередь, функции  $rm$  и  $rc$  могут быть определены таким образом, что  $rc(\text{папку}) = \text{папка}$ ,  $rc(\text{файл}) = \text{файл}$ ,  $ru(\text{или}) = \{\vee\}$ ,  $ru(\text{и}) = \{\wedge\}$ .

Все математические объекты, охарактеризованные выше, являются частями морфологического базиса.

**Определение 1.** Пусть  $B$  — произвольный концептуальный базис (к.б.). Тогда морфологическим базисом, согласованным с к.б.  $B$ , называется произвольный упорядоченный набор  $Morph$  вида  $(Wd, Wsep, Wconn, Wprep, Mprop, M, < =:, rm, rc, ru)$ , где

- $Wd, Wsep, Wconn, Wprep, Mprop$  — попарно непесекающиеся конечные множества символов;
- $M$  — непустое подмножество булеана  $Boolean(Mprop)$ , и  $< =:$  — частичный порядок на множестве  $M$ ;
- $rm: W \rightarrow Boolean(M)$  — отображение из  $W$  в  $Boolean(M)$ , где  $W$  — объединение множеств  $Wd, Wsep, Wconn, Wprep$ ;
- $rc: W \rightarrow W$  — отображение из  $W$  в  $W$ ;
- $ru: W \rightarrow Boolean(X(B))$  — отображение из  $W$  в  $Boolean(X(B))$ , где  $X(B)$  — первичный информационный универсум к.б.  $B$ .

Морфологический базис в дальнейшем будет использован для установления связей между лексическими единицами текста.

### 3. Формальная модель лингвистической базы данных

Для построения новой математической модели лингвистической базы данных в дополнение к понятию морфологического базиса нам потребуется ввести понятия шаблона К-представления и словаря лексических фреймов.

#### 3.1. Шаблоны К-представлений

Для построения К-представлений нам потребуется ввести понятие шаблона К-представления. Например, шаблоном может являться цепочка вида  $x1 * (\text{Объект1}, x2)$ , где  $x1, x2$  — переменные типа *инф.объект*. Попробуем применить эту цепочку для

обработки фразы "Удали папку "Документы"". Тогда можно заменить переменные  $x1$  и  $x2$  К-цепочками, описывающими соответственно действие и объект, над которым проводится действие. Например, переменная  $x1$  может быть заменена К-цепочкой *удаление1*, а переменная  $x2$  — К-цепочкой *нек папка2 \* (Наименование1, "Документы")*. Тогда результатом замены будет КП *удаление1 \* (Объект1, нек папка2 \* (Наименование1, "Документы"))*.

**Определение 2.** Пусть  $B$  — произвольный к.б. Тогда *шаблоном К-представления* будем называть произвольный упорядоченный набор вида  $(frame, x, y)$ , где  $x$  и  $y$  — переменные к.б.  $B$ , а  $frame$  — некоторая цепочка символов, которая может быть преобразована в выражение (другими словами, К-цепочку,  $l$ -формулу) СК-языка  $Ls(B)$  заменой вхождений символов  $x$  и  $y$  на некоторые выражения (К-цепочки,  $l$ -формулы) языка  $Ls(B)$ . Множество всех таких шаблонов будем обозначать через  $K-templates(B)$ .

#### 3.2. Словарь лексических фреймов и модель лингвистической базы данных

*Словарь лексических фреймов* хранит данные, по которым устанавливаются семантико-синтаксические связи между лексическими единицами обрабатываемого текста. Он определяет требования к связываемым лексическим единицам — наборам значений их морфологических признаков и семантическим характеристикам. Для описания семантических ограничений мы воспользуемся механизмом структурирования предметных областей с помощью типов, предлагаемым теорией К-представлений. Этот механизм позволит нам, в частности, избежать формирования конструкций вида *удаление1 \* (Объект1, нек копирование)*, так как семантические единицы, соответствующие действиям и файлам, будут иметь разные типы.

Так как теория К-представлений определяет отношение конкретизации  $|-$  на множестве типов  $Types(B)$ , мы можем указывать наиболее общие семантические ограничения и проверять не совпадение типов  $t_1$  и  $t_2$ , а тот факт, что тип  $t_1$  является конкретизацией типа  $t_2$  (в частности, типы  $t_1$  и  $t_2$  могут совпадать). Например, типом семантической единицы (обозначения понятия) *человек* может являться цепочка *инт.сист\*дин.физ.об*, что позволит рассматривать человека и как динамический физический объект (определяя его местоположение в пространстве), и как интеллектуальную систему (при чтении новостей о разработке новых технологий и т.д.).

**Определение 3.** Пусть  $B$  — произвольный к.б.,  $Morph$  — произвольный морфологический базис (м.б.), согласованный с к.б.  $B$ . Тогда *словарем лексических фреймов*, согласованным с  $B$  и  $Morph$ , назовем произвольное конечное множество  $LexFrames$ , состоящее из упорядоченных наборов вида

$(type_1, morph_1, type_2, morph_2, prep, template, dir)$ ,

где выполнены следующие условия:

- $type_1$  и  $type_2$  являются типами из множества  $Types(B)$ ;
- $morph_1$  и  $morph_2$  являются элементами множества  $M$ , т.е. являются наборами значений морфологических признаков;
- $prep$  является либо элементом множества  $Wprep$ , либо пустым предлогом  $nil$ ;
- $template$  является шаблоном К-представления из множества  $K-templates(B)$ ;
- $dir$  — число из множества  $\{-1, 0, 1\}$ .

**Пример.** Некоторый словарь лексических фреймов, который мы будем использовать в дальнейшем в иллюстративных целях, может включать в себя наборы:

(действие, {глагол}, [объект], {сущ},  $nil$ ,  
(нек  $x1^*(Объект1, x2), x1, x2), 0$ );

(действие, {глагол}, простр.объект, {сущ, вин.пад.},  
 $v$ , (нек  $x1^*(Назначение1, x2), x1, x2), 0$ );

([объект], {сущ}, литерал, {иск.имя},  
(нек  $x2^*(Наименование1, x4), x2, x4), 1$ ).

**Определение 4.** Лингвистической базой данных называется упорядоченная тройка  $LingDb$  вида

$(B, Morph, LexFrames)$ ,

где  $B$  — произвольный к.б.,  $Morph$  — произвольный м.б., согласованный с к.б.  $B$ ;  $LexFrames$  — некоторый словарь лексических фреймов, согласованный с к.б.  $B$  и м.б.  $Morph$ .

#### 4. Граф семантико-синтаксических связей

Рассмотрим новую формальную интерпретацию понятия "семантико-синтаксическая структура естественно-языкового текста".

**Определение 5.** Графом семантико-синтаксических связей входного текста  $T = t_1...t_n$  для лингвистической базы данных  $(B, Morph, LexFrames)$  называется упорядоченная пара  $SemGraph = (V, E)$ , где  $V = \{1, ..., n \mid t_i \text{ входит в } Wd\}$  и  $E$  — такое множество всевозможных упорядоченных троек вида  $(i, j, (sem_1, sem_2, k, fr))$ , что  $i, j$  принадлежат множеству  $V$ ,  $k \geq 0$ ;  $fr$  — шаблон К-представления из множества  $K-templates(B)$ , и для определенного набора вида  $(type_1, morph_1, type_2, morph_2, prep, template, dir)$  словаря лексических фреймов  $LexFrames$  выполнены следующие условия:

- (1) множество семантических единиц  $ru(t_i)$ , соответствующее лексической единице  $t_i$ , включает такую единицу  $sem_1$ , что тип  $tp(sem_1)$  является конкретизацией типа  $type_1$  в частично упорядоченном множестве типов  $Types(B)$ ;
- (2) множество наборов значений морфологических признаков  $rm(t_i)$ , ассоциированное с лексиче-

ской единицей  $t_i$ , включает такой набор  $x$ , что  $morph_1 < =: x$ ;

- (3) множество семантических единиц  $ru(t_j)$ , соответствующее лексической единице  $t_j$ , включает такую единицу  $sem_2$ , что тип  $tp(sem_2)$  является конкретизацией типа  $type_2$  в частично упорядоченном множестве типов  $Types(B)$ ;
- (4) множество наборов значений морфологических признаков  $rm(t_j)$ , ассоциированное с лексической единицей  $t_j$ , включает такой набор  $y$ , что  $morph_2 < =: y$ ;
- (5) при  $k = 0$  элемент  $prep$  является пустым предлогом  $nil$ ; в случае  $k > 0$   $rc(t_k) = prep$  и  $k < j$ ; кроме того, если  $i < j$ , то  $i < k < j$ ;
- (6)  $fr = template$ ;
- (7) если  $dir = -1$ , то  $i > j$ ; если  $dir = 1$ , то  $i < j$ ; если  $dir = 0$ , то  $i$  отлично от  $j$ .

Перечисленные условия интерпретируются следующим образом. Условия 1—4 накладывают ограничения на пару слов, между которыми устанавливается семантическое отношение, которое затем будет описано К-представлением; это КП формируется с помощью шаблона  $template$  (условие 6). При этом условие 1 накладывает ограничение на тип семантической единицы, поставленной в соответствие главной лексической единице, а условие 2 — на значения морфологических признаков лексической единицы. Условия 3 и 4 накладывают аналогичные ограничения на подчиненную лексическую единицу. Условие 5 учитывает участие предлогов в установлении семантического отношения. Кроме того, это условие говорит о том, что предлог предшествует подчиненной лексической единице. Условие 7 ограничивает или нет порядок следования главной и подчиненной лексических единиц: 0 — ограничений нет, 1 — главная лексическая единица предшествует подчиненной лексической единице, -1 — главная лексическая единица следует после подчиненной.

Нужно отметить, что вместо слов входного текста вершинами графа являются их индексы. Это связано с тем, что одно и то же слово может встречаться несколько раз. С точки зрения семантики входного текста это будут разные единицы текста, по-разному связанные с другими словами текста.

**Пример.** Используя предложенные в предыдущих примерах множество значений морфологических признаков и словарь лексических фреймов, можно построить граф семантико-синтаксических связей для текста (в скобках даны индексы слов и конструктов) *Переместить(1) файл(2) "a.txt"(3) в(4) папку(5) "Документы"(6)*. Граф семантико-синтаксических связей будет выглядеть следующим обра-



зом (приведено только множество ребер, так как множество вершин очевидно:  $\{1, 2, 3, 5, 6\}$ ):

$\{(1, 2, (\text{нек перемещение}1, \text{нек файл}1, 0, (\text{нек } x1^*(\text{Объект}1, x2), x1, x2))),$   
 $(1, 5, (\text{нек перемещение}1, \text{нек папка}2, 4, (\text{нек } x1^*(\text{Назначение}1, x3), x2, x3))),$   
 $(2, 3, (\text{нек файл}1, "a.txt", 0, (\text{нек } x2^*(\text{Наименование}1, x4), x2, x4))),$   
 $(5, 6, (\text{нек папка}2, "Документы", 0, (\text{нек } x2^*(\text{Наименование}1, x4), x2, x4))),$   
 $(2, 6, (\text{нек файл}1, "Документы", 0, (\text{нек } x2^*(\text{Наименование}1, x4), x2, x4)))\}.$

### 5. Использование предложенных формальных понятий для разработки системы управления файлами NLC-2

Предложенная формальная модель лингвистической базы данных и понятие графа семантико-синтаксических связей позволили разработать новый алгоритм семантического анализа предписаний. Эти результаты послужили основой создания программы управления файловой системой Natural Language Commander — Version 2 (NLC-2). Эта программа является следующим поколением системы NLC-1, разработанной для исследований и экспериментов в области ЕЯИ приложений, основанных на действиях [31, 32]. Данная программа принимает на вход команды пользователя из некоторого ограниченного подмножества ЕЯ (русского, английского или другого). Далее программа с помощью алгоритма семантического анализа преобразует введенную пользователем команду в ее первичное КП и выполняет необходимые преобразования из базы преобразований. В результате строится вторичное КП, принадлежащее тому же СК-языку, что и исходное представление. Однако во вторичном КП сложные понятия, такие как "документ", "музыка", "видео", заменены более простыми понятиями с дополнительными квалификаторами. Например, "документ" интерпретируется как "файл с расширением doc, docx или odt". Такое преобразование возможно благодаря новым выразительным механизмам СК-языков. В частности, можно определить концептуальный базис  $B$  так, что СК-язык  $Ls(B)$  включает выражение

$(\text{документ}1 \equiv \text{файл}1 * (\text{Расширение-прогр},$   
 $("doc" \vee "docx" \vee "odt"))).$

Затем вторичное КП преобразуется в сценарий на языке командной оболочки операционной системы. NLC-2 использует для этих целей Borne Again Shell (BASH) как для POSIX-совместимых операционных систем, так и для Windows.

## Заключение

Статья предлагает новые формальные средства для разработки алгоритмов семантического анализа (семантического парсинга) предписаний на естественном языке (русском, английском и других). Предложенная математическая модель лингвистической базы данных (ЛБД) является значительно более общей, менее детальной по сравнению с двумя моделями ЛБД, рассматриваемыми в теории К-представлений. В новой модели уменьшено число используемых сущностей, в частности, за счет унификации словарей глагольно-предложных и предложных фреймов. Был рассмотрен новый подход к формальному описанию семантико-синтаксической структуры как промежуточной структуры при формировании К-представления исходного текста. С этой целью введено понятие графа семантико-синтаксических связей. Изложенные результаты создают основу для разработки алгоритма семантического анализа (парсинга) предписаний, обещающего быть весьма компактным по сравнению с другими алгоритмами, представленными в научной литературе. Предполагается, что для построения семантических представлений предписаний новый алгоритм будет использовать класс СК-языков (стандартных концептуальных языков), предложенный теорией К-представлений.

### Список литературы

1. Schank R. C., Goldman N. M., Rieger C. J., Riesbeck C. K. Conceptual Information Processing. Amsterdam, Oxford: North-Holland Publ. Company; New York: American Elsevier Publ. Comp., Inc., 1975. Перевод на русский язык: Шенк Р., Голдман Н. М., Ригер Ч. Дж., Ризбек К. К. Обработка концептуальной информации: Пер. с англ. / Под ред. В. М. Брябрина. М.: Энергия, 1980. 361 с.
2. Artzi Y., Zettlemoyer, L. Weakly Supervised Learning of Semantic Parsers for Mapping Instructions to Actions // Transactions of the Association for Computational Linguistics. 2013. N 1. P. 49–62. URL: <https://aclweb.org/anthology/Q/Q13/Q13-1005.pdf>, проверено 11.03.2016.
3. Babes-Vroman M., MacGlashan J., Gao R., Winner K., Ad-jogah R., desJardins M., Littman M., Muresan S. Learning to Interpret Natural Language Instructions // Proceedings of the Second Workshop on Semantic Interpretation in an Actionable Context, pages 1–6, Montreal, Canada, June 3–8, 2012. URL: <https://aclweb.org/anthology/W/W12/W12-2801.pdf>, проверено 11.03.2016.
4. Bastianelli E., Castellucci C., Croce D., Basili R. Textual Inference and Meaning Representation in Human Robot Interaction // Proceedings of the Joint Symposium on Semantic Processing. Textual Inference and Structures in Corpora, 2013. P. 65–69. URL: <https://aclweb.org/anthology/W/W13/W13-3820v2.pdf>, проверено 9.03.2016.
5. Marge M., Rudnicky A. Comparing Spoken Language Route Instructions for Robots across Environment Representations // Proceedings of the SIGDIAL 2010 Conference, pages 157–164, The University of Tokyo, September 24–25, 2010. P. 157–164. URL: <https://aclweb.org/anthology/W/W10/W10-4328.pdf>, проверено 10.03.2016.
6. Misra D. K., Tao K., Liang P., Saxena A. Environment-Driven Lexicon Induction for High-Level Instructions // Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Vol. 1: Long Papers), Beijing, China, July 26–31, 2015. P. 992–1002. URL: <https://aclweb.org/anthology/P/P15/P15-1096.pdf>, проверено 7.03.2016.
7. She L., Yang S., Cheng Y., Jia Y., Cha J., Xi N. Back to the Blocks World: Learning New Actions through Situated Human-Robot Dialogue // Proceedings of the 15th Annual Meeting of the Spe-

cial Interest Group on Discourse and Dialogue (SIGDIAL), Philadelphia, U. S.A., 18–20 June 2014. 2014. P. 89–97. URL: <https://aclweb.org/anthology/W/W14/W14-4313.pdf>, проверено 11.03.2016.

8. **Banarescu L., Bonial C., Cai S., Georgescu M., Griffitt K., Hermjakob U., Knight K., Koehn P., Palmer M., Schneider N.** Abstract Meaning Representation for Sembanking // Proceedings of the 7th ACL Linguistic Annotation Workshop and Interoperability with Discourse, Sofia, Bulgaria, August 8–9, 2013. URL: [www.aclweb.org/anthology/W13-2322](http://www.aclweb.org/anthology/W13-2322); проверено 12.03.2016.

9. **Benotti L., Villalba M., Lau T., Cerruti J.** Corpus-based Interpretation of Instructions in Virtual Environments // Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Vol. 2: Short Papers), pages 181–186, Jeju, Republic of Korea, 8–14 July 2012. P. 181–186. URL: <https://aclweb.org/anthology/P/P12/P12-2036.pdf>, проверено 8.03.2016.

10. **Carlos C. S.** Natural Language Programming Using Class Sequential Rules // Proceedings of 5th International Joint Conference on Natural Language Processing, Chiang Mai, Thailand, November 8–13, 2011. P. 237–245. URL: <https://aclweb.org/anthology/I/I11/I11-1027.pdf>, проверено 13.03.2016.

11. **Lei T., Long F., Barzilay R., Rinard M.** From Natural Language Specifications to Program Input Parsers // Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1294–1303, Sofia, Bulgaria, August 4–9 2013. P. 1294–1303. URL: <https://aclweb.org/anthology/P/P13/P13-1127.pdf>, проверено 10.03.2016.

12. **Siri (Электронный ресурс).** URL: <http://www.apple.com/ios/siri/> (проверено 17.02.2014)

13. **Sobesednik HD (Электронный ресурс):** URL: <http://sobesednik.idevelops.com/> (проверено 17.12.2014)

14. **Braina Project (Электронный ресурс).** URL: <http://www.brainasoft.com/braina/#overview> (проверено 17.12.2014)

15. **NLUI Server (Электронный ресурс):** URL: [http://www.linguasys.net/web\\_production/server-item/NLUI%20Server](http://www.linguasys.net/web_production/server-item/NLUI%20Server) (проверено 17.12.2014)

16. **Фомичев В. А.** Представление информации средствами К-исчислений: Учебное пособие. М.: Московский институт электронного машиностроения (МИЭМ), 1988.

17. **Fomichov V. A.** Integral Formal Semantics and the Design of Legal Full-Text Databases // Cybernetica. Quarterly Review of the International Association for Cybernetics (Belgium, Namur). 1994. Vol. 37, N. 2. P. 145–177.

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

19. **Fomichov V. A.** Theory of Restricted K-calculus as a Comprehensive Framework for Constructing Agent Communication Languages // Fomichov V. A., Zeleznikar A. P. (eds.). Special Issue on NLP and Multi-Agent Systems. Informatica. An Intern. Journal of Computing and Informatics (Slovenia). 1998. V. 22. N. 4. P. 451–463.

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

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

22. **Фомичев В. А.** Понятие текстообразующей системы как компонент нового формального аппарата для проектирования лингвистических процессоров // Информационные технологии. 2005. № 8. С. 22–27.

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

24. **Фомичев В. А.** Формализация проектирования лингвистических процессоров. М.: МАКС Пресс. 2005. 368 с.

25. **Фомичев В. А.** Математические основы представления содержания посланий компьютерных интеллектуальных агентов. М.: ГУ-ВШЭ, издательство "ТЕИС". 2007. 176 с.

26. **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. Vol. 1, N. 1. P. 143–163.

27. **Fomichov V. A.** Semantics-Oriented Natural Language Processing: Mathematical Models and Algorithms. New York, Dordrecht, Heidelberg, London: Springer, 2010. 354 p.

28. **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. Vol. 34, N. 3. P. 387–396.

29. **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. V. 8644. Cham: Springer International Publishing Switzerland. 2014. P. 394–401.

30. **Захлебин И. В., Фомичев В. А.** Разработка метода семантического поиска специалистов в корпоративной базе данных по естественно-языковым запросам // Информационные технологии. 2015. Т. 21, № 5. С. 323–331.

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. Vol. 31. P. 1005–1011; open access, URL: <http://authors.elsevier.com/sd/article/S1877050914005304>.

32. **Фомичев В. А., Разоренов А. А.** Значение теории К-представлений для исследований по автоматическому выявлению семантических ролей // Информационные технологии. 2015. Т. 21, № 6. С. 403–411.

**A. A. Razorenov**, Ph. D. student, e-mail: [arazorenov@hse.ru](mailto:arazorenov@hse.ru),

**V. A. Fomichov**, Professor of Computer Science, e-mail: [vfomichov@hse.ru](mailto:vfomichov@hse.ru),

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

## Compact Formalization of Instructions Semantic Parsing Algorithms' Input and Intermediary Data Structures

*The paper suggests new formal means for the development of the algorithms fulfilling semantic processing of natural language instructions. The theory of K-representations (knowledge representations) is chosen as the methodological basis of the study. Nowadays it is the only theory being convenient for constructing semantic representations of arbitrary complex natural language instructions. The following theoretical results are set forth: (a) a new formal model of a linguistic database is developed, this model is considerably more compact than two models introduced by the theory of K-representations; (b) the notion of semantic-syntactic representation of a natural language text is formalized in a new way. The developed formal means are used in the design of a useful for practice natural language interface of file manager.*

**Keywords:** natural language processing, semantic parsing of instructions, semantic representation, theory of K-representations, SK-language, morphological basis, template of a K-representation, dictionary of lexical frames, linguistic database, natural language control of file system, Natural Language Commander

## References

1. **Schank R. C., Goldman N. M., Rieger C. J., Riesbeck C. K.** *Conceptual Information Processing*. Amsterdam, Oxford: North-Holland Publ. Company; New York: American Elsevier Publ. Comp., Inc., 1975. Перевод на русский язык: Шенк Р., Голдман Н. М., Ригер Ч. Дж., Ризбек К. К. Обработка концептуальной информации. Пер. с англ. Под ред. В. М. Брябрина. М.: Энергия, 1980. 361 с.
2. **Artzi Y., Zettlemoyer L.** Weakly Supervised Learning of Semantic Parsers for Mapping Instructions to Actions. *Transactions of the Association for Computational Linguistics*, 2013, no. 1, pp. 49–62. URL: <https://aclweb.org/anthology/Q/Q13/Q13-1005.pdf>, retrieved 11.03.2016.
3. **Babes-Vroman M., MacGlashan J., Gao R., Winner K., Adjogah R., desJardins M., Littman M., Muresan S.** Learning to Interpret Natural Language Instructions, *Proceedings of the Second Workshop on Semantic Interpretation in an Actionable Context*, Montreal, Canada, June 3–8, 2012, pp. 1–6. URL: <https://aclweb.org/anthology/W/W12/W12-2801.pdf>, retrieved 11.03.2016.
4. **Bastianelli E., Castellucci C., Croce D., Basili R.** Textual Inference and Meaning Representation in Human Robot Interaction, *Proceedings of the Joint Symposium on Semantic Processing. Textual Inference and Structures in Corpora*, 2013, pp. 65–69. URL: <https://aclweb.org/anthology/W/W13/W13-3820v2.pdf>, retrieved 9.03.2016.
5. **Marge M., Rudnicky A.** Comparing Spoken Language Route Instructions for Robots across Environment Representations, *Proceedings of the SIGDIAL 2010 Conference*, The University of Tokyo, September 24–25, 2010, pp. 157–164. URL: <https://aclweb.org/anthology/W/W10/W10-4328.pdf>, retrieved 10.03.2016.
6. **Misra D. K., Tao K., Liang P., Saxena A.** Environment-Driven Lexicon Induction for High-Level Instructions, *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Beijing, China, July 26–31, 2015, pp. 992–1002. URL: <https://aclweb.org/anthology/P/P15/P15-1096.pdf>, retrieved 7.03.2016.
7. **She L., Yang S., Cheng Y., Jia Y., Cha J., Xi N.** Back to the Blocks World: Learning New Actions through Situated Human-Robot Dialogue, *Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, Philadelphia, U. S. A., 18–20 June 2014, pp. 89–97. URL: <https://aclweb.org/anthology/W/W14/W14-4313.pdf>, retrieved 11.03.2016.
8. **Banarescu L., Bonial C., Cai S., Georgescu M., Griffitt K., Hermjakob U., Knight K., Koehn P., Palmer M., Schneider N.** Abstract Meaning Representation for Sembanking, *Proceedings of the 7th ACL Linguistic Annotation Workshop and Interoperability with Discourse*, Sofia, Bulgaria, August 8–9, 2013. URL: [www.aclweb.org/anthology/W13-2322](http://www.aclweb.org/anthology/W13-2322); retrieved 12.03.2016.
9. **Benotti L., Villalba M., Lau T., Cerruti J.** Corpus-based Interpretation of Instructions in Virtual Environments, *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Vol. 2: Short Papers)*, Jeju, Republic of Korea, 8–14 July 2012, pp. 181–186. URL: <https://aclweb.org/anthology/P/P12/P12-2036.pdf>, retrieved 8.03.2016.
10. **Carlos C. S.** Natural Language Programming Using Class Sequential Rules, *Proceedings of 5th International Joint Conference on Natural Language Processing*, Chiang Mai, Thailand, November 8–13, 2011, pp. 237–245. URL: <https://aclweb.org/anthology/I/I11/I11-1027.pdf>, retrieved 13.03.2016.
11. **Lei T., Long F., Barzilay R., Rinard M.** From Natural Language Specifications to Program Input Parsers, *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Sofia, Bulgaria, August 4–9 2013, pp. 1294–1303. URL: <https://aclweb.org/anthology/P/P13/P13-1127.pdf>, retrieved 10.03.2016.
12. **Home Page Siri**, URL: <http://www.apple.com/ios/siri/> — Electronic resource (retrieved 17.12.2014).
13. **Home Page Sobesednik HD**, URL: <http://sobesednik.idevelops.com/> — Electronic resource (retrieved 17.12.2014).
14. **Braina Project Home Page**, URL: <http://www.brainasoft.com/braina/#overview> — Electronic resource (retrieved 17.12.2014).
15. **NLUI Server Home Page**, URL: [http://www.linguasys.net/web\\_production/server-item/NLUI%20Server](http://www.linguasys.net/web_production/server-item/NLUI%20Server) — Electronic resource (retrieved 17.12.2014).
16. **Fomichov V. A.** *Representing Information by Means of K-Calculuses: Textbook*, Moscow, Moscow Institute of Electronic Engineering, 1988.
17. **Fomichov V. A.** Integral Formal Semantics and the Design of Legal Full-Text Databases, *Cybernetica. Quarterly Review of the International Association for Cybernetics* (Belgium, Namur), 1994, vol. 37, no. 2, pp. 145–177.
18. **Fomichov V. A.** A Mathematical Model for Describing Structured Items of Conceptual Level, *Informatica. An Intern. Journal of Computing and Informatics* (Slovenia), 1996, vol. 20, no. 1, pp. 5–32.
19. **Fomichov V. A.** Theory of Restricted K-calculuses as a Comprehensive Framework for Constructing Agent Communication Languages. Fomichov V. A., Zeleznikar A. P. (eds.). Special Issue on NLP and Multi-Agent Systems. *Informatica, An Intern. Journal of Computing and Informatics* (Slovenia), 1998, vol. 22, no. 4, pp. 451–463.
20. **Fomichov V. A.** Matematicheskie osnovy predstavleniya smysla tekstov dlya razrabotki lingvisticheskikh informatsionnykh tekhnologii. Part I, *Informatsionnye Tekhnologii*, 2002, no. 10, pp. 16–25 (in Russian).
21. **Fomichov V. A.** Matematicheskie osnovy predstavleniya smysla tekstov dlya razrabotki lingvisticheskikh informatsionnykh tekhnologii. Part II, *Informatsionnye Tekhnologii*, 2002, no. 11, pp. 34–45 (in Russian).
22. **Fomichov V. A.** Ponyatie tekstoobrazuyshchei sistemy kak component formalnogo apparata dlya proektirovaniya lingvisticheskikh protessorov, *Informatsionnye Tekhnologii*, 2005, no. 8, pp. 22–27 (in Russian).
23. **Fomichov V. A.** Novyi metod preobrazovaniya yestestvennoyazykovykh tekstov v semanticheskie predstavleniya, *Informatsionnye Tekhnologii*, 2005, no. 10, pp. 25–35 (in Russian).
24. **Fomichov V. A.** *Formalizatsiya proektirovaniya lingvisticheskikh protessorov*. Moscow: MAKS Press, 2005. 368 p.
25. **Fomichov V. A.** *Matematicheskie osnovy predstavleniya sodержaniya poslanii kompyuternykh intellektualnykh agentov*. Moscow: State University — Higher School of Economics, Publishing House "TEIS". 2007. 176 p.
26. **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, vol. 1, no. 1, pp. 143–163.
27. **Fomichov V. A.** *Semantics-Oriented Natural Language Processing: Mathematical Models and Algorithms*. New York, Dordrecht, Heidelberg, London: Springer, 2010. 354 p.
28. **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, vol. 34, no. 3, pp. 387–396.
29. **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, vol. 8644. Cham: Springer International Publishing Switzerland, 2014, pp. 394–401.
30. **Zakhlebin I. V., Fomichov V. A.** Development of a method for semantic search of specialists in corporate databases using natural language queries, *Informatsionnye Tekhnologii*, 2015, vol. 21, no. 5, pp. 323–331 (in Russian).
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, 2014, vol. 31, pp. 1005–1011; open access, URL: <http://authors.elsevier.com/sd/article/S1877050914005304>.
32. **Fomichov V. A., Razorenov A. A.** The Significance of the K-Representations Theory for the Studies in Automatic Semantic Role Labeling, *Informatsionnye Tekhnologii*, 2015, vol. 21, no. 6, pp. 403–411 (in Russian).

А. С. Домников, канд. техн. наук, доц., asdomnikoff@mail.ru,

В. В. Белоус, канд. техн. наук, доц., walentina.belous@bmstu.ru

Московский государственный технический университет им. Н. Э. Баумана, Москва, Россия

## Синтез рационального упорядочения учебных модулей методом Бержа

*Рассматривается задача построения рационального упорядочения учебных модулей электронного курса. Исследованы особенности этой задачи и сделан вывод о невозможности ее решения классическими методами математического программирования и математической статистики. Исходную информацию для синтеза рационального порядка учебных модулей дает экспертный опрос или данные, заключенные в гипертекстовой структуре ссылок. Рассматриваются два типа экспертной информации: четкое и нечеткое бинарные отношения предпочтения на множестве учебных модулей. Предлагается способ сведения всех трех структур к нормализованному виду — матрице парных сравнений с простой калибровкой. Решать задачу упорядочения предлагается с помощью метода Бержа, который прошел широкую апробацию в различных прикладных областях.*

**Ключевые слова:** электронный курс, упорядочение, теория принятия решений, бинарные отношения, нечеткое бинарное отношение, дискретное математическое программирование, калибровочное условие, метод Бержа, матрица парных сравнений

### Введение

В наше время происходит бурное развитие электронных обучающих технологий и их экспансия в различные образовательные отрасли и учреждения. Электронное образование — совокупность методов и технических средств, предназначенных для дистанционного синтеза и приобретения знаний. Оно базируется на возможностях современной вычислительной техники и информационно-коммуникационных технологиях и обладает множеством преимуществ по сравнению с традиционными образовательными методиками — это, прежде всего, интерактивность, доступность, оперативность, гибкость, мобильность, экономичность, массовость, возможность создания единых образовательных сред, доступ к большим электронным библиотекам и базам данных и др. [1, 2].

Массовое внедрение электронных образовательных технологий в учебный процесс требует решения множества методических, педагогических, инженерных и научных задач. Основной массив публикаций по этой тематике посвящен обсуждению педагогических проблем и реализации методических приемов и прикладных дисциплин в конкретных программных средах, например Moodle или Sakai [1, 3]. В публикациях на русском языке достаточно полно освещена задача генерации модульного состава электронных образовательных дисциплин [4]. Важнейшая проблема синтеза рациональной модульной структуры учебных курсов находится на периферии исследователей. Работы [2, 5, 6] составляют немногочисленные исключения из этого правила.

В статье рассматривается проблема построения линейной последовательности заданного множества электронных учебных модулей. Предлагается

метод решения этой задачи, общий для трех различных источников исходной информации о парных предпочтениях отдельных модулей.

### Электронные учебные модули

Современные системы электронного образования основаны на модульной парадигме организации учебных курсов. Учебный материал делится на относительно замкнутые фрагменты, которые могут включать в себя текстовые, графические, мультимедийные блоки и содержать ссылки на другие внешние и внутренние фрагменты. Среди специалистов в области педагогики и электронного обучения существуют некоторые разногласия по поводу точного определения учебного модуля. Чаще всего модулем считается относительно замкнутый фрагмент учебного контента, дающий достаточно полное толкование некоторого вопроса или темы и имеющий минимальное число внешних смысловых отсылок.

В современных системах электронного обучения модули хранятся в общем репозитории и служат элементарными разделяемыми единицами контента для различных учебных материалов, которые собираются из этих элементарных блоков. Стандарт SCORM (Sharable Content Object Reference Model), который является стандартом де-факто для современных систем в области электронного образования, ориентирован на глубокую структуризацию учебных материалов и поддерживает их модульную структуру.

Модульная организация образовательного контента имеет несколько бесспорных преимуществ, главными из которых являются:

- гибкость и высокая совместимость;
- реентерабельность модулей;
- возможность организации гипертекстовых учебных курсов.

Все современные электронные курсы снабжены гипертекстовой ссылочной разметкой, которая допускает нелинейную последовательность изучения с возможными возвратами и ревизией ранее пройденного материала. Однако линейная упорядоченность, когда порции учебного материала следуют последовательно одна за другой, имеет большое число методических и организационных преимуществ по сравнению с другими "маршрутами обучения". Так, в форме линейного упорядочения модулей учебные курсы представляются в методических документах, учебных программах, рекламных материалах, индивидуальных планах и др. Эта форма используется в оперативном управлении учебным процессом, при выборе индивидуальной стратегии обучения, для решения логистических и организационных задач, обеспечивающих учебный процесс [6].

**Особенности рационального упорядочения учебных модулей.** Рациональное упорядочение элементов дискретного множества — это хорошо изученная задача математического программирования и теории расписаний [7—9]. Однако попытка применения классического аппарата для упорядочения электронных учебных модулей сталкивается с принципиальными затруднениями, причинами которых являются особенности этой задачи. Перечислим главные.

- **Сложность.** В общем случае электронные учебные курсы имеют сложную многосвязную структуру, состоящую из нескольких десятков модулей и отличающуюся высокой плотностью связей.
- **Структурная динамичность.** Структура учебного курса может динамически перестраиваться. Это может вызываться различными причинами: новыми научными данными, публикацией новаторских учебных материалов, модификацией методических установок и приемов и др.
- **Структурная неопределенность.** Сведения о разбиении курсов на модули и связях между ними формируются на основании экспертных опросов. Любые, даже самые искусные, процедуры экспертного опроса не могут преодолеть субъективность экспертного знания, которая проявляется в принципиально неустранимой неопределенности формируемых структур.
- **Высокая цикломатика.** Множество связей, задающих парное упорядочение модулей, порождает в исходной структуре большое число ориентированных циклов, для элементов которых не может быть построена непротиворечивая ранжировка.
- **Структурная неоднородность.** Структура модулей учебного курса отличается принципиальной неоднородностью. В общем случае эта структура может состоять из нескольких компонентов связности и включать в себя обширные линейные и многосвязные фрагменты.

"Мягкие" модели, основанные на парадигме теории принятия решений и теории нечетких мно-

жеств, полнее учитывают своеобразие задачи упорядочения учебных модулей и лучше подходят для ее решения.

## Постановка задачи

Для синтеза рационального упорядочения учебных модулей можно использовать различную исходную информацию. Таковой может быть мнение эксперта (сообщества экспертов), который на основе анализа множества модулей учебной программы формирует систему парных предпочтений предшествования. Эти предпочтения могут быть представлены на языке четких бинарных или нечетких бинарных отношений и записаны в виде матрицы парных сравнений или графа предпочтений.

Пусть нужно решить задачу формирования состава модулей некоторого электронного учебного курса или программы. Обозначим через  $X = \{x_i\}_{i=1}^n$  множество модулей курса. Будем считать, что проведен опрос эксперта и получены экспертные данные о парных предпочтениях на множестве  $X$ . Представим эти данные в виде матрицы парных сравнений  $A = \|a_{ij}\|$ , где

$$a_{ij} = \begin{cases} 1, & \text{если } x_i \text{ предшествует } x_j, \\ 0 & \text{в противном случае.} \end{cases}$$

Легко видеть, что матрицу  $A$  можно рассматривать как матрицу смежности некоторого графа  $G = (X, D)$ , у которого  $X = \{x_i\}_{i=1}^n$ ,  $D = \{d_j\}$  — множество вершин и дуг соответственно. Дуга  $d = (x_i, x_j)$  соединяет вершину  $x_i$  с вершиной  $x_j$  тогда и только тогда, когда  $a_{ij} = 1$ . Граф  $G$  будем называть *графом предпочтений*. Требуется найти такую ранжировку множества модулей (вершин графа)  $X$ , которая наилучшим образом согласуется со структурой предпочтений эксперта, заданной в виде матрицы  $A$  или графа  $G$ .

Во многих ситуациях экспертная информация о парных предшествованиях модулей задается в виде нечеткого бинарного отношения предпочтения с функцией принадлежности  $\mu: X \times X \rightarrow [0, 1]$ . Тогда элементы  $a_{ij}$  матрицы парных сравнений  $A$  получают значения функции принадлежности нечеткого отношения предпочтения  $\mu_{ij} = \mu(x_i, x_j) \in [0, 1]$ , а граф предпочтений  $G = (X, D)$  формируется по следующим правилам: дуга  $d = (x_i, x_j)$  соединяет вершину  $x_i$  с вершиной  $x_j$  тогда и только тогда, когда  $\mu_{ij} > 0$ . Дуги  $d \in D$  взвешиваются значениями функции принадлежности.

Иной тип исходных данных дают гипертекстовые или гиперссылочные структуры на множестве учебных модулей, которые являются отражением понятийной и терминологической организации предметной области. Не будет преувеличением сказать, что все современные электронные учебные курсы разработчики снабжают развитой системой гиперссылочных команд. Они реализуют смысловые связи между отдельными единицами контента:

терминами, определениями, текстовыми фрагментами, глоссарием, алфавитным указателем и др. Ссылочная система строится на более объективных основаниях, нежели экспертные предпочтения, поскольку отражает понятийную структуру данной предметной области.

Не будем учитывать гиперссылки технического характера (например, на подрисовочные подписи, номера таблиц, формулы и др.) и ограничимся рассмотрением ссылок текстовых фрагментов, которые выражают интенциональные порядковые отношения между отдельными модулями. Иными словами, если существует ссылка из модуля  $u$  в модуль  $x$ , то это означает, что для освоения  $u$  необходимо знать материал, изложенный в  $x$ , т. е.  $x$  должен предшествовать  $u$  в процессе обучения.

Пусть  $X = \{x_i\}_{i=1}^n$  — множество модулей, а структура электронного курса задана в виде графа  $G = (X, D)$ , в котором вершины представляют модули, а дуга  $d = (x, y) \in D$  соединяет вершину  $x$  с вершиной  $y$  тогда и только тогда, когда существует гиперссылка из модуля  $y$  на модуль  $x$ . В больших электронных курсах, где плотность связей высока, могут существовать кратные взаимные гиперссылки модулей. В этом случае структура электронного курса задается в виде взвешенного ориентированного графа  $G = (X, D)$ , в котором каждой дуге  $d = (x, y) \in D$  сопоставляется число  $a_{xy}$ , равное числу гиперссылок, связывающих  $y$  с  $x$ .

Задача синтеза рационального упорядочения модулей заключается в поиске такого упорядочения (частичного или линейного) множества модулей  $X = \{x_i\}_{i=1}^n$ , которое наилучшим образом согласуется с исходной структурой предпочтений  $G = (X, D)$ .

Структуры предпочтений будем обозначать далее следующим образом:  $G_1$  — четкое отношение предпочтения;  $G_2$  — нечеткое отношение предпочтения;  $G_3$  — гиперссылочная структура.

Постановка задачи рационального упорядочения учебных модулей как задачи поиска наилучшей аппроксимации структуры предпочтений отличается большой гибкостью, поскольку допускает применение большого числа различных способов решения, заимствованных из различных разделов дискретной математики и информационных технологий. Например, ее можно свести к следующим задачам:

- задача о наилучшей приближенной триангуляции матрицы [10];
- задача поиска асимметричной части бинарного отношения предпочтения или квазипорядка [4];
- задача дискретного математического программирования [6];
- задача принятия решений [9] и др.

### Синтез рационального упорядочения модулей

В предыдущих публикациях авторов рассматривалась задача рационального упорядочения учебных модулей в различных постановках. Так, поиск

наилучшей линейной аппроксимации для четкой структуры предпочтений  $G_1$  обсуждался в работе [5]. Задача решалась с помощью аппарата бинарных отношений и дискретного математического программирования. Исходная структура  $G_1$  рассматривалась как бинарное отношение предпочтения, из которого выделялись асимметричная  $G_{1a}$  и симметричная  $G_{1c}$  части:  $G_1 = G_{1a} + G_{1c}$ . Транзитивное замыкание  $T(G_{1a})$  асимметричной части отношения представляет собой частичный порядок на множестве модулей, который может быть продолжен до линейного порядка.

В [6] предложена модель линейного упорядочения для обработки гипертекстовой структуры предпочтений  $G_3$ . В этой статье задача решалась с помощью классического метода максимального согласования. Исходная структура предпочтений в общем случае представляется в виде сильно связанного графа  $G = (X, D)$ , который включает в себя набор контуров (ориентированных циклов). В методе максимального согласования граф предпочтений приводится к бесконтурному виду. Для этого ищется минимальное множество дуг, которые принадлежат всем контурам графа, и эти дуги исключаются из  $G$ . В результате получается бесконтурный граф, который допускает непротиворечивое упорядочение вершин многими разными способами, например с помощью алгоритмов топологической сортировки Тарьяна или Кана [7].

Синтез рационального упорядочения модулей в нечеткой постановке (для структуры  $G_2$ ) рассмотрен в [6]. В этой статье предложены несколько способов синтеза рациональных ранжировок, которые учитывают расплывчатый характер исходной информации: упорядочение на основе функции доминированности, синтез упорядочения разложением нечеткого множества на уровни и сведение задачи к задаче дискретного математического программирования.

В данной статье для обработки всех трех исходных структур предпочтений  $G_1 - G_3$  предлагается использовать один метод.

В любой задаче принятия решений элементы матрицы парных сравнений  $A = \|a_{ij}\|_{n \times n}$  (матрицы смежности графа  $G$ ) не могут принимать произвольные значения. Они должны подчиняться ограничениям, которые в теории принятия решений называются калибровочными условиями, или калибровками [10]. Калибровочное условие зависит от способа извлечения исходной информации и решающим образом влияет на выбор метода рационального упорядочения.

Основными калибровками являются следующие:

- простая калибровка (ПК):

$$\forall i, j, i \neq j, a_{ij} = \begin{cases} 1, & \text{если } x_i > x_j; \\ 0, & \text{если } x_i < x_j; \end{cases}$$

- турнирная калибровка (Т):

$$\forall i, j, a_{ij} \geq 0, a_{ij} + a_{ji} = \text{const};$$

- степенная калибровка (С):

$$\forall i, j, a_{ij} > 0, a_{ij} \times a_{ji} = 1;$$

- кососимметрическая калибровка (К):

$$\forall i, j, a_{ij} + a_{ji} = 0;$$

- вероятностная калибровка (В):

$$\forall i, j, 0 \leq a_{ij} \leq 1, a_{ij} + a_{ji} = 1.$$

Если элементы матрицы парных сравнений принимают произвольные целочисленные значения, то исходную структуру предпочтения называют *взвешенной структурой*.

Легко видеть, что четкое отношение предпочтения  $G_1$  имеет простую калибровку (ПК), нечеткое отношение предпочтения  $G_2$  не подчиняется ни одной калибровке, а гиперссылочная структура  $G_3$  представляет собой произвольный граф, взвешенный целыми числами (взвешенная структура).

Рассмотрим метод построения рационального порядка, который способен обработать любую из рассмотренных исходных структур  $G = (X, D)$ . В публикациях по теории принятия решений этот метод называется методом Бержа или Брука—Буркова. Метод был испытан на множестве различных прикладных задач в экономике и показал свою адекватность и высокую эффективность. Он удовлетворяет большей части аксиом рационального упорядочения критериев, которые используются в теории принятия решений для проверки качества методов ранжировки. Обязательное условие применения метода — простая калибровка матрицы парных сравнений [10].

Структура  $G_3$ , ребра которой взвешены произвольными целыми числами, сводится к простой калибровке следующим образом:

$$b_{ij} = [\text{sign}(a_{ij} - a_{ji}) + 1], \forall i, j, i \neq j. \quad (1)$$

Структуру  $G_2$  можно привести к вероятностной калибровке с помощью простого преобразования, которое не вносит значительных искажений в исходные предпочтения. Пусть  $a_{ij}$  и  $a_{ji}$  — любые два элемента матрицы парных сравнений  $A = \|a_{ij}\|_{n \times n}$ , которая описывает  $G_2$ . По определению  $0 \leq a_{ij}, a_{ji} \leq 1$ . Преобразуем элементы этой матрицы следующим

образом:  $\forall i, j, i \neq j, b_{ij} = \frac{a_{ij}}{a_{ij} + a_{ji}}$ . Легко видеть, что

$b_{ij} + b_{ji} = 1$ , т. е. элементы матрицы  $B = \|b_{ij}\|_{n \times n}$  удовлетворяют вероятностной калибровке. Модифицированная матрица парных  $B$  сравнений приводится к простой калибровке посредством преобразования (1), что дает возможность решать данную задачу методом Бержа.

Метод Бержа состоит из двух этапов: грубое ранжирование; тонкое ранжирование. Напомним основные положения *грубого ранжирования*. На этом этапе исходная структура предпочтений  $G = (X, D)$

разбивается на совокупность бикомпонент. Бикомпонентой называется максимальное по включению множество вершин графа, любая пара элементов которого состоит в отношении взаимной достижимости. Деление на бикомпоненты является разбиением множества вершин, т. е. задает отношение эквивалентности. Факторизация отношения предпочтения  $G = (X, D)$  по данной эквивалентности представляет собой ациклическое отношение (ациклический граф), которое допускает непротиворечивую топологическую сортировку своих вершин. Результатом грубого ранжирования является частичный порядок, построенный на бикомпонентах графа  $G = (X, D)$ .

На этапе *тонкого ранжирования* ищется упорядочение вершин каждой бикомпоненты. Если такое упорядочение будет построено, то "склеивание" двух порядков (грубого и тонкого) даст результирующую ранжировку. Все операции грубого ранжирования хорошо изучены и имеют эффективные алгоритмы решения.

Рассмотрим этап тонкого ранжирования, который по сравнению с грубым ранжированием отличается достаточно громоздким обоснованием и вычислительной сложностью. Пусть  $C$  — некоторая бикомпонента графа  $G$ . Перенумеруем вершины графа таким образом, чтобы элементы бикомпоненты  $C$  получили номера от 1 до  $r$ . Обозначим  $M = \|a_{ij}\|_{r \times r}$  часть матрицы парных сравнений, которая соответствует объектам бикомпоненты  $C$ . Будем использовать спортивные аналогии и вместо громоздких терминов "доминирование" и "число доминирований" будем говорить о победе и силе объекта. Силой первого порядка объекта  $x_i, i = 1, \dots, r$ , назовем

строчную сумму  $\sum_{i=1}^r a_{ij}$  в матрице  $M$ . Будем говорить, что объект  $x_i$  одержал победу второго порядка над  $x_j$ , если существует объект  $x_k$  такой, что  $x_i \geq x_k$  и  $x_k \geq x_j$ . Числом побед второго порядка, одержанных  $x_i$  над  $x_j$ , называется сумма

$$\sum_{k=1}^r a_{ik}a_{kj}. \quad (2)$$

Силой второго порядка объекта  $x_i$  назовем общее число побед второго порядка, одержанных этим объектом над всеми иными объектами, т. е. величину

$$\sum_{j=1}^r \sum_{k=1}^r a_{ik}a_{kj}. \quad (3)$$

Легко видеть, что выражение (2) есть число, стоящее на пересечении  $i$ -й строки и  $j$ -го столбца матрицы  $M^2 = M \times M$ , а выражение (3) — сумма всех элементов  $i$ -й строки в этой матрице.

По аналогии можно определить силу третьего, четвертого и так далее порядков объекта  $i$  как строчные суммы в матрицах  $M^3, M^4, \dots$ . Обозначим через  $d_i^n$  силу  $n$ -го порядка объекта  $i$ . Под относительной силой  $n$ -го порядка понимается величина

$$t_i^n = \frac{d_i^n}{\sum_{k=1}^r d_k^n}$$

$t_i = \lim_{n \rightarrow \infty} t_i^n$ , который называется относительной силой объекта  $x_i$ , а вектор  $t = (t_1, t_2, \dots, t_r)$  — предельным вектором.

Тонкое ранжирование объектов, принадлежащих одной бикомпоненте, выполняется по значениям предельного вектора по следующему простому правилу:  $x_i \geq x_j \Leftrightarrow t_i \geq t_j$ . Оказалось, что предельный вектор является собственным вектором матрицы парных сравнений  $M = \|a_{ij}\|_{r \times r}$ , что позволяет искать этот вектор не как предел, а использовать для этого глубоко разработанные методы вычислительной математики [11, 12].

Описанный метод тонкого ранжирования отличается высокой чувствительностью к деталям исходной структуры предпочтений. Так, он учитывает не только прямую силу, т. е. число побед, которые одержал объект, но и опосредованные победы разной длины. При этом победы над сильными объектами ценятся выше, и это различное качество подсчитывается автоматически, без использования различных весовых коэффициентов.

### Пример

Рассмотрим применение этого метода для обработки структуры предпочтений, показанной на рис. 1. На этом рисунке показаны отношения предшествования, которые, по мнению эксперта, существуют между отдельными учебными модулями, входящими в дисциплину "искусственный интеллект".

Матрица парных сравнений  $A = \|a_{ij}\|_{11 \times 11}$  этой структуры показана в табл. 1.



Рис. 1. Структура предпочтений модулей дисциплины "искусственный интеллект"

Разбиение данной структуры на сильносвязные фрагменты показало, что она состоит из трех бикомпонент  $C_1 = \{5\}$ ,  $C_2 = \{2, 3, 4, 5, 6, 7, 8, 9, 10, 11\}$  и  $C_3 = \{1\}$ , грубое ранжирование которых представляется следующей схемой:  $\{5\} \rightarrow \{2, 3, 4, 5, 6, 7, 8, 9, 10, 11\} \rightarrow \{1\}$ .

Рассмотрим тонкое ранжирование бикомпоненты  $C_2 = \{2, 3, 4, 5, 6, 7, 8, 9, 10, 11\}$ . Матрица парных сравнений  $M = \|a_{ij}\|_{9 \times 9}$  этого фрагмента приведена в табл. 2.

Нахождение предельного вектора — это весьма трудоемкая вычислительная задача. Вместо относительных сил объектов подсчитаем их силы конечного порядка как строчные суммы в матрицах  $M, M^2, M^3, \dots, M^9$ . Сведем результаты в табл. 3. В этой таблице число, стоящее на пересечении  $i$ -й строки и  $j$ -го столбца, есть сила  $j$ -го порядка объ-

Таблица 1  
Матрица парных сравнений исходной структуры предпочтений

	1	2	3	4	5	6	7	8	9	10	11
1											
2	1		1							1	
3		1									
4										1	1
5										1	1
6									1	1	1
7		1		1		1		1	1		1
8		1				1					1
9						1					
10								1			
11											1

Таблица 2  
Матрица парных сравнений бикомпоненты  $C_2$

	2	3	4	6	7	8	9	10	11
2		1						1	
3	1								
4								1	1
6								1	1
7	1		1	1		1	1		1
8	1			1					1
9				1		1			
10									1
11						1			

Таблица 3  
Силы конечного порядка модулей

	$d_i^1$	$d_i^2$	$d_i^3$	$d_i^4$	$d_i^5$	$d_i^6$	$d_i^7$	$d_i^8$	$d_i^9$
2	2	2	3	8	15	30	61	135	278
3	1	2	2	3	8	15	30	61	135
4	2	2	7	18	34	68	154	341	672
6	2	2	7	18	34	68	151	322	665
7	6	12	22	46	105	217	448	946	2010
8	3	5	10	22	48	95	203	429	905
9	2	5	7	17	40	82	163	354	751
10	1	1	6	12	22	46	105	217	448
11	1	6	12	22	46	105	217	448	946



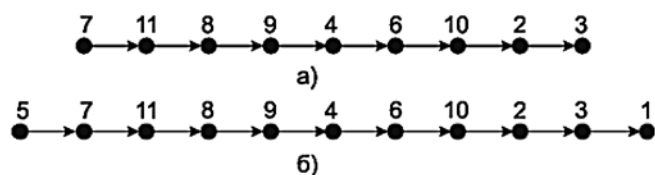


Рис. 2. Тонкое ранжирование (а) и итоговое упорядочение модулей (б)

екта с номером  $i$ . Построим упорядочение объектов не по значениям предельного вектора, а по силам конечного порядка.

Как видно из табл. 3, столбцы  $d^k$  при  $k = 1, 2, \dots, 6$  имеют совпадающие значения, что не дает возможности синтезировать строгое упорядочение объектов. Все значения в столбцах  $d^7, d^8$  и  $d^9$  различаются, что позволяет построить тонкое ранжирование, показанное на рис. 2, а.

Объединив результаты грубого и тонкого ранжирования, получим линейное упорядочение, представленное на рис. 2, б.

### Перспективы дальнейших исследований

Описанные результаты не исчерпывают, конечно, всей глубины решаемой проблемы. Представляется перспективной попытка построения некоторой интегральной структуры  $G = G_1 \oplus G_2 \oplus G_3$ , которая учитывает информацию о парных предпочтениях, хранящуюся в четкой  $G_1$ , нечеткой  $G_2$  и гипертекстовой  $G_3$  структурах. В простейшем случае интегральная структура может представлять собой простую взвешенную сумму матриц смежности, задающих  $G_1, G_2, G_3$  или только  $G_2$  и  $G_3$ , если четкое исходное предпочтение считать частным случаем нечеткого.

Еще одно направление исследований, которое представляется весьма перспективным для решения поставленной задачи, состоит в использовании аппарата принятия групповых решений. Обозначим через  $\mathcal{M}_i$  некоторый метод синтеза рационального упорядочения, который может быть применен для обработки исходной структуры предпочтений  $G$  непосредственно или после предварительной модификации калибровочных условий, например:  $\mathcal{M}_1$  — метод Бержа;  $\mathcal{M}_2$  — метод максимального согласования;  $\mathcal{M}_3$  — метод Жуковина и др. [10]. Ранжировку  $P_i(\mathcal{M}_i, G)$ , порождаемую методом  $i$  при обработке структуры  $G$ , можно рассматривать как мнение  $i$ -го эксперта. Набор ранжировок  $P_i, i = 1, 2, \dots$  можно рассматривать как групповой профиль — множество частных мнений экспертов по поводу решения одной задачи. Требуется найти такую интегральную ранжировку  $P$ , которая в некотором заданном смысле была наилучшим образом согласована с данным групповым профилем.

В теории принятия решения разработаны различные способы группового согласования: правило

Кумбса, функция Борда, метод Кемени—Снелла и др. [8,14]. Авторы планируют посвятить следующую статью применению методов группового принятия решений для решения задачи синтеза рационального упорядочения учебных курсов.

### Заключение

В работе обсуждается задача рационального упорядочения модулей электронного курса. Показано, что она обладает особенностями, которые не позволяют найти решение классическими методами построения рациональных ранжировок объектов, основанными на применении алгоритмов дискретного математического программирования.

Задача синтеза рационального упорядочения учебных модулей ставится как задача поиска наилучшей аппроксимации исходной структуры предпочтений, заданной в виде четкого или нечеткого графов, в классе всех линейных порядков на множестве вершин графа. Информация для формирования графа предпочтений может быть получена опросом эксперта (экспертов) или извлечена из гипертекстовой структуры ссылок.

Рассматриваются два типа экспертных данных: четкое и нечеткое бинарные отношения предпочтения на множестве учебных модулей. Кроме того, обсуждается исходная структура предпочтений, основанная на гипертекстовой разметке модулей учебного курса. Предлагается способ сведения всех типов исходных данных к нормализованной структуре, которая имеет простую калибровку. Предложенные преобразования вносят минимальные искажения в исходные структуры и сохраняют базовые предпочтения, заданные лицом, принимающим решение.

Структуры с простой калибровкой могут быть обработаны методом Бержа, который прошел глубокую апробацию на решении экономических и организационных задач и удовлетворяет большинству аксиом рационального упорядочения.

Научную новизну работы составляют постановка задачи упорядочения учебных модулей как задачи поиска наилучшей аппроксимации графовой структуры предпочтений в классе всех линейных порядков, способ преобразования структур  $G_1 - G_3$  к нормализованной структуре с простой калибровкой и новое применение классического метода Бержа для синтеза рациональной ранжировки множества электронных модулей.

*Работа выполнена при поддержке Минобрнауки России (проект 2014-14-579-0144).*

### Список литературы

1. Белоус В. В., Смирнова Е. В. Электронное обучение. Платформы и системы // Инженерный вестник: электронное научно-техническое издание. 2013. № 7. С. 501—512. URL: <http://engbul.bmstu.ru/doc/654234.html> (дата обращения 01.07.2015).

2. **Галямова Е. В.** Оценка качества электронного учебного материала // Международная научно-метод. конф. "Управление качеством инженерного образования и инновационные образовательные технологии" (Москва, 28—30 октября 2008 г.): докл. Ч. 2. М.: МГТУ им. Н. Э. Баумана, 2008. С. 29—34.
3. **Иванников А. Д.** Тематические интернет-порталы как средство агрегации электронного контента в заданной предметной области // Информационные технологии. 2014. № 3. С. 43—48.
4. **Домников А. С., Белоус В. В.** Интеллектуальный анализ данных в электронных обучающих системах // Инженерный вестник: электронное научно-техническое издание. 2013. № 12. С. 567—586. URL: <http://engbul.bmstu.ru/doc/656610.html> (дата обращения 01.07.2015).
5. **Домников А. С., Белоус В. В.** Рациональное упорядочение модулей учебного курса // Наука и образование. МГТУ им. Н. Э. Баумана. Электрон. журн. 2014. № 5. С. 192—205. DOI: 10.7463/0514.0710096
6. **Домников А. С., Белоус В. В.** Рациональное упорядочение модулей учебного курса на основе нечеткой исходной информации // Наука и образование. МГТУ им. Н. Э. Баумана. Электрон. журн. 2015. № 4. С. 215—227. DOI: 10.7463/0415.0765620

7. **Adam F., Humphreys P.** Encyclopedia of Decision Making and Decision Support Technologies // Information Science Reference, 2008. P. 114—152.
8. **Алескеров Ф. Т., Хабина Э. Л., Шварц Д. А.** Бинарные отношения, графы и коллективные решения. М.: Издательский дом ГУ ВШЭ, 2006. 301 с.
9. **Ларичев О. И.** Теория и методы принятия решений. М.: Логос, 2002. 392 с.
10. **Белкин А. Р., Левин М. Ш.** Принятие решений: Комбинаторные модели аппроксимации. М.: Наука, 1990. 160 с.
11. **Фаддеев Д. К., Фаддеева В. Н.** Вычислительные методы линейной алгебры. М.: Лань, 2002. 736 с.
12. **Юдин Д. Б.** Вычислительные методы теории принятия решений. М.: Наука, 1989. 320 с.
13. **Абаев Л. Ч.** Экспертное упорядочение альтернатив в задачах большой размерности // Управление большими системами: сб. тр. Вып. 40. М.: ИПУ РАН, 2012. С. 5—34.
14. **Баллод Б. А., Елизарова Н. Н.** Методы и алгоритмы принятия решений в экономике. М.: Финансы и статистика, 2009. 224 с.

**A. S. Domnikov**, Associate Professor, **V. V. Belous**, Associate Professor, [walentina.belous@bmstu.ru](mailto:walentina.belous@bmstu.ru)  
Moscow State Technical University n.a. N. E. Bauman, Moscow, Russia

## Synthesis of Rational Taught Modules Ordering by Berzha's Method

*The work considers a problem of creating a rationally ordered electronic course modules. Features of this task are investigated and the conclusion is drawn that it is impossible to solve it by classical methods of mathematical programming and mathematical statistics. The expert poll or data included in hypertext structure of links provides initial information for the synthesis of a rational order of the taught modules. The paper considers two types of expert information: clear and fuzzy binary relations of preference based on a set of the taught modules. It offers to reduce all three structures to the normalized view, i.e. a matrix of paired comparisons with simple calibration. The proposal is to solve a problem of ordering by Berzha's method, which has received wide practical approval in a diversity of applied fields.*

**Keywords:** *electronic course, ordering, theory of decision-making, binary*

### References

1. **Belous V. V., Smirnova E. V.** Ehlektronnoe obuchenie. Platformy i sistemy (-learning. Platforms and systems), *Inzhenernyj vestnik: ehlektronnoe nauchno-tehnicheskoe izdanie*, 2013, no. 7, pp. 501—512. URL: <http://engbul.bmstu.ru/doc/654234.html> (data obrashcheniya 01.07.2015) (in Russian).
2. **Galyamova E. V.** Ocenka kachestva ehlektronnoho uchebnogo materiala (The assessment of quality of e-learning material). *Mezh-dunarodnaya nauchno-metodicheskaya konferenciya "Upravlenie kachestvom inzhenernogo obrazovaniya i innovacionnye obrazovatel'nye tekhnologii"* (Moscow, 28—30 oktyabrya 2008 g.): dokl. CH. 2. М.: МГТУ им. Н.Э. Баумана, 2008, pp. 29—34 (in Russian).
3. **Ivannikov A. D.** Tematicheskie internet-portaly kak sredstvo agregacii ehlektronnogo kontenta v zadannoj predmetnoj oblasti (Thematic the Internet-portals as means of aggregation of an electronic content in the set subject domain), *Informacionnye tekhnologii*, 2014, no. 3, pp. 43—48 (in Russian).
4. **Domnikov A. S., Belous V. V.** Intellektual'nyj analiz dannyh v ehlektronnyh obuchayushchih sistemah (Intelligent data analysis in e-learning systems), *Inzhenernyj vestnik: ehlektronnoe nauchno-tehnicheskoe izdanie*, 2013, no. 12, pp. 567—586. Rezhim dostupa: <http://engbul.bmstu.ru/doc/656610.html> (data obrashcheniya 01.07.2015) (in Russian).
5. **Domnikov A. S., Belous V. V.** Racional'noe uporyadochenie modulej uchebnogo kursa (Rational orderliness of study course modules), *Nauka i obrazovanie*. МГТУ им. Н. Э. Баумана. Электрон. журн., 2014, no. 5, pp. 192—205. DOI: 10.7463/0514.0710096 (in Russian).
6. **Domnikov A. S., Belous V. V.** Racional'noe uporyadochenie modulej uchebnogo kursa na osnove nechetoj iskhodnoj infor-

- macii (Fuzzy Input Information-based Rational Ordering of Curriculum Modules), *Nauka i obrazovanie*, МГТУ им. Н. Э. Баумана. Электрон. журн., 2015, no. 4, pp. 215—227. DOI: 10.7463/0415.0765620 (in Russian).
7. **Adam F., Humphreys P.** Encyclopedia of Decision Making and Decision Support Technologies, *Information Science Reference*, 2008, pp. 114—152.
8. **Aleskerov F. T., Habina E. L., Shvarc D. A.** *Binarnye ot-nosheniya, grafy i kolektivnye resheniya* (Binary Relations, Graphs and Group Actions). Moscow: Izdatel'skiy dom GU VSHEH, 2006. 301 p. (in Russian).
9. **Larichev O. I.** *Teoriya i metody prinyatiya reshenij* (Theory and methods of decision-making). Moscow: Logos, 2002. 392 p. (in Russian).
10. **Belkin A. R., Levin M. Sh.** *Prinyatie reshenij: Kombinatornye modeli approksimacii* (Decision making: combinatorial models of approximation of information). Moscow: Nauka, 1990. 160 p. (in Russian).
11. **Faddeev D. K., Faddeeva V. N.** *Vychislitel'nye metody linejnoy algebry* (Computational methods of linear algebra). Moscow: Lan'apost, 2002. 736 p. (in Russian).
12. **Yudin D. B.** *Vychislitel'nye metody teorii prinyatiya reshenij* (Computational methods of decision-making theory). Moscow: Nauka, 1989. 320 p. (in Russian).
13. **Abaev L. Ch.** *Ehkspertnoe uporyadochenie al'ternativ v zadachah bol'shoj razmernosti* (Expert ranking of alternatives in the problems of big dimension), *Upravlenie bol'shimimi sistemami: sb. tr., Vyp. 40*. Moscow: IPU RAN, 2012, pp. 5—34. (in Russian)
14. **Balloд B. A., Elizarova N. N.** *Metody i algoritmy prinyatiya reshenij v ehkonomike* (Methods and algorithms for decision making in economy). Moscow: Finansy i statistika, 2009, 224 p. (in Russian).

**И. Г. Малыгин**, д-р техн. наук, проф., директор,  
**В. И. Комашинский**, д-р техн. наук, зам. директора по научной работе, kama54@rambler.ru,  
 ИПТ РАН

## Информационные технологии и искусственный интеллект — основные двигатели четвертой индустриальной революции (Industrie 4.0)

*Рассмотрены основные черты новой индустриальной революции в сфере транспорта. Показано, что ключевой технологической платформой для новой индустриальной революции (независимо от сфер ее приложения) являются информационно-сетевые технологии, интегрированные (точнее, конвергированные) с технологиями индустриального искусственного интеллекта. Отмечается также, что важным условием успешного проведения новой индустриальной революции является подготовка профессионалов, способных разрабатывать и создавать интеллектуальные промышленные инфраструктуры и поддерживать их эффективное функционирование и последующее развитие.*

**Ключевые слова:** четвертая индустриальная революция, индустриальные информационно-управляющие сети, индустриальный искусственный интеллект, когнитивные индустриальные системы, интеллектуальный транспорт

### Индустриальные революции и их отличительные черты

Анализируя траекторию развития технологий построения транспортных средств и транспортных магистралей, трудно не заметить положительную ее корреляцию с траекторией развития информационно-сетевых и телекоммуникационных технологий (в широком их смысле). Это объясняется, прежде всего тем, что технологии построения транспортных систем являются информационно емкими (для их реализации требуется некоторый, критичный, объем информации и знаний). Например, для перехода от гужевого транспорта к паровозам и железным дорогам потребовалось несколько тысяч лет формирования, технологий сбора, накопления и применения знаний в самых разных областях, прежде чем появилась возможность изобрести и внедрить технологии построения транспортных систем на паровой тяге.

Переход от паровых двигателей к двигателям внутреннего сгорания и современным транспортным системам произошел гораздо быстрее в связи с ускорением прогресса в области информационных, телекоммуникационных и индустриальных технологий и формированием новых, электронных методов получения, хранения, переноса и применения данных информации и знаний.

Особенностью наступающей постинформационной эпохи является усиление акцента на технологиях получения новых знаний и разработке способов их применения посредством специальных технических систем, получивших название искусственных

когнитивных технических систем [1, 2]. Применительно к транспортной сфере эти технологии уже используются в рамках создания автономных (роботизированных) транспортных средств и интеллектуальных магистралей.

Совершенствование информационных и сетевых технологий (рис. 1) провоцирует так называемые индустриальные революции (рис. 2), которые всегда были и остаются ключевыми движущими силами развития национальных и глобальной экономики на протяжении последних двух столетий. Опыт прошедших лет свидетельствует о том, что смена индустриальной парадигмы не происходит в одночасье, она протекает последовательно шаг за шагом [3–5].

В этом контексте очередная (рис. 2), четвертая индустриальная революция (Industrie 4.0) требует



Рис. 1. Ковэволюция информационных, индустриальных и транспортных технологий

	<b>1784 г. Первая индустриальная революция.</b> Проявилась в появлении механических ткацких станков, широком использовании ветряных, водяных и паровых двигателей
	<b>1870 г. Вторая индустриальная революция.</b> Проявилась в появлении конвейерного массового производства, основанного на разделении труда и широком использовании электрических, бензиновых и дизельных двигателей
	<b>1969 г. Третья индустриальная революция.</b> Проявилась в появлении программируемых логических контроллеров в широком промышленном использовании электроники и ИТ, в масштабной автоматизации производства
	<b>2015 г. Четвертая индустриальная революция.</b> Проявляется в конвергенции промышленного искусственного интеллекта, производственных информационных сетей и индустриального оборудования в интересах создания самоорганизующегося, чрезвычайно гибкого производства. Производственные линии заменяются на промышленные матричные сети

Рис. 2. Основные этапы развития индустриальных технологий

пристального внимания, эффективного управления и кропотливой совместной работы ученых, инженеров, экономистов и политиков. Анализ особенностей предшествующих индустриальных революций, их движущих сил и вызванных ими экономических и социальных последствий позволяет более осмысленно подойти к организации и проведению очередной — неоиндустриальной революции.

**Первая индустриальная революция** (см. рис. 2) началась с появления паровых двигателей, введения механического производственного оборудования и создания транспорта на паровой тяге (пароходов, паровозов и железных дорог).

**У второй индустриальной революции отличительной особенностью** стало широкое использование электричества и двигателей внутреннего сгорания с электрическим зажиганием, что привело к появлению электрического промышленного оборудования, конвейерной ленты, электротранспорта и широкого перечня электрифицированных бытовых изделий, а также нового поколения транспортных средств (автомобилей, мотоциклов, теплоходов, самолетов и т.д.).

**Третья индустриальная революция** характеризуется автоматизацией производственных процессов на основе широкого использования электроники, информационных и коммуникационных технологий, а также выпуском более совершенных компьютеризированных бытовых и промышленных изделий, автоматизированных транспортных систем — наземных, воздушных, морских, космических (в том числе и автопилотных).

### Промышленные информационно-телекоммуникационные системы и их роль в четвертой индустриальной революции

Основой наступающей четвертой индустриальной революции станет широкое использование эле-

ментов промышленного искусственного интеллекта и когнитивных информационно-управляющих систем. **Когнитивные индустриальные информационно-управляющие системы** призваны обеспечить сетевую интеграцию встроенных информационных структур и элементов искусственного интеллекта в объекты, материалы и машины, а также системы логистики, координации и управления процессами и их совместное сетевое взаимодействие [1, 2].

Для более отчетливого рассмотрения изменений, ожидаемых при переходе к Industrie 4.0, целесообразно рассмотреть инфраструктурные особенности (рис. 3), характерные для текущей индустрии (в соответствии с используемой терминологией — Industrie 3.0).

Нужно отметить что Industrie 3.0 отличается от Industrie 2.0 широкой автоматизацией производственных процессов на основе применения программ-

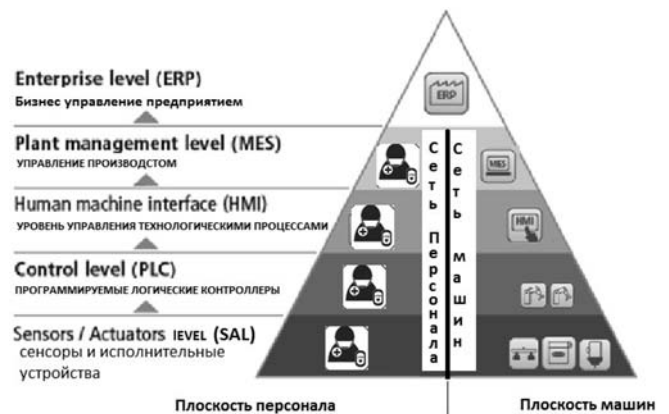


Рис. 3. Иерархическая архитектура существующей системы автоматизированного управления промышленным предприятием

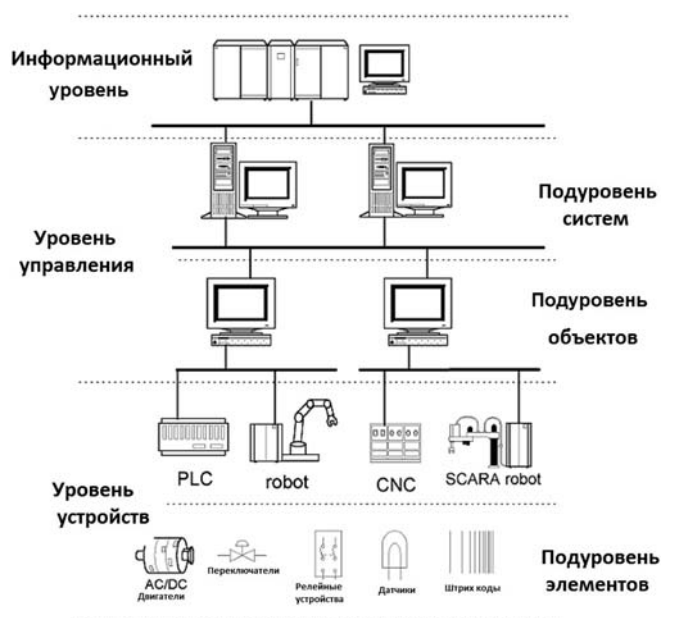


Рис. 4. Обобщенная архитектура современных индустриальных информационно-управляющих сетей

руемых логических контроллеров (PLS) и промышленных информационно-управляющих сетей (рис. 4).

**На информационном уровне** (рис. 4) осуществляется формирование, хранение и распределение информационных моделей изделий (например, судов и локомотивов, автомобилей, самолетов, и т.д.) и алгоритмов их преобразования в промышленные изделия, материализуемые на уровне устройств.

**На уровне управления** осуществляется распределение и контроль выполнения задач (детализированных информационных моделей) между логически взаимосвязанными группами производственных объектов (через сети подуровня систем), распределение задач и управление различными исполнительными устройствами (роботами, станками, инструментами и т.д.) посредством сетей подуровня объектов.

**На уровне устройств** обеспечивается выполнение физических действий (в соответствии с информационными моделями) на основе преобразования предоставленных ресурсов (информационных материальных и энергетических) в конечную продукцию.

Как следует из рис. 3 и рис. 4, в промышленных информационно-управляющих сетях можно выделить две логические плоскости: **плоскость персонала** и **плоскость машин**.

Плоскость персонала является доминирующей в процессах разработки (информационных моделей продукции и алгоритмов их преобразования в промышленные изделия), построения (формирования промышленной инфраструктуры) и управления (процессом производства изделий).

Иными словами, в ходе третьей промышленной революции — Industrie 3.0 — была алгоритмизирована и автоматизирована (передана техническим устройствам) значительная часть физически тяжелых работ людей, задействованных в промышленном производстве, при этом интеллект людей оставался незаменим на всех этапах жизненного цикла промышленных предприятий (и создаваемых ими изделий). Необходимо еще раз подчеркнуть, что базовой платформой, на которой была проведена третья промышленная революция, явились новые (на то время) информационно-сетевые технологии.

Существующая в настоящее время (см. рис. 3 и рис. 4) жесткая иерархическая архитектура системы автоматизированного управления промышленным предприятием (в рассматриваемом случае — занимающегося выпуском автомобилей) определяет жесткую последовательную процедуру выполнения работ (рис. 5). В таких условиях [6—10] переход на выпуск новой или модифицированной продукции достаточно сложен, поскольку требует введения корректировок на всех уровнях архитектуры (см. рис. 3 и рис. 4). При этом индивидуальные требования клиента не всегда могут быть удовлетворены, например, практически нет возможностей по

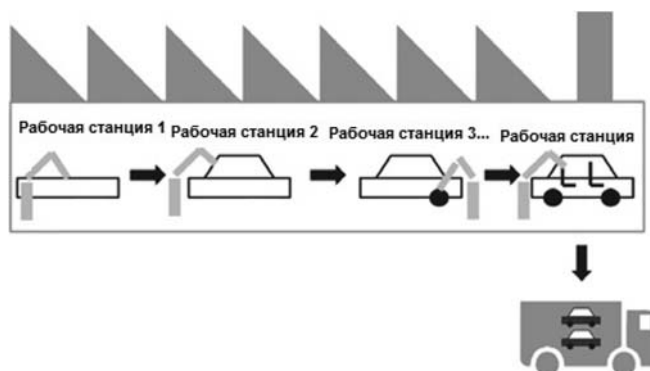


Рис. 5. Выполнение работ на существующей производственной линии

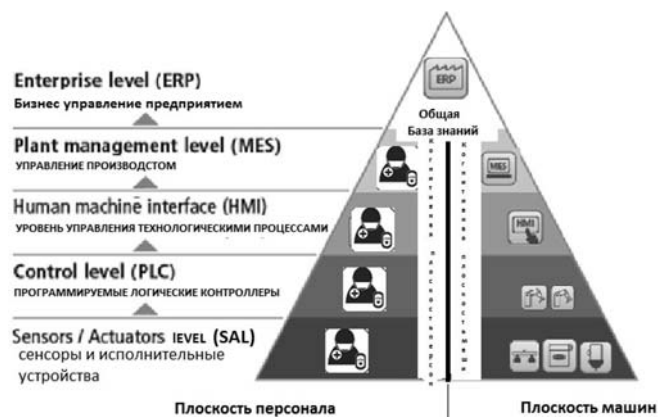


Рис. 6. Иерархическая архитектура перспективной системы интеллектуального управления промышленным предприятием

установке компонентов из другой группы продукции, изготавливаемых в этой же компании.

Отличительной особенностью новой промышленной эпохи — Industrie 4.0 — является постепенный перенос (передача) части интеллектуальных функций (ранее поддерживаемых только людьми) специальным техническим (когнитивным) промышленным системам. Поэтому в процессе реализации концепции Industrie 4.0 потребуется, прежде всего, создание плоскости специализированной когнитивной инфокоммуникационной сети персонала (левая часть рис. 6), которая предназначена для предоставления ему точных контекстных декларативных и процедурных промышленных знаний в реальном масштабе времени. И только после этого у персонала появляются возможности формирования плоскости промышленной когнитивной информационно-управляющей сети (правая часть рис. 6) и загрузки (передачи) промышленных знаний (первичных) для промышленного оборудования.

Обобщенная архитектура перспективной когнитивной промышленной информационно-управляющей сети представлена на рис. 7.

Таким образом, система управления интеллектуальным промышленным предприятием включает когнитивную плоскость людей (в составе интел-

лектуальных прикладных процессов и баз знаний для обслуживающего персонала) и когнитивную плоскость машин (в составе информационных приложений, промышленных операционных систем и промышленного искусственного интеллекта).

Логическая архитектура (см. рис. 6) открывает широкие возможности не только для существенного повышения эффективности производственного процесса, но и для функциональных и качественных характеристик выпускаемой продукции (насчет ее интеллектуализации).

В частности, автоматизированные производственные линии (см. рис. 5), используемые в настоя-

щее время в рамках концепции Industrie 3.0, трансформируются в ходе реализации концепции Industrie 4.0 в интеллектуальные производственные сети (рис. 8), при этом каждое изготавливаемое изделие (в нашем случае — автомобиль) может перемещаться (коммутироваться) в пределах завода. Такая динамически реконфигурируемая производственная сеть [11—13] позволяет смешивать и сочетать оборудование, которым оснащается каждый автомобиль (рис. 8), в соответствии с конкретной модификацией и комплектацией модели или на основе индивидуальной комплектации, выбранной по предзаказу в автосалоне конкретным покупателем (что характерно для немецких автоконцернов). Кроме того, индивидуальные вариации в изделии могут быть реализованы на любом этапе производства в ответ на обнаруженные технические проблемы или для внедрения очередных инновационных решений.

Как было отмечено ранее, отличительной особенностью каждой индустриальной революции являлось не только повышение производительности и качества труда, но и совершенствование производимой продукции и сопутствующей ей инфраструктуры. Например, изобретение паровых двигателей (в период первой индустриальной революции) привело к появлению не только индустриальных паровых молотов и станков, но и паровозов и железных дорог. Изобретение электрических двигателей и двигателей внутреннего сгорания (в период второй индустриальной революции) привело не только к радикальным изменениям в производственной сфере, но и к появлению автомобильного и электрического транспорта, а также сопутствующих им автомобильных и электрифицированных транспортных магистралей.

### Некоторые последствия четвертой индустриальной революции

Очередная индустриальная революция также приведет к существенным изменениям в производимой ею продукции и сопутствующей ей инфраструктуре [7—12]. Интеллектуализация транспортной индустрии приведет к появлению интеллектуального транспорта и потребует интеллектуализации транспортных инфраструктур (автомобильных, железнодорожных, морских, авиационных и др.)

Отличительной особенностью умной промышленной продукции является то, что, помимо аппаратных и программных компонентов, в ее состав входят датчики, исполнительные устройства, базы данных (информации и знаний), микропроцессоры, элементы искусственного интеллекта, а также сетевые инфраструктуры, обеспечивающие взаимодействие всех составных частей друг с другом. Новая промышленная продукция (по сравнению с традиционной) будет отличаться более высокой функциональностью и сложностью. Например, интел-

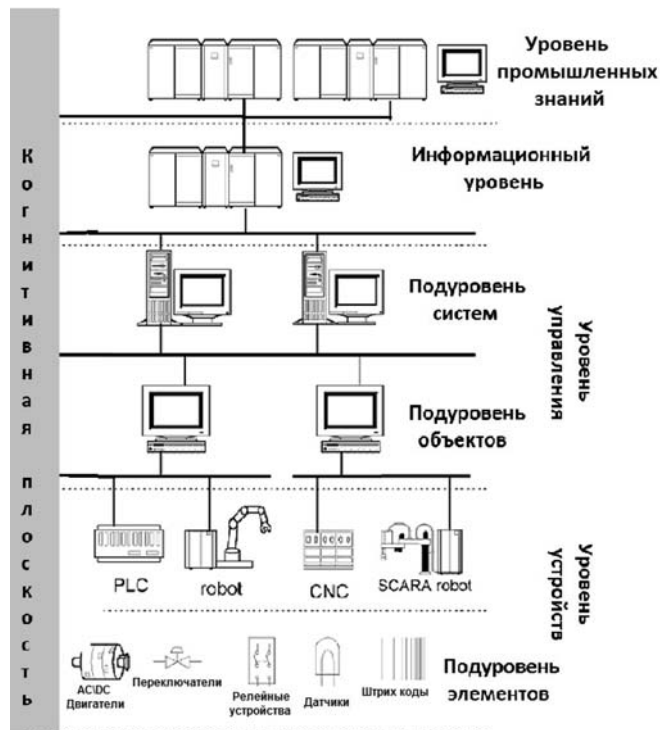


Рис. 7. Обобщенная архитектура перспективной когнитивной индустриальной информационно-управляющей сети

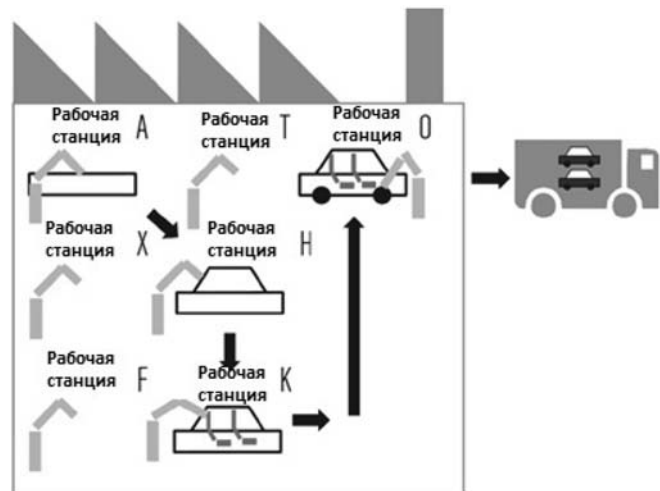


Рис. 8. Выполнение работ в перспективной интеллектуальной производственной сети



лектуальные автомобили (рис. 9) будут оснащены датчиками, позволяющими им автономно оценивать внешнее окружающее пространство и свое внутреннее состояние.

Входящие в состав интеллектуального автомобиля процессоры, программное обеспечение и элементы искусственного интеллекта делают автомобиль "умным" (обеспечивают обработку информации, получение знаний и планирование действий), дают ему возможность автономно принимать решения, самообучаться и разумно выполнять действия.

Сетевые инфраструктуры интеллектуального автомобиля призваны обеспечивать взаимодействие его внутренних элементов, а также взаимодействие с "умной" внешней окружающей средой (другими интеллектуальными автомобилями и "умной" автодорогой) в интересах общего эффективного и безаварийного функционирования. Интеллектуальные автомобили используют различные исполнительные устройства (приводы) для изменения собственного состояния и различные системы сигналов (акустических, оптических, электромагнитных), для того чтобы оказывать влияние на свое внешнее окружение. Интеллектуальные автомобили способны автономно и/или на основе внешних команд реагировать и адаптироваться к изменениям окружающей среды.

Очевидно, что возможности интеллектуальных автомобилей будут эффективно применяться и могут быть существенно расширены за счет интеллектуализации автомобильных дорог (рис. 10, см. четвертую сторону обложки), дополненных датчиками, интеллектуальным дорожным оборудованием ("умными" светофорами, адаптивными дорожными знаками и указателями), другими системами, поддерживающими функциональное и информационное взаимодействие с автомобилями, водителями и пешеходами.

Важно также отметить такую особенность новой индустриальной революции [13], как непрерывное сетевое взаимодействие всех перечисленных элементов, в результате которого становится реализуемо эффективное управление всеми звеньями жизненного цикла автотранспорта (рис. 11, см. четвертую сторону обложки).

Применительно к автотранспортной индустрии логическая архитектура Industrie 4.0 призвана управлять всеми этапами жизненного цикла автомобиля, начиная от получения персонализированного заказа от покупателя и заканчивая утилизацией (рис. 12, см. четвертую сторону обложки).

Длительное сетевое взаимодействие всех интеллектуальных сетевых элементов (интеллектуальных промышленных предприятий, интеллектуального

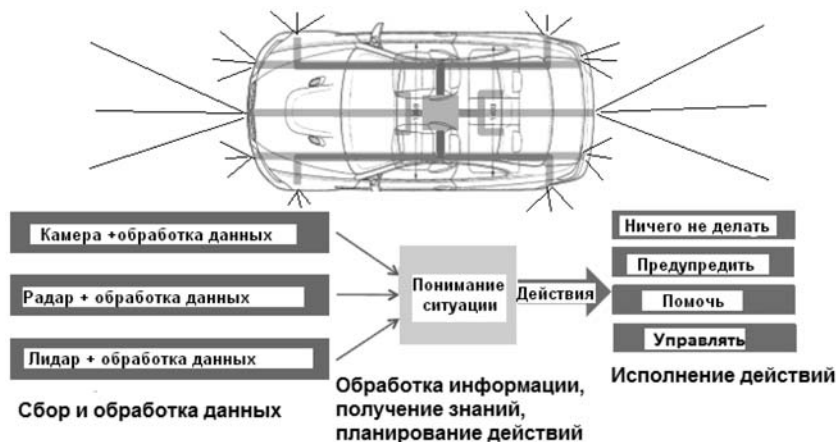


Рис. 9. Обобщенная архитектура интеллектуального автомобиля

транспорта, "умных" дорог и интеллектуального автосервиса) в конечном итоге приводит к увеличению производительности и стоимости каждого из них. Таким образом, формируется "самовоспроизводящийся цикл повышения стоимости".

Сетевые элементы (промышленные, встраиваемые в продукцию и дорожную инфраструктуру) выполняют две основные функции:

во-первых, они дают возможность обмена данными между продуктом и его операционной средой, производителем, системами технического обслуживания и пользователями;

во-вторых, некоторые функции физического продукта (в нашем случае автомобиля) могут быть переданы на внешние системы (например, в случае выхода из строя системы управления интеллектуального автомобиля функции управления им может взять на себя "интеллект" дороги).

**Выводы.** В данной работе рассмотрены особенности проявления новой индустриальной революции в сфере транспорта. В других сферах промышленного производства (авиастроения, ракетостроения, судостроения и т.д.) будут в целом сохраняться отмеченные проявления (естественно, будут иметь место и характерные для каждой сферы особенности). Вместе с тем важно еще раз подчеркнуть, что ключевой технологической платформой для новой индустриальной революции (независимо от сфер ее приложения) продолжают оставаться промышленные информационно-сетевые технологии (как и на протяжении третьей индустриальной революции), интегрированные (точнее, конвергированные) с технологиями индустриального искусственного интеллекта. Еще более важным условием успешного проведения новой индустриальной революции является подготовка специалистов-профессионалов, способных разрабатывать и создавать интеллектуальные индустриальные инфраструктуры и поддерживать их эффективное функционирование и дальнейшее развитие.

## Список литературы

1. **Комашинский В. И.** Когнитивные системы и телекоммуникационные сети // Вестник связи. 2011. № 10. С. 4—8.
2. **Малыгин И. Г., Комашинский В. И., Афонин П. Н.** Системный подход к построению когнитивных транспортных систем и сетей // Проблемы управления рисками в техносфере. 2015. № 4. С. 68—73.
3. **Комашинский В. И., Комашинский Д. В.** Когнитивная метафора в развитии телекоммуникационных и промышленных сетевых инфраструктур, или первые шаги к постинформационной эпохе // Технологии и средства связи. 2015. № 1. С. 62—67.
4. **Малыгин И. Г., Комашинский В. И.** Некоторые проблемы построения когнитивных транспортных систем и сетей // Труды международной конференции "Транспорт России: проблемы и перспективы — 2015 год". Том. 1. С. 3—8.
5. **Bauer J., Schlund P., Marrenbac D., Ganscha O.** Industrie 4.0 — Volkswirtschaftliches Potenzial für Deutschland, Berlin, 2014. P. 5—30.
6. **Bauernhansl T., Hompel M., Vogel-Heuser B.** Industrie 4.0 in Produktion, Automatisierung und Logistik // Anwendung, Technologie, Migration, Wiesbaden, 2014. P. 12—17.
7. **Kagermann H., Riemensperger F., Hoke D., Helbig J., Stocksmeier D., Wahlster W., Scheer A. W., Schweer D.** Smart Service Welt — Umsetzungsempfehlungen für das Zukunftsprojekt Internetbasierte Dienste für die Wirtschaft, Berlin, 2014. P. 14—34.
8. **Kagermann H., Wahlster W., Helbig J.** Umsetzungsempfehlungen für das Zukunftsprojekt Industrie 4.0 — Abschlussbericht des Arbeitskreises Industrie 4.0. Frankfurt/Main, 2013. P. 5—105.
9. **Kersten W., Schröder M., Indorf M.,** Industrie 4.0 — Auswirkungen auf das Supply Chain Risikomanagement // Kersten W., Koller H., Lödding H. (Hrsg.), Industrie 4.0 — Wie intelligente Vernetzung und kognitive Systeme unsere Arbeit verändern, Berlin, 2014. P. 101—126.
10. **Mertens P.** Industrie 4.0 — Herausforderungen auch an Rechnungswesen und Controlling im Überblick, in: Controlling — Zeitschrift für erfolgsorientierte Unternehmensteuerung, 27. Jg., 2015. H. 8/9. P. 27—29.
11. **Porter M. E., Heppelmann J. E.,** Wie smarte Produkte den Wettbewerb verändern // Harvard Business Manager. 2014. N 12. P. 34—61.
12. **Berger R.** (Hrsg.). INDUSTRIE 4.0 -The new industrial revolution // How Europe will succeed. München, 2014. P. 1—16.
13. **Industrie 4.0 Controlling in the Age of Intelligent Networks.** URL: [https://www.icv.controlling.com/fileadmin/Assets/Content/AK/Ideenwerkstatt/Files/Dream\\_Car\\_Industrie\\_4.0\\_EN.pdf](https://www.icv.controlling.com/fileadmin/Assets/Content/AK/Ideenwerkstatt/Files/Dream_Car_Industrie_4.0_EN.pdf)

I. G. Malygin, Professor, Director, V. I. Komashinskiy, Professor, Deputy Director, kama54@rambler.ru  
IPT RAS

## Information Technology and Artificial Intelligence — the Main Engines of the Fourth Industrial Revolution (Industrie 4.0)

*In paper the fundamental features of new industrial revolution in motor transport sphere are observed. It is shown, that a key technological platform for new industrial revolution (it is not dependent on spheres of its application) are когнитивные информационно-сетевые производственные инженерные интегрированные (более точно, сходящиеся) с производственной инженерией промышленного искусственного интеллекта. It is marked also, that the important condition of successful conducting of new industrial revolution is preparation of professionals, capable to develop and establish intellectual industrial infrastructures and to sustain their effective operation and the subsequent development.*

**Keywords:** fourth industrial revolution, industrial information and control networks, industrial artificial intelligence, cognitive industrial systems, intelligent transport

### References

1. **Komashinskiy V. I.** Kognitivnye sistemy i telekommunikatsionnye seti, *Vestnik svyazi*, 2011, no. 10, pp. 4—8.
2. **Malygin I. G., Komashinskiy V. I., Afonin P. N.** Sistemnyy podhod k postroeniyu kognitivnykh transportnykh sistem i setej, *Problemy upravleniya riskami v tekhnosfere*, 2015, no. 4, pp. 68—73.
3. **Komashinskiy V. I., Komashinskiy D. V.** Kognitivnaya metafora v razvitii telekommunikatsionnykh i industrial'nykh setevykh infrastruktur, ili pervye shagi k postinformatsionnoy ehpohe, *Tekhnologii i sredstva svyazi*, 2015, no. 1, pp. 62—67.
4. **Malygin I. G., Komashinskiy V. I.** Nekotorye problemy postroeniya kognitivnykh transportnykh sistem i setej, *Trudy mezhdunarodnoy konferencii "Transport Rossii: problemy i perspektivy — 2015 god"*, vol. 1, pp. 3—8.5.
5. **Bauer J., Schlund P., Marrenbac D., Ganscha O.** *Industrie 4.0 — Volkswirtschaftliches Potenzial für Deutschland*, Berlin, 2014, pp. 5—30.
6. **Bauernhansl T., Hompel M., Vogel-Heuser B.,** *Industrie 4.0 in Produktion, Automatisierung und Logistik — Anwendung, Technologie, Migration*, Wiesbaden, 2014, pp. 12—17.
7. **Kagermann H., Riemensperger F., Hoke D., Helbig J., Stocksmeier D., Wahlster W., Scheer A. W., Schweer D.** *Smart Service Welt — Umsetzungsempfehlungen für das Zukunftsprojekt Internetbasierte Dienste für die Wirtschaft*, Berlin, 2014, pp. 14—34.
8. **Kagermann H., Wahlster W., Helbig J.** Umsetzungsempfehlungen für das Zukunftsprojekt Industrie 4.0, *Abschlussbericht des Arbeitskreises Industrie 4.0*, Frankfurt/Main, 2013. P. 5—105.
9. **Kersten W., Schröder M., Indorf M.,** *Industrie 4.0 — Auswirkungen auf das Supply Chain Risikomanagement*, in: Kersten W., Koller H., Lödding H. (Hrsg.), *Industrie 4.0 — Wie intelligente Vernetzung und kognitive Systeme unsere Arbeit verändern*, Berlin, 2014, P. 101—126.
10. **Mertens P.** Industrie 4.0 — Herausforderungen auch an Rechnungswesen und Controlling im Überblick, in: *Controlling — Zeitschrift für erfolgsorientierte Unternehmensteuerung*, 27. Jg. 2015, H. 8/9, pp. 27—29.
11. **Porter M. E., Heppelmann J. E.,** Wie smarte Produkte den Wettbewerb verändern, *Harvard Business Manager*, 2014, no. 12, pp. 34—61.
12. **Berger R.** (Hrsg.). *INDUSTRIE 4.0 — The new industrial revolution: How Europe will succeed*, München 2014. P. 1—16.
13. **Industrie 4.0. Controlling in the Age of Intelligent Networks** [https://www.icv.controlling.com/fileadmin/Assets/Content/AK/Ideenwerkstatt/Files/Dream\\_Car\\_Industrie\\_4.0\\_EN.pdf](https://www.icv.controlling.com/fileadmin/Assets/Content/AK/Ideenwerkstatt/Files/Dream_Car_Industrie_4.0_EN.pdf)



# МОДЕЛИРОВАНИЕ И ОПТИМИЗАЦИЯ MODELING AND OPTIMIZATION

УДК 621.675

**И. Я. Львович**<sup>1</sup>, д-р техн. наук, проф., office@vivt.ru,  
**Я. Е. Львович**<sup>2</sup>, д-р техн. наук, проф., office@vivt.ru,  
**А. П. Преображенский**<sup>2</sup>, д-р техн. наук, проф., app@vivt.ru,  
**О. Н. Чопоров**<sup>2</sup>, д-р техн. наук, проф., choporov\_oleg@mail.ru  
<sup>1</sup>Панъевропейский университет, Братислава  
<sup>2</sup>Воронежский институт высоких технологий, Воронеж

## Особенности методов вычислительной гидродинамики для моделирования турбулентности

*Проводится аналитический обзор методов, применяемых для моделирования турбулентности на персональных компьютерах. Указаны вычислительные особенности рассматриваемых методов, которые могут использоваться в математическом обеспечении подсистем моделирования турбулентности в соответствующих САПР. Результаты исследований показывают, что вычислительная гидродинамика может занять доминирующее место, превзойдя по пространности экспериментальные и аналитические методы.*

**Ключевые слова:** турбулентность, моделирование, математическое обеспечение САПР, вычислительная физика, уравнение, вихрь, интеграл, инженерное приложение, жидкость, метод

### Введение

В настоящее время продолжает развиваться интеграция САПР (computer-aided design, CAD) с системами моделирования (Computer-aided engineering, CAE), позволяющими проводить тестирование создаваемого объекта непосредственно во время проектирования. Кроме того, средства, которые дают возможность ставить вычислительные эксперименты, достаточно давно используются в науке, когда натурные эксперименты слишком дороги, невозможны или их необходимо повторить большое число раз в одинаковых условиях.

Несмотря на то что большинство технических процессов описывается известными физическими законами, часто практически невозможно заранее представить, как поведет себя некий объект в тех или иных условиях. Данное обстоятельство обусловлено лишь чрезвычайной сложностью современных систем. Вычислительная физика представляет собой третий способ проведения научных и технических исследований в дополнение к эксперименту и теории. При этом компьютер играет роль прибора, который дает новые возможности для изучения свойств различных физических моделей. Развитие компьютерной техники и численных методов вычислительной физики позволяет не только работать с все более сложными моделями, но и применять различные способы визуализации полученного результата.

Целью работы является проведение обзора методов, используемых для моделирования турбулентности на персональных компьютерах. При разработке САПР подсистема моделирования турбулентности может динамически подключаться в виде отдельного модуля.

Большинство физических явлений описываются дифференциальными уравнениями в частных производных (Partial Differential Equations, PDE). Как известно, аналитические решения для PDE могут быть получены лишь в простейших модельных случаях, которые на данный момент представляют больше научный интерес, чем технический. В практических задачах применяются численные и приближенные методы решения. Вычислительные эксперименты можно условно разделить на три категории. К первой относятся эксперименты, предназначенные для моделирования сложных устройств, с тем чтобы оценить большое число вариантов, прежде чем создавать оптимальную модель с применением дорогостоящей технологии. Вторая группа вычислительных экспериментов направлена на получение информации в тех ситуациях, когда имеется большой разрыв между возможностями теории и эксперимента. Лабораторные эксперименты сталкиваются со всей сложностью природы: условия могут с трудом поддаваться контролю, измерения не всегда можно легко осуществить, и,

как следствие, результаты зачастую трудно интерпретировать однозначно.

К третьей группе относятся теоретические эксперименты, которые дают ценные рекомендации для развития теории.

### Вычислительная гидродинамика

Одним из самых крупных разделов вычислительной физики является вычислительная гидродинамика (Computational Fluid Dynamics, CFD). В нее входят методы численного моделирования для описания движения жидких сред, что позволяет существенно ускорить разработку различных технических устройств. Кроме технических приложений вычислительная гидродинамика широко применяется и в других областях, таких как моделирование атмосферных условий, динамики огня, биохимических процессов и т.д.

Развитие методов численного моделирования течений началось в 50-х годах прошлого столетия с появлением вычислительной гидроаэродинамики. С середины 70-х до 80-х годов из-за недостатка компьютерных мощностей вычислительная гидродинамика в большей степени использовалась в научных целях для моделирования относительно простых и элементарных случаев течения. Именно в это время закладывались ее математический и физический фундаменты.

С наступлением 90-х годов и по настоящее время развитие вычислительной техники позволило широко применять методы численного моделирования. Вычислительная гидродинамика существенно расширила свой математический и физический аппарат, усовершенствовались модели турбулентности, горения, излучения, акустики, двухфазных течений, методы дискретизации и численного решения дифференциальных уравнений и т.д.

С развитием компьютерной техники и совершенствованием методов вычисления и анализа данных вычислительная гидродинамика может занять доминирующее место, превзойдя по распространенности экспериментальные и аналитические методы. К достоинствам численного моделирования течений в инженерной практике можно отнести: сокращение времени разработки технических устройств, моделирование процессов и течений, для которых недоступно проведение эксперимента, детальная картина процессов, снижение стоимости разработки. Вместе с тем, вычислительная гидродинамика имеет такие недостатки, как конечная точность расчета, которая в некоторых случаях способна перечеркнуть преимущества численного моделирования.

Особый интерес представляет моделирование турбулентных течений, в первую очередь из-за того, что такие течения наиболее распространены. Кроме того, до сих пор не существует полной теории, которая бы описывала подобные течения. Исследование экспериментальными методами также за-

труднено. Поэтому моделирование турбулентных течений занимает главенствующую роль в практических расчетах и теоретических исследованиях.

### Турбулентность

Течения жидкостей и газов, наблюдаемых в природных условиях и технических устройствах, делятся на два принципиально различающихся типа: ламинарные (спокойные, плавные, регулярные) и турбулентные (хаотические, неупорядоченные во времени и в пространстве).

Ламинарные течения являются скорее исключением, чем правилом. Их отличают малые размеры и большая вязкость среды. Типичными примерами ламинарных течений служат обтекание сферы низкоскоростным потоком и течение смазочного масла в подшипнике.

Большинство движений жидкостей, газов и плазмы, встречающихся в технике, являются турбулентными. Течение воды в реках и каналах, течение нефти и природного газа в трубопроводах, течения в проточных трубах турбомашин и газовых трактах ракетных двигателей являются турбулентными. Турбулентность оказывает влияние на процессы горения, смешения и переноса частиц дисперсионной примеси. Среди течений с геофизическими масштабами следует отметить течение воздуха в атмосферном пограничном слое, струйные течения в верхней атмосфере, течения воды ниже поверхности океана. Облака в атмосфере, облака межзвездного газа, газовые туманности также находятся в состоянии турбулентного движения.

Состояние течения можно охарактеризовать некоторым числом, называемым числом Рейнольдса, определяемым по следующей формуле:

$$Re = \frac{l_0 u_0}{\nu_0}, \quad (1)$$

где  $l_0$  — характерная длина;  $u_0$  — характерная скорость;  $\nu_0$  — коэффициент кинематической вязкости среды. Число Рейнольдса характеризует соотношение между инерционными и вязкостными силами. Если это соотношение мало, т.е. инерциальные силы сравнимы с вязкостными, течение имеет упорядоченную структуру, представляя собой ламинарное течение.

Если соотношение велико, т.е. инерциальные силы преобладают над вязкостными, течение является нестабильным и возмущения разрушают упорядоченную структуру, приводя к турбулентной структуре течения. Критическое значение числа Рейнольдса зависит от конкретного вида течения. Для потока воды в трубе числа Рейнольдса выше 4000, скорее всего, соответствуют турбулентным потокам, в то время как число Рейнольдса ниже 2100 указывает на его ламинарность. Область  $2100 < Re < 4000$  называется *переходной зоной*. Процесс образования турбулентности в переходной

зоне описывается фрактальной геометрией и теорией перколяции [1]. В частности, теория перколяции на основе метода статистических испытаний моделирует распространение турбулентного режима по объему жидкости и позволяет в среднем оценить время перехода в турбулентный режим, а также структуру течения в больших масштабах. Процесс возникновения турбулентности в течении, изначально являющемся безвихревым, обусловлен вязкостью.

Из теоремы Кельвина следует, что нулевая завихренность сохраняется в процессе движения идеальной жидкости. Присутствие твердых границ порождает завихренность. Завихренность увеличивается вследствие механизма растяжения вихревых нитей, и в этой области течения, как правило, становится турбулентным.

С теоретической точки зрения турбулентные течения представляют собой пример нелинейной открытой системы с большим числом степеней свободы. В реальных турбулентных потоках спектр изменения волновых чисел (диапазон изменения масштабов турбулентного движения) достигает нескольких порядков.

Турбулентное течение можно представить как множество вращающихся в разные стороны и в разных плоскостях вихрей. Большие вихри переходят в маленькие вихри, размеры которых немного превышают длину молекулярного пробега, рассеивают энергию и теплоту посредством молекулярной вязкости. Этот процесс был математически описан Колмогоровым и назван *энергетическим каскадом*. Возможен и обратный процесс образования крупных вихрей из малых вихрей, получивший название *обратный распад*. Определяя длину волны  $\lambda$  обратно пропорционально размеру вихря  $l$  как

$$\lambda = \frac{2\pi}{l}, \quad (2)$$

можно представить спектр кинетической энергии как функцию от длины волны (см. рисунок).



Спектр турбулентной кинетической энергии  $E_w$  в зависимости от волнового числа  $k_w$

Спектр турбулентности кинетической энергии имеет три характерные зоны:

- область малых длин волн, содержащих крупные вихри, несущие большую часть кинетической энергии;
- инерционная зона, где под воздействием инерционных сил крупные вихри распадаются на более мелкие;
- область больших волн, образующих диссипативную (вязкостную) зону, где мельчайшие вихри диссипируют в теплоту под действием сил трения.

В итоге турбулентное течение характеризуется такими свойствами: случайные параметры относительно времени и пространства; нестационарные параметры; трехмерность; диссипативные параметры; вихревые особенности.

Поэтому описать турбулентности можно с помощью вероятностных законов, используя функцию плотности вероятности. Однако в рамках данной работы рассматриваются только осредненные параметры (скорость, давление, плотность и т.д.) течения.

### Проблема описания турбулентности

Уравнения Навье—Стокса, описывающие движение множества турбулентных течений вязкой ньютоновской жидкости с достаточно большой точностью, имеют вид:

$$\begin{aligned} \frac{\partial V_1}{\partial t} + V_1 \frac{\partial V_1}{\partial x_1} + V_2 \frac{\partial V_1}{\partial x_2} + V_3 \frac{\partial V_1}{\partial x_3} &= \\ = F_1 - \frac{1}{\rho} \frac{\partial p}{\partial x_1} + \nu \nabla^2 V_1; \\ \frac{\partial V_2}{\partial t} + V_1 \frac{\partial V_2}{\partial x_1} + V_2 \frac{\partial V_2}{\partial x_2} + V_3 \frac{\partial V_2}{\partial x_3} &= \\ = F_2 - \frac{1}{\rho} \frac{\partial p}{\partial x_2} + \nu \nabla^2 V_2; \\ \frac{\partial V_3}{\partial t} + V_1 \frac{\partial V_3}{\partial x_1} + V_2 \frac{\partial V_3}{\partial x_2} + V_3 \frac{\partial V_3}{\partial x_3} &= \\ = F_3 - \frac{1}{\rho} \frac{\partial p}{\partial x_3} + \nu \nabla^2 V_3; \\ \frac{\partial V_1}{\partial x_1} + \frac{\partial V_2}{\partial x_2} + \frac{\partial V_3}{\partial x_3} &= 0 \end{aligned} \quad (3)$$

или кратко, в векторной форме:

$$\begin{aligned} \frac{\partial V}{\partial t} + (V \cdot \nabla) V &= F - \frac{1}{\rho} \nabla p + \nu \Delta V; \\ \nabla \cdot V &= 0, \end{aligned} \quad (4)$$

где  $t$  — время;  $\rho$  — плотность;  $p$  — давление;  $V$  — скорость;  $\nu$  — коэффициент кинематической вязкости;  $F$  — векторное поле массовых сил.

В анализе решений данных уравнений заключается суть одной из открытых проблем, за решение которых Математический институт Клэя назначил

Основные существующие модели турбулентности и стратегии моделирования

Модель турбулентности	Особенности моделей
<b>I. DNS</b>	Учет всех структур, присущих турбулентным течениям; сверхвысокие требования к компьютерным ресурсам
<b>II. LES</b>	Учет крупных структур, присущих турбулентным течениям, высокие требования к компьютерным ресурсам
<b>III. RANS</b>	
<i>Первого порядка</i>	
Алгебраические уравнения	Эмпирическая информация, простота калибрования; привязка к определенным видам течений
С одним дифференциальным уравнением	Полуэмпирическая информация; простота калибрования; учет ряда эффектов, присущих турбулентным течениям
С двумя дифференциальными уравнениями	Не требуют эмпирических оценок для каждого течения; изотропная турбулентная вязкость/диффузия
Нелинейные уравнения	Анизотропная турбулентная вязкость/диффузия; учет многих эффектов, строгая привязка к калибровочным коэффициентам при нелинейных членах
EARSM/EASFM	Анизотропная турбулентная вязкость/диффузия; учет многих эффектов турбулентности; нахождение коэффициентов при нелинейных членах из упрощенного алгебраического уравнения, получаемого из моделей второго порядка
<i>Второго порядка</i>	
RSM/RSFM	Учет многих эффектов; сложны; требуют решения 7/3 дополнительных нелинейных дифференциальных уравнений для компонент тензора рейнольдсовых напряжений и скалярного переноса
<i>Примечание.</i> Порядком модели в данном случае называют порядок соответствующего дифференциального уравнения	

премию в 1 млн долларов США. Необходимо доказать или опровергнуть существование глобального гладкого решения задачи Коши для трехмерных уравнений Навье—Стокса. Нахождение общего аналитического решения системы Навье—Стокса для пространственного или плоского потока осложняется тем, что оно нелинейное и в значительной степени зависит от начальных и граничных условий. Нелинейность системы обусловлена наличием конвективной составляющей в левой части уравнений (3) [2].

До сих пор решения этих уравнений найдены лишь для некоторых частных случаев. В настоящее время существует несколько ситуаций (обусловленных простой геометрией), которые решены в аналитическом виде. В остальных случаях используется численное моделирование.

При описании реальных течений в разных технических устройствах приемлемую точность в численном решении можно получить лишь для такой расчетной сетки, в которой ячейки будут меньше, чем самый мелкий вихрь. Это определяет очень большие затраты по расчетному времени на современных компьютерах. В связи с этим создавались разные модели турбулентности, которые упрощают расчеты по реальным потокам.

### Модели турбулентности

На настоящий момент есть большое число моделей турбулентности, но ни одна среди существующих моделей не будет универсальной по всем классам инженерных задач. Существуют определенные отличия трех основных методов описания турбулентности: DNS, LES и RANS, разрешенных по времени [3].

DNS и LES позволяют получить информацию о нестационарных эффектах турбулентного течения, RANS оперирует осредненными величинами. Для моделирования нестационарных явлений в RANS используется методика URANS [4].

Процесс выбора оптимальных моделей турбулентности определяется типом течения, специфическим классом задачи, необходимой точностью решения, доступными вычислительными ресурсами и т.п. В табл. 1 приведены основные недостатки и преимущества моделей и подходов для исследования турбулентных течений.

### Проведение прямого численного моделирования (DNS)

Прямое численное моделирование (DNS) исходит из решения полных нестационарных уравнений Навье—Стокса и уравнения неразрывности, при этом нет необходимости в дополнительном моделировании и идет учет по всем эффектам, присущим течению. В DNS все физические и химические масштабы длины и времени должны быть разрешены на численной сетке и во временных шагах.

Трудности DNS заключаются в ограниченности компьютерных ресурсов. Число требуемых сеточных узлов и временных шагов существенно превосходят современные вычислительные мощности. Размер ячейки  $\eta_k$  должен удовлетворять колмогоровскому масштабу длины

$$\eta_k = \left(\frac{\nu^3}{\varepsilon}\right)^{1/4}, \quad (5)$$

а временной шаг — колмогоровском масштабу времени:

$$\tau_k = \left(\frac{\nu}{\varepsilon}\right)^{1/2}. \quad (6)$$

В указанных формулах  $\varepsilon$  — скорость диссипации.

Таблица 2

**Время расчета турбулентного течения в канале с DNS при различных числах Рейнольдса**

Re	$5 \cdot 10^3$	$5 \cdot 10^4$	$5 \cdot 10^5$	$5 \cdot 10^6$	$5 \cdot 10^8$
200 Mflop/s	68 дней	444 дня	610 лет	$\infty$	$\infty$
1 Tflop/s	13 дней	88 дней	122 года	$\infty$	$\infty$

В табл. 2 представлена показывающая зависимость времени расчета от числа Рейнольдса для канала [5].

Очевидно, что DNS позволяет получать результаты при современном развитии вычислительной техники только при малых числах Рейнольдса. Кроме этого, существует проблема постановки граничных условий. Для описания случайного характера турбулентного течения необходимо адекватное задание нестационарных входных и выходных условий. С практической точки зрения статистика, полученная с DNS, может быть использована для тестирования и калибровки моделей, базирующихся на осредненных уравнениях Навье—Стокса.

### Проведение моделирования крупных вихрей (LES)

Когда моделируют крупные вихри, то они напрямую разрешаются на численной сетке и по времени, а моделируют мельчайшие вихри в подсеточном масштабе. В наибольших вихрях есть максимум рейнольдсовых напряжений, и его необходимо рассчитать. В мелких же вихрях есть небольшие рейнольдсовые напряжения, также мелкомасштабные турбулентности близки к изотропным и имеют характеристики, близкие к универсальным, они легче поддаются моделированию. Подсеточные модели, используемые в LES, обычно имеют значительную диффузию и диссипацию, что дает возможность преодолевать вычислительные проблемы, которые касаются представления весьма мелких вихрей на выбранных сетках, и стабилизировать численные расчеты. Основное отличие LES от DNS заключается в концепции процедуры фильтрации для LES, т.е. разделения мелкомасштабных и крупных структур. Математически разделение величины  $\bar{\phi}$  и мелких структур  $\phi'$  через обобщенный фильтр [6] записывается как

$$\begin{aligned} \phi &= \bar{\phi} + \phi'; \\ \bar{\phi}(x, t) &= \iiint \phi(x - x^*, t) G(x^*) d^3x^*, \end{aligned} \quad (7)$$

здесь  $x^*$  — переменная интегрирования.

Фильтрующая функция  $G$  нормализуется как

$$\iiint G(x - \varepsilon, \Delta) d^3\varepsilon = 1, \quad (8)$$

здесь  $\Delta = (\Delta x \Delta y \Delta z)^{1/3}$  — ширина фильтра.

Выбор функции для фильтра — один из центральных пунктов в моделировании крупных вихрей [6].

Ниже приведены некоторые из широко используемых фильтров.

Осредненный по объему коробочный фильтр:

$$G(x - \varepsilon; \Delta) = \begin{cases} \frac{1}{\Delta^3}, & |x_i - \varepsilon_i| < \frac{\Delta x_i}{2}; \\ 0, & |x_i - \varepsilon_i| > \frac{\Delta x_i}{2}. \end{cases} \quad (9)$$

Фильтр Гаусса:

$$G(x - \varepsilon; \Delta) = \left( \frac{6}{\pi \Delta^2} \right)^{3/2} \exp \left( -6 \frac{|x_i - \varepsilon_i|^2}{\Delta^2} \right). \quad (10)$$

Сокращенный фильтр Фурье:

$$G(x - \varepsilon; \Delta) = \frac{1}{\Delta^3} \prod_{i=1}^3 \frac{\sin(x_i - \varepsilon_i)/\Delta}{(x_i - \varepsilon_i)/\Delta}. \quad (11)$$

Существует множество других фильтров, однако все фильтры вводят масштаб  $\Delta$ , который представляет наименьший масштаб турбулентности, допустимый фильтром.

Отфильтрованные уравнения Навье—Стокса и уравнение неразрывности имеют следующий вид:

$$\begin{aligned} \frac{\partial \bar{u}_i}{\partial x_i} &= 0; \\ \frac{\partial \bar{u}_i}{\partial t} + \frac{\partial \bar{u}_i \bar{u}_j}{\partial x_j} &= -\frac{1}{\rho} \frac{\partial \bar{p}}{\partial x_i} + \nu \frac{\partial^2 \bar{u}_i}{\partial x_j \partial x_j} - \frac{\partial \tau_{ij}^{SGS}}{\partial x_i}, \end{aligned} \quad (12)$$

где подсеточные напряжения  $\tau_{ij}^{SGS}$  вычисляются по следующей формуле:

$$\tau_{ij}^{SGS} = \bar{u}_i \bar{u}_j - \bar{u}_i \bar{u}_j. \quad (13)$$

Здесь  $\bar{\mu}$  — средний коэффициент вязкости среды,  $\bar{p}$  — среднее давление.

Член  $\bar{u}_i \bar{u}_j$  является новым неизвестным, который показывает влияние мельчайших структур, индекс  $j$  показывает дополнительное разбиение сетки. Сокращение SGS (Sub Grid Stress — напряжения подсеточного масштаба) подчеркивает, что  $\tau_{ij}^{SGS}$  играет схожую роль в отфильтрованных уравнениях, как и тензор рейнольдсовых напряжений в RANS. Иногда  $\tau_{ij}^{SGS}$  называют тензором напряжений мельчайших структур.

### Модель Смагоринского

Большинство моделей подсеточного масштаба основываются на предположении о вихревой вязкости. Влияние мельчайших структур на крупные структуры учитывается прибавлением вязкости  $\mu_t$ :

$$\mu_{eff} = \mu + \mu_t. \quad (14)$$

Для неизвестного подсеточного тензора напряжений  $\tau_{ij}^{SGS}$  используется следующее модельное уравнение:

$$\tau_{ij}^{SGS} - \frac{1}{3} \tau_{ij}^{SGS} \delta_{ij} \approx \tau_{ij}^{mod} \delta_{ij} = 2\mu_t \bar{S}_{ij}. \quad (15)$$

Здесь  $\delta_{ij}$  — оператор Кронекера,  $\tau_{ij}^{mod}$  — подсеточный тензор напряжений, используемый в данной модели,  $\bar{S}_{ij}$  — тензор скоростей деформации.

Тензор скоростей деформации выражается через крупные структуры как

$$\bar{S}_{ij} = \frac{1}{2} \left( \frac{\partial \bar{u}_i}{\partial x_j} - \frac{\partial \bar{u}_j}{\partial x_i} \right). \quad (16)$$

Существует множество моделей для определения турбулентной вязкости  $\mu_t$ . Первая модель для подсеточных напряжений была предложена Смагоринским. По аналогии с моделью Прандтля турбулентная вязкость моделируется как

$$\mu_t = l^2 |\bar{S}_{ij}|, \quad (17)$$

где  $|\bar{S}_{ij}| = \sqrt{\bar{S}_{ij} \bar{S}_{ij}}$ , а  $l$  представляет собой масштаб турбулентности. Для определения масштаба турбулентности используют ширину фильтра с коэффициентом пропорциональности  $C_S$ :

$$l = C_S \Delta. \quad (18)$$

Коэффициент пропорциональности  $C_S$  носит название *константы Смагоринского*. Так же, как и в модели Прандтля, коэффициент пропорциональности  $C_S$  можно скалировать. Диапазон выбора  $C_S$  для различных течений колеблется от 0,1 до 0,24. Обычно выбирается  $C_S = 0,1$ .

Модель Смагоринского отличается простотой и хорошей сходимостью вследствие ее диссипативности. Однако имеются и значительные недостатки, например сложность правильного выбора константы  $C_S$ . Так как модель базируется на статистическом анализе, невозможно наблюдать нестационарные эффекты между разрешенными крупными структурами и подсеточными масштабами, как, например, при кратковременных переходах турбулентной кинетической энергии от мелких структур к крупным (*back-scattering*). Другой недостаток заключается в том, что модель Смагоринского не делает различий между турбулентными и ламинарными градиентами, поэтому в переходных зонах незначительные возмущения демпфируются и не наблюдается перехода от ламинарного к турбулентному течению. Обойти данные недостатки позволяет модель, использующая динамическую процедуру, предложенную Германо.

В модели Германо коэффициент пропорциональности  $C_S$  рассчитывается с помощью динамической процедуры. Идея динамической процедуры базируется на предположении о схожем характере малых вихрей крупных структур и вихрей подсеточного масштаба. Таким образом, появляется возможность оценить коэффициент пропорциональности (по названию модели — *константа Германо*) из разрешенных крупных структур. Динамическая процедура не зависит от используемой модели для подсеточных структур. Для осуществления данной процедуры используют два фильтра: обычный и тестовый. При тестовой фильтрации отделяется часть крупных структур, которые рассматриваются как новые отфильтрованные мельчайшие структуры, из которых можно найти значения обычных отфильтрованных мельчайших структур  $\tau_{ij}^{SGS}$ . Таким образом, в области между обычным фильтром  $\Delta$  и тестовым фильтром  $\hat{\Delta}$  находится необходимая информация для явного нахождения мельчайших структур и их сравнения с модельными.

Уравнение импульсов при тестовом фильтровании имеет вид

$$\frac{\partial \hat{u}_i}{\partial t} + \frac{\partial \hat{u}_i \hat{u}_j}{\partial x_j} = -\frac{1}{\rho} \frac{\partial \hat{p}}{\partial x_j} + \nu \frac{\partial^2 \hat{u}_i}{\partial x_j \partial x_j} - \frac{\partial T_{ij}^{SGS}}{\partial x_j}, \quad (19)$$

где подсеточные напряжения  $T_{ij}^{SGS}$  вычисляются по следующей формуле:

$$T_{ij}^{SGS} = \widehat{\overline{u_i u_j}} - \widehat{u_i} \widehat{u_j}, \quad (20)$$

$\rho$  — плотность жидкости.

Нормальные мельчайшие структуры из уравнений (13) и (20) для тестового фильтра можно записать как

$$\widehat{\overline{\tau_{ij}^{SGS}}} = \widehat{\overline{u_i u_j}} - \widehat{u_i} \widehat{u_j}. \quad (21)$$

Из двух последних соотношений можно записать:

$$L_{ij} = T_{ij} - \widehat{\overline{\tau_{ij}^{SGS}}} = \widehat{u_i} \widehat{u_j} - \widehat{\overline{u_i u_j}}. \quad (22)$$

Для разрешения мелких структур для обоих фильтров используется модель Смагоринского:

$$\tau_{ij}^{SGS} - \frac{1}{3} \tau_{ij}^{SGS} \delta_{ij} = -2C\Delta^2 |\bar{S}| \bar{S}_{ij}; \quad (23)$$

$$\tau_{ij}^{SGS} - \frac{1}{3} \tau_{ij}^{SGS} \delta_{ij} = -2C\hat{\Delta}^2 |\hat{S}| \hat{S}_{ij}. \quad (24)$$

В обоих уравнениях (23) и (24) константа Смагоринского  $C_S$  заменена на константу Германо  $C$ . Подставляя (23) и (24) в (22), можно найти конс-

танту Германо, а через величины тестового фильтра получить осредненную величину:

$$L_{ij} = -2C\hat{\Delta}^2 \left[ \frac{\hat{\Delta}^2}{\hat{\Delta}} |\hat{S}| \hat{S}_{ij} \right] = -2CM_{ij}.$$

Здесь  $M_{ij}$  — вспомогательный тензор. Отсюда

$$C = \frac{L_{ij}M_{ij}}{2M_{ij}^2}.$$

Значение константы Германо может сильно колебаться в пространстве и во времени, что может привести к отрицательной вязкости. Это означает перенос энергии от подсеточных масштабов к крупным структурам. Такие процессы происходят в реальности, однако это ведет к вычислительной нестабильности. Чтобы обойти этот недостаток, можно пренебречь отрицательными значениями константы Германо и применять осреднение. После осреднения демпфируются высокочастотные гармоники, т.е. на  $(n + 1) - m$  временном шаге

$$C_{\text{filtered}}^{(n+1)} = (1 - \varepsilon)C^n + \varepsilon C^{(n+1)},$$

где  $\varepsilon$  — коэффициент нижней релаксации.

Так как LES включает моделирование мельчайших вихрей, характерные размеры ячеек могут быть намного больше, чем колмогоровский масштаб длины, а временные шаги можно выбрать более крупные, чем они возможны в DNS. С учетом бурного развития техники области использования LES значительно возросли. Есть мнение, что LES постепенно вытеснит RANS. Указанную точку зрения можно считать весьма спорной. До настоящего времени в LES не решили проблему пристеночных течений. Понятно, что около стенок вихри будут малыми и анизотропными, и сеточные временные шаги, необходимые для LES, падают для величин, характерных для DNS. Существующие решения, такие как анизотропные фильтры и описанные выше динамические процедуры, пока не дают удовлетворительных результатов. Одним из решений данной проблемы является комбинирование LES и RANS. Главная идея таких гибридных моделей состоит в том, что в пристеночной зоне используется RANS, а во внешней области — LES, что позволяет значительно увеличить скорость вычислений. В настоящее время существуют две стратегии соединения LES и RANS: так называемое изолированное (зональное) LES и присоединенное DES. При зональном методе используются две отдельные численные сетки — одна для LES и вторая для RANS. Такое моделирование носит название "двухслойное моделирование TLM" (Two-Layer Model) и было впервые предложено Балларасом и Беночи [9]. В DES используется одна сетка и осуществляется только

переход от LES к RANS в пристеночной зоне. Более подробно о DES изложено в работе Спаларта [10]. Однако в гибридных моделях до сих пор остаются не полностью решенными проблемы, например, смешивание зон, т.е. передача информации между зонами. Несмотря на существующие проблемы, моделирование крупных вихрей является перспективным направлением в исследовании турбулентных течений в настоящее время.

### Моделирование на базе осредненных уравнений Стокса (RANS)

Этот метод базируется на решении осредненных уравнений Навье—Стокса или уравнений Рейнольдса.

При этом мгновенные значения гидродинамических параметров представляют как сумму осредненной величины  $\bar{\phi}$  и ее пульсационной составляющей  $\phi'$ :

$$\phi(x_j, t) = \bar{\phi}(x_j, t) + \phi'(x_j, t). \quad (25)$$

В интегральной форме осреднение по времени можно записать как

$$\phi(\chi) = \lim_{t \rightarrow \infty} \frac{1}{t} \int_0^t \phi(x_j, t) dt. \quad (26)$$

Применяя данное осреднение, названное *осреднением по Рейнольдсу*, можно получить определенные уравнения по сохранению масс, импульсов и скаляров (энергия, температура, массовая доля) [11]. Для несжимаемой жидкости это записывается в следующем виде:

$$\frac{\partial \bar{u}_i}{\partial x_j} = 0; \quad (27)$$

$$\frac{\partial \bar{u}_i}{\partial t} + \frac{\partial \bar{u}_i \bar{u}_j}{\partial x_j} = -\frac{1}{\rho} \frac{\partial \bar{p}}{\partial x_i} - \frac{\partial}{\partial x_j} \left( \nu \frac{\partial \bar{u}_i}{\partial x_j} + \overline{\phi' u_i' u_j'} \right) + g_i; \quad (28)$$

$$\rho \frac{\partial \bar{\phi}}{\partial t} + \rho \bar{u}_j \frac{\partial \bar{\phi}}{\partial x_j} = \frac{\partial}{\partial x_j} \left( \rho D \frac{\partial \bar{\phi}}{\partial x_j} - \overline{\phi' u_j'} \right) + \bar{S}, \quad (29)$$

где  $\overline{u_i' u_j'}$  — компоненты тензора рейнольдсовых напряжений;  $\overline{\phi' u_j'}$  — составляющие потока скаляра.

Составляющие тензорных рейнольдсовых напряжений являются дополнительными шестью неизвестными в уравнении для скаляра. В результате указанная выше система уравнений будет незамкнутой.

При расчете двухфазных течений методом Эйлера—Лагранжа дополнительно к уравнениям (27)—(29) решаются аналогичные уравнения для частиц. Полученные решения суммируются. При методе Эйлера—Лагранжа учет частиц происходит

добавлением источниковых членов в уравнения (28)—(29), и решение происходит итерациями между двумя фазами.

### Предположение Буссинеска

Модели турбулентности, используемые в инженерных приложениях в настоящее время, основаны на концепции вязкости и турбулентности диффузии. В 1877 г. Буссинеск [12] выдвинул предположение, что реинольдсовы напряжения могли быть связаны со скоростью средней деформации через турбулентную вязкость. Для тензора реинольдсовых напряжений это дает:

$$-\overline{u'_i u'_j} = \nu_t \left( \frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) - \frac{2}{3} k \delta_{ij}, \quad (30)$$

где  $k$  — удельная кинетическая энергия турбулентности.

Данное уравнение не определяет модель турбулентности, а лишь характеризует, какая структура подобной модели. При этом основной задачей является задание функции турбулентной вязкости  $\nu_t$ . В отличие от коэффициента молекулярной вязкости  $\nu$ , коэффициент  $\nu_t$  определяется состоянием турбулентного течения и не связан со свойствами жидкости. Значение  $\nu_t$  может значительно изменяться от точки к точке в пространстве в зависимости от характера течения. Предположение Буссинеска относят к моделям первого уровня.

Данное предположение не выполняется даже во многих простых течениях, например в течении в трубе, вращающейся вокруг своей оси, не говоря уже о более сложных течениях. Вместе с тем, во многих случаях, особенно при анализе течений, в которых основное влияние оказывает лишь одна из компонент тензора реинольдсовых напряжений  $\tau_{xy}$ , недостаток гипотезы Буссинеска не приводит к заметным погрешностям.

### Алгебраические модели

В алгебраических моделях турбулентности (модели нулевого порядка) связь между тензорами реинольдсовых напряжений и характеристиками среднего движения задается алгебраическими соотношениями. В теории Прандтля принимается, что местное изменение средней скорости потока определяется первой производной от средней скорости по поперечной координате:

$$\tau_{xy} = \rho l^2 \left( \frac{\partial \bar{u}}{\partial y} \right)^2, \quad (31)$$

где  $x$  — координата, совпадающая с направлением потока,  $y$  — поперечная координата. Соображения размерности приводят к введению линейного масштаба  $l$  (масштаба турбулентности или пути перемешивания), который определяется с помощью дополнительных гипотез или данных измерений.

Формула Прандтля оставляет некоторые возможности для учета интегральных свойств потока и его предыстории благодаря неопределенности пути смещения. В модели Прандтля движение частиц жидкости в турбулентном потоке полагается подобным движению молекул (масштаб турбулентности соответствует длине свободного пробега молекул, а пульсационная скорость — их тепловой скорости). Турбулентная вязкость связывается со средней скоростью посредством масштаба длины (пути смещения), на котором жидкие частицы сохраняют средние значения количества движения. Несмотря на то что концепция вихря играет существенную роль в построении полуэмпирических моделей турбулентности и, в частности, в модели Прандтля, она является абстрактной, поскольку преследует одну цель — схематично изобразить турбулентное течение.

### Заключение

В настоящий момент сформировано большое число разных моделей для проведения расчетов турбулентных течений. Они различаются сложностью решения и точностью описания течения.

Был проведен обзор основных моделей турбулентности и выявлены их слабые и сильные стороны. Полученные результаты могут быть полезны при выборе модели турбулентности для конкретной задачи в подсистемах моделирования.

### Список литературы

1. **Зеленый Л. М., Милованов А. В.** Фрактальная топология и странная кинетика: от теории перколяции к проблемам космической электродинамики // Успехи физических наук. 2004. Т. 174, № 8. С. 809—852.
2. **Лойцянский Л. Г.** Механика жидкости и газа. М.-Л.: Гос-техиздат, 1950, 676 с.
3. **Вихреразрешающее** моделирование как один из методов описания турбулентных течений. URL: <http://www.inm.ras.ru/laboratory/direct2.htm>
4. **Yun A.** Development and Analysis of Advanced Explicit Algebraic Turbulence and Scalar Flux Models for Complex Engineering Configurations. Doctor thesis. Darmstadt. 2005. 103 p.
5. **Menter F. R.** Methoden, Moeglichkeiten und Grenzen numerischer Stroemungsberechnungen. Numet. Erlangen. 2002. 114 p.
6. **Wachter E. M.** Anwendung der Instationären Flamelet Methode auf Diffusions Flammen im Post-Processing-Modus. VDI Verlag. 2005. 195 p.
7. **Smagorinsky J.** General circulation experiments with the primitive equations. I: The basic experiment // Monthly Weather Review. 1963. Vol. 91, N. 3. P. 99—165.
8. **Piomelli U.** Large-Eddy Simulation: Achievements and Challenges // Progress in Aerospace Sciences. 1999. Vol. 35. P. 335—362.
9. **Balaras E., Benocci C., Piomelli U.** Two-Layer Approximate Boundary Conditions for Large-Eddy Simulations // AIAA Journal. 1996. Vol. 34, N. 6. P. 1111—1119.
10. **Spalart P. R.** Detached-eddy simulation // Annual Review Fluid Mechanic. 2009. Vol. 41. P. 181—202.
11. **Wilcox D. C.** Turbulence modeling for CFD. California, 1994. 460 p.
12. **Boussinesq J.** Theorie de l' Ecoulemen Tourbillant. Mem // Presents par Divers Savants Acad. Sci. Inst. Fr. 1977. Vol. 23. P. 46—50.
13. **Волков К. Н., Емельянов В. Н.** Моделирование крупных вихрей в расчетах турбулентных течений. М.: Физматлит, 2008, 370 с.



I. Ya. Lvovich<sup>1</sup>, Professor, office@vivt.ru, Ya. E. Lvovich<sup>2</sup>, Professor, office@vivt.ru,  
A. P. Preobrazhensky<sup>2</sup>, Professor, app@vivt.ru, O. N. Choporov<sup>2</sup>, Professor, choporov\_oleg@mail.ru  
<sup>1</sup>Paneuropean university, Bratislava  
<sup>2</sup>Voronezh institute of high technologies, Voronezh

## The Features of the Methods of Computational Fluid Dynamics for Turbulence Modeling

*In the paper an analytical review of methods for turbulence modeling on personal computers is carried out. Turbulent flow can be represented as a multiple rotating in different directions and in different planes of vortices. The computational features of the methods that can be used in mathematical software subsystems for the simulation of turbulence in the relevant CAD are provided. The results that computational fluid dynamics can take a dominant place, surpassing the prevalence of experimental and analytical methods are shown. These differences in the three main methods of description of turbulence are discussed. It was provided that, the development of computer technology and numerical methods of computational physics, allows to work not only with increasingly complex models, but also to apply different ways of visualizing the obtained result. The obtained results can be helpful when selecting a turbulence model for a specific task in the simulation subsystems.*

**Keywords:** turbulence, modeling, software, CAD, computational physics, equation, vortex, integral engineering application, liquid, method

### References

1. Zeleny L. M., Milovanov A. V. Fraktalnaya topologiya i stran-naya kinetika: ot teorii perkolyacii k problemam kosmicheskoy elektrodinamiki, *Uspehi fizicheskikh nauk*, 2004, vol. 174, no. 8, pp. 809–852.
2. Loycyanskiy L. G. *Mehanika zhidkosti i gaza*, Moscow—Leningrad, Gostehizdat, 1950. 676 s.
3. Vihreerazreshayuschee modelirovanie kak odin iz metodov opisaniiya turbulentnykh techeniy, URL: <http://www.inm.ras.ru/laboratory/direct2.htm>
4. Yun A. *Development and Analysis of Advanced Explicit Algebraic Turbulence and Scalar Flux Models for Complex Engineering Configurations*. Doctor thesis. Darmstadt. 2005, 103 p.
5. Menter F. R. *Methoden, Moeglichkeiten und Grezen numerischer Stromungsberechnungen*. Numet. Erlangen, 2002, 114 p.
6. Wachter E. M. *Andwendung der Instationären Flamelet Methode auf Diffusions Flammen im Post-Processing-Modus*. VDI Verlag. 2005, 195 p.
7. Smagorinsky J. General circulation experiments with the primitive equations. I: The basic experiment, *Monthly Weather Review*, 1963, vol. 91, no. 3, pp. 99–165.
8. Piomelli U. Large-Eddy Simulation: Achievements and Challenges, *Progress in Aerospace Sciences*, 1999, vol. 35, pp. 335–362.
9. Balaras E., Benocci C., Piomelli U. Two-Layer Approximate Boundary Conditions for Large-Eddy Simulations, *AIAA Journal*, 1996, vol. 34, no. 6, pp. 1111–1119.
10. Spalart P. R. Detached-eddy simulation, *Annual Review Fluid Mechanics*, 2009, vol. 41, pp. 181–202.
11. Wilcox D. C. *Turbulence modeling for CFD*. California, 1994, 460 p.
12. Boussinesq J. *Theorie de l' Ecoulemen Tourbillant*. Mem, *Presentes par Divers Savants Acad. Sci. Inst. Fr.*, 1977, vol. 23, pp. 46–50.
13. Volkov K. N., Emelyanov V. N. *Modelirovaniye krupnykh vihrey v raschetah turbulentnykh techeniy*, Moscow, Fizmatlit. 2008. 370 p.

УДК 519.87

Е. М. Бронштейн, д-р физ.-мат. наук, проф., e-mail: bro-efim@yandex.ru,  
А. А. Давлетбаев, аспирант, e-mail: Arvid\_NF@mail.ru,  
Уфимский государственный авиационный технический университет, Уфа

## Задача маршрутизации с учетом зависимости стоимости транспортировки от погодных условий

*Рассмотрена задача построения маршрута, по которому можно доставить однородный груз от пункта производства множеству потребителей транспортным средством ограниченной вместимости с минимальными затратами. Стоимость транспортировки груза между пунктами зависит от времени. Построены квадратичная и линейная целочисленные модели. Проведены вычислительный эксперимент решения задач различной размерности в среде IBM ILOG CPLEX Optimization studio 12.6.1 и сравнительный анализ эффективности моделей.*

**Ключевые слова:** маршрутизация, нестационарность, линейное целочисленное программирование, метод ветвей и отсечений

### Введение

Рассматривается следующая задача. Транспортное средство (ТС) должно доставить однородный

груз из пункта производства потребителям. Процесс погрузки и разгрузки является трудоемким, поэтому в каждом пункте потребления ТС разгру-

жается один раз. Состояние дорог изменяется каждый день, поэтому стоимость проезда между пунктами зависит от времени. Задана соответствующая величина расхода бензина для каждого дня. ТС должно ежедневно возвращаться на базу по окончании перевозок, если они в этот день выполнялись. Вводится ограничение на расстояние, которое может проехать ТС в течение одного дня. Требуется сформировать график перевозок с минимальной суммарной стоимостью.

Впервые оптимизационная задача транспортной логистики (VRP — Vehicle Routing Problem) была сформулирована в работе [1]. За время более чем полвека сформулировано множество подобных задач, в постановке которых учитываются различные ограничения, возникающие на практике. Классификация таких задач приведена, например, в [2]. На сайте [3] аккумулируется информация об этой области исследований. Применяется множество методов решения, как точных, так и эвристических.

Задача, сформулированная в данной статье, опирается на ряд задач транспортной логистики, ранее описанных в работах [4–9]. Нестационарность в данной работе понимается так же, как при постановке нестационарной задачи коммивояжера [10, 11].

В [12] рассматривалась задача, в которой ТС должно было осуществить выезд (и, соответственно, въезд) с базы только один раз, при этом ограничивалось число переездов.

При реализации некоторых из алгоритмов в тех или иных пакетах прикладных программ может понадобиться переход от логических переменных  $b$  к их числовым значениям  $[b]$ . Опишем подобное приведение с использованием линейных неравенств. Заметим, что для пакета CPLEX, который мы использовали, подобного преобразования не требуется.

Пусть  $X \subset [-a, a]$  — переменная, принимающая конечное множество значений.

Введем булевы переменные  $(x)_+$ ,  $(x)_-$ ,  $(x)_0$ :  $X \rightarrow \{0, 1\}$  — индикаторы неотрицательности, неположительности и нулевого значения (если  $X$  такие значения принимает).

**Предложение 1.** Булеву переменную  $(x)_+$  можно задать системой линейных неравенств  $(x)_+ \leq 1 + x/a$ ,  $(x)_+ \geq (x + 0,5m)/(a + 0,5m)$ , где  $m = \min\{|x|: x \in X, x \neq 0\}$ .

Действительно, при  $x \geq 0$  неравенства принимают вид  $(x)_+ \leq 1 + p$ ,  $(x)_+ \geq r$ , где  $p > 0$ ,  $r \in (0, 1]$ , т.е.  $(x)_+ = 1$ . При  $x < 0$  неравенства принимают вид  $(x)_+ < 1$ ,  $(x)_+ \geq r$ , где  $r < 0$ , т.е.  $(x)_+ = 0$ .

Для  $(x)_-$  аналогичное представление можно получить из того, что  $(x)_- = (-x)_+$ . Наконец,  $(x)_0 = (x)_+ + (x)_- - 1$ .

В частности, если переменная  $X$  целочисленная, то можно принять  $m = 1$ . Далее мы не будем явно выписывать подобные преобразования.

Математические модели, построенные в данной работе, ориентированы на применение специализированных пакетов прикладных программ, в частности, использован оптимизационный пакет CPLEX. В связи с этим важно, чтобы число ограничений задачи росло полиномиально с ростом размерности. В [13] такие модели названы компактными.

**Обозначения:**

$N$  — число пунктов потребления, пронумерованных числами  $1, \dots, N$  (база имеет номер 0);

$a_i, i = 1, \dots, N$ , — потребность в грузе в  $i$ -м пункте потребления;

$S$  — вместимость ТС;

$P$  — число дней доставки (предусмотрено, что в некоторые дни доставка может не проводиться);

$b^p \geq 1, p = 1, \dots, P$ , — коэффициент расхода бензина в  $p$ -й день в зависимости от погодных условий, нормальным погодным условиям соответствует  $b^p = 1$ ;

$r_{ij}^p, i, j = 0, \dots, N, p = 1, \dots, P$ , — расстояние (км) между  $i$ -м и  $j$ -м пунктами в  $p$ -й день, оно может изменяться в связи с ремонтными работами и т.д.;

$G$  — максимальное расстояние, которое ТС может проехать при нормальных погодных условиях за один день (км); полагаем, что в зависимости от погодных условий допустимое расстояние уменьшается в  $b^p$  раз.

**Квадратичная модель.** В качестве неизвестных примем булевы переменные:

$x_{ij} (i, j = 0, \dots, N)$ , равные 1 тогда и только тогда, когда ТС переезжает из  $i$ -го пункта в  $j$ -й;

$y_i^p (i = 0, \dots, N, p = 1, \dots, P)$ , равные 1 тогда и только тогда, когда ТС посещает  $i$ -й пункт в  $p$ -й день; считаем, что  $y_0^p = 1$  при всех  $p$  (если в  $p$ -й день груз не доставляется, то  $y_i^p = 1$  только при  $i = 0$ ).

**Ограничения:**

$$\sum_{p=1}^P y_i^p = 1 \quad (i = 1, \dots, N); \tag{1}$$

$$\sum_{j=0}^N x_{ij} = 1 \quad (i = 1, \dots, N); \tag{2}$$

$$\sum_{i=0}^N x_{ij} = 1 \quad (j = 1, \dots, N); \tag{3}$$

$$x_{jj} = 0 \quad (j = 1, \dots, N). \tag{4}$$

Условие (1) означает, что каждый пункт потребления посещается в некоторый день (единственный), условие (2) — что в каждый пункт потребления ТС въезжает, условие (3) — что из каждого

пункта потребления ТС выезжает. Смысл условия (4) очевиден.

$$x_{ij} \leq 2 - \sum_{p=1}^P [y_i^p \neq y_j^p] \quad i, j = 1, \dots, N. \quad (5)$$

Условие (5) означает, что если есть переезд из  $i$ -го пункта в  $j$ -й, то эти пункты посещаются в один день. Действительно, в силу (1) каждому пункту соответствует единственное значение  $p$  — номер дня. Тем самым, если эти пункты посещаются в один день, то  $\sum_{p=1}^P [y_i^p \neq y_j^p] = 0$ ; если в разные, то

$\sum_{p=1}^P [y_i^p \neq y_j^p] = 2$ . Отсюда, если пункты посещаются в разные дни, то  $x_{ij} = 0$ , и наоборот.

По условию (1) все пункты потребления разбиваются на непересекающиеся подмножества, соответствующие отдельным дням.

Рассмотрим граф, который порождается переездами в какой-либо день. Из условий (2), (3) следует, что каждая связная компонента графа является либо простой цепью, либо простым циклом, за исключением 0-го пункта, который может входить в несколько цепей или циклов.

По условиям (2), (3), (5) простой цепью связная компонента быть не может. Действительно, в этом случае начальный или конечный пункт цепи (пусть начальный) ненулевой, для определенности  $j$ -й. По условию (2)  $x_{ij} = 1$  для некоторого  $i = 0, \dots, N$ . Противоречие.

Таким образом, связные компоненты являются либо простыми циклами, не содержащими 0-й пункт, либо объединением простых циклов, содержащих 0-й пункт.

Пусть целые числа  $l_j$  удовлетворяют условиям

$$1 \leq l_j \leq N \quad (j = 0, \dots, N); \quad (6)$$

$$l_0 = 0; \quad (7)$$

$$l_j \leq l_i + x_{ij} + (1 - x_{ij})N \quad (i = 0, \dots, N, j = 1, \dots, N); \quad (8)$$

$$l_j \geq l_i + x_{ij} - (1 - x_{ij})N \quad (i = 0, \dots, N, j = 1, \dots, N). \quad (9)$$

Из условий (8), (9) при  $x_{ij} = 1$  получаем  $l_j = l_i + 1$ , при  $x_{ij} = 0$  какое-либо дополнительное ограничение на  $l_j$  не возникает.

**Предложение 2.** В построенном графе нет циклов, не содержащих 0.

**Доказательство.** Пусть связная компонента рассматриваемого графа есть цикл, не содержащий 0, для определенности,  $x_p, x_j, x_k$ .

Тогда получаем:

$$l_j = l_i + 1,$$

$$l_k = l_j + 1 = l_i + 2,$$

$$l_i = l_k + 1 = l_i + 3 \text{ — противоречие} \bullet.$$

Тем самым граф имеет единственную связную компоненту, которая является объединением не-

скольких циклов с общим нулевым пунктом. Отсюда с учетом (7) значения  $l_j$  определяются однозначно, причем эти значения — номера пунктов от нулевого в порядке прохождения цикла.

Для обеспечения единственности цикла используем условие

$$x_{0i} + x_{0j} + y_i^p + y_j^p \leq 3 \quad (i, j = 1, \dots, N, p = 1, \dots, P, i \neq j). \quad (10)$$

При выполнении условия (10), если ТС выезжает в какие-нибудь различные пункты из нулевого, то это происходит в разные дни.

Ограничение на вместимость

$$\sum_{i=1}^N y_i^p a_i \leq S, \quad p = 1, \dots, P; \quad (11)$$

ограничение на пройденный путь в течение дня

$$\sum_{j=1}^{N-1} x_{0j} y_j^p r_{ij}^p + \sum_{i=1}^N \sum_{j=0}^N x_{ij} y_i^p r_{ij}^p \leq G/b^p, \quad p = 1, \dots, P. \quad (12)$$

**Целевая функция** квадратичной модели имеет вид

$$\sum_{p=1}^P \sum_{i=0}^N \sum_{j=0}^N x_{ij} y_i^p b^p r_{ij}^p \rightarrow \min. \quad (13)$$

**Линейная модель.** В качестве неизвестных прием булевы переменные  $z_{ij}^p$  ( $i, j = 0, \dots, N, p = 1, \dots, P$ ), равные 1 тогда и только тогда, когда ТС переезжает из  $i$ -го пункта в  $j$ -й в  $p$ -й день.

**Ограничения:**

$$\sum_{p=1}^P \sum_{j=0}^N z_{ij}^p = 1, \quad i = 1, \dots, N; \quad (14)$$

$$\sum_{p=1}^P \sum_{i=0}^N z_{ij}^p = 1, \quad j = 1, \dots, N \quad (15)$$

(каждый пункт, кроме 0, посещается однократно);

$$\sum_{i=0}^N z_{ij}^p = \sum_{e=0}^N z_{je}^p, \quad j = 0, \dots, N, p = 1, \dots, P \quad (16)$$

(выезд из пункта и въезд в него осуществляются в один и тот же день);

$$\sum_{i=1}^N \sum_{j=1}^N z_{ij}^p a_i \leq S, \quad p = 1, \dots, P \quad (17)$$

(недопустимость переполнения ТС каждый день);

$$\sum_{i=0}^N \sum_{j=0}^N z_{ij}^p r_{ij}^p \leq G/b^p, \quad p = 1, \dots, P \quad (18)$$

(ограничение на расстояние, проезжаемое за один день);

$$1 \leq l_j \leq N \quad (j = 0, \dots, N); \quad (19)$$

$$l_0 = 0; \quad (20)$$

$$l_j \leq l_i + \sum_{p=1}^P z_{ij}^p + \left(1 - \sum_{p=1}^P z_{ij}^p\right)N, \quad (21)$$

$$i = 0, \dots, N, j = 1, \dots, N;$$

$$l_j \geq l_i + \sum_{p=1}^P z_{ij}^p - \left(1 - \sum_{p=1}^P z_{ij}^p\right)N, \quad (22)$$

$$i = 0, \dots, N, j = 1, \dots, N;$$

$$l_j \leq \sum_{i=0e=1}^N \sum_{e=1}^N z_{ie}^p + (1 - z_{j0}^p)N, \quad (23)$$

$$j = 1, \dots, N, p = 1, \dots, P;$$

$$l_j \geq \sum_{i=0e=1}^N \sum_{e=1}^N z_{ie}^p - (1 - z_{j0}^p)N, \quad (24)$$

$$j = 1, \dots, N, p = 1, \dots, P.$$

Условия (21), (22) аналогичны условиям (8), (9), а условия (23), (24) не допускают наличия более чем одного цикла, включающего в себя базу (нулевой пункт), в течение одного дня.

**Целевая функция** линейной модели имеет вид

$$\sum_{p=1}^P \sum_{i=0}^N \sum_{j=0}^N z_{ij}^p b^p r_{ij}^p \rightarrow \min. \quad (25)$$

### Вычислительный эксперимент

Вычисления проводились в пакете IBM ILOG CPLEX Optimization studio 12.6.1, установленном на компьютере с процессором Intel Core (тактовая частота 2,4 ГГц, ОЗУ 4 Гбайт).

Задача решалась двумя методами, реализованными в пакете:

- динамического поиска,
- ветвей и отсечений.

Первый метод — это "ноу-хау" фирмы IBM, никакие подробности не разглашаются. Применяемый в пакете метод ветвей и отсечений основан на непрерывных релаксациях.

В процессе вычислений исключались заведомо нулевые величины  $r_{ii}^p$  ( $i = 1, \dots, N, p = 1, \dots, P$ );  $x_{ii}$  ( $i = 0, \dots, N$ );  $z_{ii}^p$  ( $i = 1, \dots, N, p = 1, \dots, P$ ).

Веса  $a_i$  ( $i = 2, \dots, N$ ) генерировались случайным образом (в программе Pascal ABC с помощью функции *random*) целыми из отрезка [1; 20], отражающие состояние погоды, коэффициенты,  $b^p$  ( $p = 1, \dots, P$ ) — из отрезка [1; 2].

Генерация расстояний происходила следующим образом:

1) случайно генерировались координаты точек плоскости таким образом, чтобы расстояние между ними не превышало 100 км;

2) расстояние между полученными точками увеличивалось на 100 км, таким образом расстояние варьировалось в пределах [100, 200] км;

3) генерировались изменения полученного расстояния в какие-либо дни: если для заданных дня,

Квадратичная модель

$N$	Динамический метод	Метод ветвей и отсечений
4	$4,4 \cdot 10^4/1,6 \cdot 10^3/27,34$	$4,5 \cdot 10^4/1,6 \cdot 10^3/27,61$
5	$6,2 \cdot 10^5/2,4 \cdot 10^4/741,61$	$6,2 \cdot 10^5/2,4 \cdot 10^4/724,41$

Таблица 2

Линейная модель

$N$	Динамический метод	Метод ветвей и отсечений
4	$2,7 \cdot 10^2/11/0,28$	$3,8 \cdot 10^2/29/0,37$
5	$2,8 \cdot 10^3/461/0,57$	$4,2 \cdot 10^3/826/0,70$
6	$9,4 \cdot 10^3/1,3 \cdot 10^3/1,12$	$1,3 \cdot 10^4/2,2 \cdot 10^3/1,33$
7	$7,7 \cdot 10^4/8,7 \cdot 10^3/5,09$	$1,0 \cdot 10^5/1,4 \cdot 10^4/4,46$
8	$3,0 \cdot 10^5/3,4 \cdot 10^4/23,29$	$4,5 \cdot 10^5/6,4 \cdot 10^4/20,95$
9	$1,5 \cdot 10^6/1,5 \cdot 10^5/115,11$	$2,8 \cdot 10^6/3,6 \cdot 10^5/137,63$
10	$5,2 \cdot 10^6/4,5 \cdot 10^5/550,89$	$9,0 \cdot 10^6/1,0 \cdot 10^6/627,75$

пунктов въезда и выезда функция *random*(6) возвращала число 5, на этот и все последующие дни для текущего переезда задавалось изменение расстояния не более чем на 50 км (также с помощью функции *random*) и в итоге окончательный диапазон расстояний составлял [50, 250] км. Например, если расстояние между пунктами  $i$  и  $j$  с 1-го по  $(s-1)$ -й день 150 км, а с  $s$ -го дня начались ремонтные работы, то с  $s$ -го по  $P$ -й день расстояние составит 200 км (придется ехать в обход).

Вместимость  $S$  аналогично [14] принималась равной  $2\max\{a_i\}$ . Число дней принималось равным числу пунктов, т.е.  $P = N + 1$ .

Для каждого числа пунктов ( $N$ ) генерировалось 10 примеров. Результаты эксперимента отражены в табл. 1 (квадратичная модель) и табл. 2 (линейная модель), в ячейках приведены последовательно среднее число итераций, среднее число ветвей в дереве решений и среднее время выполнения (в секундах).

Для решения задачи при большем числе пунктов ресурсов ПК не хватило. Квадратичная модель оказалась очень ресурсно-затратной.

Линейная модель оказалась эффективнее квадратичной. Из таблицы видно, что по времени решения при  $N < 9$  существенной разницы между двумя методами не наблюдается, а при  $N = 9, 10$  более эффективен динамический метод. Для задач размерностей 4, 5 значения целевых функций, полученные при использовании обеих моделей, естественно, совпали.

### Заключение

При сравнении квадратичной и линейной моделей эффективнее по времени решения и размерности решаемых задач оказалась линейная модель. При этом по числу переменных квадратичная мо-

дель (число переменных порядка  $N^2 + NP$ , ограничений порядка  $N^2P$ ) существенно экономнее линейной модели (число переменных порядка  $N^2P$ , ограничений порядка  $N\max\{N, P\}$ ). Динамический метод оказался более эффективным, нежели метод ветвей и границ. Дальнейший прогресс связан с совершенствованием как моделей, так и программных средств. В частности, как отметила К. Archetti (Брешиа, Италия) в докладе на 3-м совещании Европейской рабочей группы VeRoLog (Осло, 2014 г) [15], CPLEX 11 (2007) работает почти в 30 000 раз быстрее, чем CPLEX 1 (1991).

#### Список литературы

1. **Dantzig G. B.** The Truck Dispatching Problem / Dantzig G. B., Ramser R. H. // *Management Science*. 1959. N. 6. P. 80–91.
2. **Бронштейн Е. М., Заико Т. А.** Детерминированные оптимизационные задачи транспортной логистики // *Автоматика и телемеханика*. 2010. № 10. С. 133–147.
3. **URL:** <http://neo.lcc.uma.es/vrp/>
4. **Ralphs T. K.** On the Capacitated Vehicle Routing Problem // *Math. Program., Ser. B*. 2003. Vol. 94. P. 343–359.
5. **Laporte G., Nobert Y., Taillefer S.** Solving a family of multi-depot vehicle routing and location-routing problems // *Transportation Science*. 1988. Vol. 22. P. 161–172.

6. **Crevier B., Cordeau J.-F., Laporte G.** The multi-depot vehicle routing problem with inter-depot routes // *European Journal of Operational Research*. 2007. Vol. 176. P. 756–773.

7. **Malandraký C., Daskin M. S.** Time Dependent Vehicle Routing Problems: Formulations, Properties and Heuristic Algorithms Malandraký // *Transportation Science*. 1992. Vol. 26. P. 185–200.

8. **Ichoua S., Gendreau M., Potvin J.-Y.** Vehicle Dispatching With Time-Dependent Travel Times // *European Journal of Operational Research*. 2003. Vol. 144. P. 379–396.

9. **Stegers E.** A Solution Method for Vehicle Routing Problems with Time-Dependent Travel Times. Delft: Delft University of Technology, 2009.

10. **Picard J.-C., Queyranne M.** The time-dependent traveling salesman problem and its application to the tardiness problem in one-machine scheduling // *Operations Research*. 1978. Vol. 26. N. 1. P. 86–110.

11. **Gouveia L., Vob S.** A classification of formulations for the (time-dependent) traveling salesman problem // *European Journal of Operational Research*. 1995. Vol. 83. P. 69–82.

12. **Бронштейн Е. М., Давлетбаев А. А.** Нестационарная задача построения маршрута транспортного средства // *Проблемы управления*. 2014. № 3. С. 23–28.

13. **Furtadoa M. R., Munaria P., Morabito R.** Pickup and delivery problem with time windows: a new compact two-index formulation. URL: [www.optimization-online.org/DB\\_HTML/2015/07/5022.h](http://www.optimization-online.org/DB_HTML/2015/07/5022.h).

14. **Бронштейн Е. М., Гиндуллин Р. В.** Точные решения некоторых оптимизационных задач транспортной логистики // *Матем. моделирование*. 2013. Т. 25, № 11. С. 121–127.

15. **URL:** <http://www.sintef.no/contentassets/cfb19ab9b7c74>

**Е. М. Bronstein**, Sc. D., Professor, e-mail: bro-efim@yandex.ru,

**A. A. Davletbaev**, Postgraduate Student, e-mail: Arvid\_NF@mail.ru,  
Ufa State Aviation Technical University

## Vehicle Routing Problem with Transport Cost Depending on Weather

*The problem of constructing a cyclic route with minimal transport cost for delivering a homogeneous cargo from a production point to consumers by limited capacity vehicle is considered. The cost of transportation between points depends on time. The corresponding integer linear and quadratic models are constructed. Computer experiments solving problems of different dimensions and comparative analysis efficiency of models were carried out in environment IBM ILOG CPLEX Optimization studio 12.6.1.*

**Keywords:** routing, nonstationarity, linear integer programming, branches and cuts method

#### References

1. **Dantzig G. B., Ramser R. H.** The Truck Dispatching Problem. *Management Science*, 1959, no. 6, pp. 80–91.
2. **Bronshtein E. M., Zaiko T. A.** Deterministic optimization problems of transportation logistics, *Automation and Remote Control*, 2010, vol. 71, no. 10, pp. 2132–2144.
3. **URL:** <http://neo.lcc.uma.es/vrp/>
4. **Ralphs T. K., Kopman L., Pulleyblank W. R., Trotter L. E. Jr.** On the Capacitated Vehicle Routing Problem, *Math. Program., Ser. B*, 2003, vol. 94, pp. 343–359.
5. **Laporte G., Nobert Y., Taillefer S.** Solving a family of multi-depot vehicle routing and location-routing problems, *Transportation Science*, 1988, vol. 22, pp. 161–172.
6. **Crevier B., Cordeau J.-F., Laporte G.** The multi-depot vehicle routing problem with inter-depot routes, *European Journal of Operational Research*, 2007, vol. 176, pp. 756–773.
7. **Malandraký C., Daskin M. S.** Time Dependent Vehicle Routing Problems: Formulations, Properties and Heuristic Algorithms, *Transportation Science*, 1992, vol. 26, pp. 185–200.
8. **Ichoua S., Gendreau M., Potvin J.-Y.** Vehicle Dispatching With Time-Dependent Travel Times, *European Journal of Operational Research*, 2003, vol. 144, pp. 379–396.

9. **Stegers E. A.** *Solution Method for Vehicle Routing Problems with Time-Dependent Travel Times*. Delft: Delft University of Technology, 2009, vol. 83.

10. **Picard J.-C., Queyranne M.** The time-dependent traveling salesman problem and its application to the tardiness problem in one-machine scheduling, *Operations Research*, 1978, vol. 26, no. 1, pp. 86–110.

11. **Gouveia L., Vob S.** A classification of formulations for the (time-dependent) traveling salesman problem, *European Journal of Operational Research*, 1995, vol. 83, pp. 69–82.

12. **Bronshtein E. M., Davletbaev A. A.** Nestacionarnaja zadacha postroeniya marshruta transportnogo sredstva, *Problemy upravleniya*, 2014, no. 3, pp. 23–28. (Non-stationary problem of creation of a route of the vehicle // *Problems of management* 2014, no. 3, pp. 23–28.)

13. **Furtadoa M., Munaria P., Morabito R.** Pickup and delivery problem with time windows: a new compact two-index formulation. URL: [www.optimization-online.org/DB\\_HTML/2015/07/5022.h](http://www.optimization-online.org/DB_HTML/2015/07/5022.h)

14. **Bronshtein E. M., Gindullin R. V.** Tochnye resheniya nekotorykh optimizatsionnykh zadach transportnoy logistiki. Exact solutions of some optimization problems of transport logistics, *Mathematical Models and Computer Simulations*, 2014, vol. 6, no. 3, pp. 332–336.

15. **URL:** <http://www.sintef.no/contentassets/cfb19ab9b7c74>

**В. Б. Ефлов**<sup>1</sup>, канд. физ.-мат. наук, доц. кафедры, e-mail: veflov@gmail.com,

**Е. А. Питухин**<sup>1</sup>, д-р техн. наук, проф., зав. каф., e-mail: eugene@petrsu.ru,

**Ф. В. Стафеев**<sup>2</sup>, консультант, e-mail: fedor.stafeev@gmail.com

<sup>1</sup>ГОУ ВПО "Петрозаводский государственный университет (ПетрГУ)"

<sup>2</sup>Администрация Петрозаводского городского округа

## Вопросы производительности при выполнении запросов в реляционной СУБД H2

*Исследованы статистические характеристики времени выполнения запросов, выявлены особенности работы режимов кэширования данных. С помощью нелинейной аппроксимации данных получена обобщенная модель зависимости времени выполнения запросов от объема кэш-памяти и размера запрашиваемых таблиц. Введен безразмерный показатель, на основе которого предложен критерий реализации в СУБД H2 режима эффективного кэширования. Результаты оценки эффективности кэширования могут быть использованы на практике для существенного снижения времени выполнения запросов.*

**Ключевые слова:** оптимизация запросов, СУБД, кэширование, H2, модель, время исполнения запросов

### Введение

Для осуществления полномочий современных органов местного самоуправления используется широкий спектр инструментов информационно-коммуникационных технологий (ИКТ). Одной из ключевых составляющих в подобном комплексе является применение СУБД в различных информационных системах. В силу ряда причин наиболее распространены в данной сфере реляционные СУБД, подчас являющиеся свободным либо свободно распространяемым программным обеспечением.

Несмотря на развитые возможности оптимизаторов таких продуктов, при промышленной эксплуатации информационных систем повсеместно встает вопрос о производительности модулей выполнения запросов. В повседневной практике запросы выборки данных, выраженные в случае языка программирования SQL посредством оператора SELECT, имеют большую значимость в силу частоты использования перед запросами других типов. Традиционные методы сокращения времени выполнения запросов, основанные на перезаписи SQL-выражений, не всегда успешны, так как администратор информационных систем подчас не имеет доступа к программному коду и не имеет возможности вносить какие-либо изменения. Также этот метод имеет точечный характер и наиболее успешен в ситуации, когда в запросе участвуют соединения нескольких отношений [1].

В представленной работе описана методика оценки производительности модуля выполнения запросов свободно распространяемой реляционной СУБД H2 при варьировании параметров аппаратно-программной платформы и предикатов SQL-выражений. Исследованы статистические характеристики времени выполнения запросов, выявлены особенности работы режимов кэширова-

ния данных. С помощью нелинейной аппроксимации экспериментальных данных получена обобщенная модель зависимости времени выполнения запросов от объема кэш-памяти и размера запрашиваемых таблиц. Введен безразмерный показатель относительного объема кэша в виде отношения объема кэша к объему базы данных, на основе которого предложен критерий реализации в СУБД H2 режима эффективного кэширования. Полученные результаты оценки эффективности кэширования могут быть использованы на практике для существенного снижения времени выполнения запросов.

Данная работа является продолжением серии исследований производительности некоммерческих реляционных СУБД [2–4]. В более ранней статье [2] не был затронут ряд вопросов, выявленных при работе с экспериментальной платформой на основе H2 Database. H2 Database — реляционная СУБД, созданная и поддерживаемая сообществом разработчиков (<http://www.h2database.com>).

### Описание эксперимента

Результатом проведенных экспериментов стал массив данных о времени выполнения запросов, выраженных посредством оператора SELECT, к экспериментальной базе данных H2 Database при варьировании ряда параметров: объема оперативной памяти платформы, размера используемой СУБД кэш-памяти, числа кортежей в таблицах, к которым выполняется обращение в запросах, типов полей предикатов, значения идентификаторов уникальных ключей таблиц [5]. Результаты эксперимента были представлены в виде выборок зависимостей времени выполнения запросов от номера идентификатора первичного ключа таблиц экспериментальной базы данных (далее — пакет запросов) (рис. 1). Номер идентификатора первичного ключа таблиц выражен как относительный от общего

числа записей в таблицах коэффициент и варьировался в диапазоне от 1 до 40. В общей сложности были получены и подвергнуты анализу более 5000 экспериментальных выборок.

Следует отметить, что данные времен, полученных при выполнении пакета запросов, отличаются первыми значениями, которые обычно имеют большую величину (рис. 1, а). Для исследований в ряде случаев использовался набор данных без первого значения (рис. 1, б), например для оценки периодичности [2].

Такая большая разница в значениях между первым и последующим запросами к одной и той же таблице может иметь следующее объяснение. При включении механизма кэширования во время выполнения первого запроса вначале происходит чтение таблицы с жесткого диска и загрузка ее в ОЗУ. При последующих запросах к таблице данные загружаются из ОЗУ и обращения к диску уже не происходит. В связи с этим время выполнения последующих запросов, по сравнению с первым, сокращается на несколько порядков.

При этом возникает проблема оценки эффективности кэширования, которая будет рассмотрена ниже. Целесообразно определить, какой объем кэш-памяти необходимо использовать при кэшировании, чтобы время выполнения первого и последующих запросов существенно различалось, как, например, на рис. 1, а.

### Статистический анализ экспериментальных данных

Для повышения достоверности полученных в ходе экспериментов результатов была проведена дополнительная серия измерений с выборочными значениями варьируемых в основном эксперименте параметров. Для каждой из комбинаций значений параметров время выполнения запроса измерялось десятикратно. Запросы выполнялись для двух таблиц из списка тестовых таблиц. Определено, что из 3214 уникальных запросов только для 35 выборок коэффициент вариации был больше 33 %. Таким образом, экспериментальные выборки можно считать однородными.

Исследовались зависимости времени выполнения запроса от размера кэша при одном объеме оперативной памяти (230 Мбайт) для таблиц двух разных объемов — 10 000 и 500 000 строк.

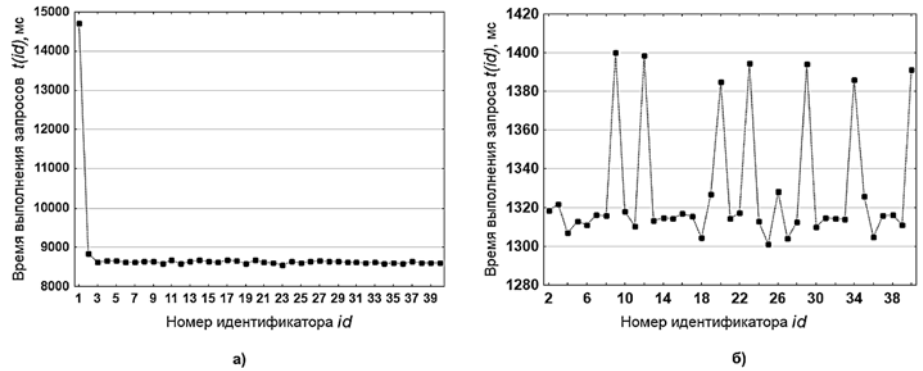


Рис. 1. Экспериментальная зависимость времени выполнения запросов от номера идентификатора первичного ключа  $t(id)$ : а — с первым значением; б — без первого значения

Статистические характеристики времени выполнения запроса для таблицы размером 10 000 строк от объема кэша приведены на рис. 2, а. Из графиков видно, что эффективная работа кэша начинается после 6000 кбайт, после чего падение среднего времени выполнения запроса составляет более 16 раз (218,2 с против 13,0 с), а падение минимального времени выполнения запроса достигает 30 раз! Максимальное время запроса при этом остается флуктуировать близко к уровню, когда кэш отсутствует или его малый размер не оказывает значительного эффекта. Среднее относительное отклонение среднего времени выполнения запроса от минимального составляет 36,5 %.

Иная картина получается при анализе работы кэша на таблице большего размера — 500 000 строк. На рис. 2, б также приведены среднее, максимальное и минимальное время выполнения запросов в зависимости от размера кэша. На рис. 2, б видно, что устойчивое падение времени выполнения запроса начинается со значений кэша от 1000 кбайт и более.

Минимальное время выполнения запроса отличается от среднего незначительно: на всем промежутке среднее относительное отклонение не превышает 2,1 %. Максимальное время выполнения запроса представляет собой характерный выброс в начале кэширования и не оказывает значительного влияния на среднее значение. Среднее время вы-

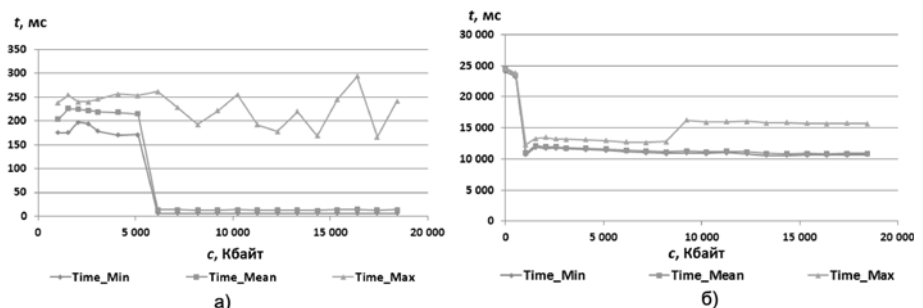
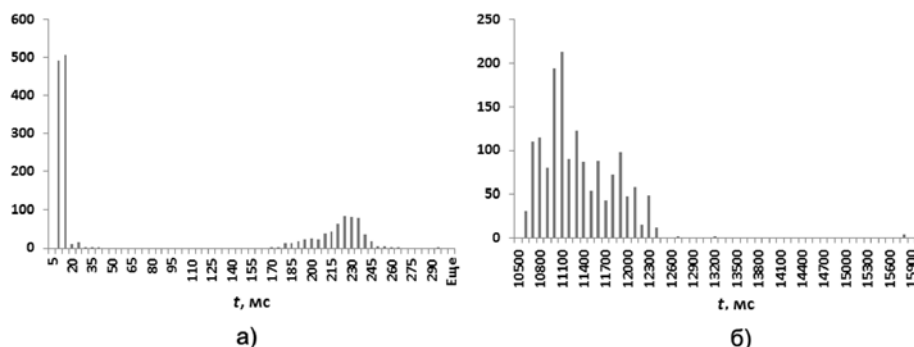


Рис. 2. Зависимость характеристик времени выполнения запроса от величины кэша: а — размер таблицы 10 000 строк; б — размер таблицы 500 000 строк



**Рис. 3. Гистограмма частот времени выполнения запросов:**  
*a* — размер таблицы 10 000 строк, *б* — размер таблицы 500 000 строк

полнения запроса после начала работы кэша с эффективным значением более 1000 кбайт падает более чем в 2 раза (23 630 мс против 11 060 мс).

Несмотря на снижение среднего времени выполнения запроса более чем в 2 раза, эффективность кэширования данных для таблицы размером 500 000 значительно меньше, чем для таблицы размером 10 000 строк (16 раз). Это может объясняться тем, что для такой большой таблицы использованные в эксперименте вариации объема кэша оказались недостаточными, чтобы кэширование перешло по-настоящему в эффективный режим.

На рис. 3, *a* и *б* приведены гистограммы частот времени выполнения запросов для таблиц размером 10 000 и 500 000 строк соответственно.

Рис. 3, *a* отображает бимодальный характер плотности распределения времени выполнения запросов для таблицы размером 10 000 строк. Хорошо различимы две моды, разделенные значительным расстоянием, соответствующие "быстрым" и "медленным" запросам. При дополнительной детализации "быстрых" запросов выяснилось, что они также характеризуются бимодальным распределением, состоящим из двух непересекающихся нормальных распределений. Представленная на рис. 3, *б* плотность распределения времени выполнения запросов для таблицы размером 500 000 строк, в от-

личие от рис. 3, *a*, составляет единую картину.

Результаты проведенного анализа показывают, что при использовании в СУБД функции кэширования ее эффективность может быть различной.

Данные факты позволяют сделать заключение о существовании двух режимов кэширования при использовании кэш-памяти в процессе выполнения запросов — "эффективного" и "неэффективного" (на порядки отличающихся по эффективности в плане срав-

нения времен выполнения запросов). Под "неэффективным" кэшированием будем понимать такой режим, когда функция кэша включена, но кэширование фактически не происходит из-за малого объема кэш-памяти.

В связи с этим необходимо провести дополнительное исследование эффективности кэширования на различных режимах.

### Оценка эффективности кэширования

Как было отмечено в [2], зависимость  $t(id)$  принимает вид, приведенный на рис. 1, *a*, в случае участия механизма кэширования данных при первом обращении к запрашиваемой таблице. В выборках, полученных при отключенной функции кэширования, первое значение также больше среднего, однако разница не превышает 20...30 %.

Представляется важным проанализировать условия, при которых наступает режим "эффективного" кэширования, когда время выполнения запросов падает в десятки раз.

Для этого по результатам экспериментов было вычислено среднее время выполнения запросов, содержащихся в одном пакете, за исключением первого запроса (см. рис. 1, *б*). В каждом пакете содержались запросы типа SELECT к определенной

**Время выполнения запросов от объема кэша и размера таблиц в случае использования функции кэширования, мс**

		Число строк таблицы										
		1000	3000	5000	10 000	30 000	50 000	80 000	100 000	200 000	300 000	400 000
Размер кэша, кбайт	2048	0,87	46,49	74,47	152,18	443,58	745,14	1256,74	1495,35	3079,65	4961,83	7537,06
	4096	1,03	2,59	3,91	163,24	477,94	789,87	1329,79	1568,01	3237,35	5180,99	8059,73
	8192	1,09	2,42	3,88	7,53	467,59	784,60	1317,74	1573,55	3154,13	5094,82	7964,23
	12 288	0,92	2,76	4,15	7,78	453,72	749,41	1256,60	1495,82	3121,82	4999,51	7612,15
	20 480	0,92	2,56	3,91	7,65	21,78	789,77	1322,23	1572,47	3249,23	5176,44	8017,19
	24 576	1,03	2,67	4,01	7,65	21,96	823,06	1378,18	1640,90	3402,40	5416,79	8236,40
	30 720	0,96	2,71	4,15	7,90	22,32	37,03	1514,38	1800,53	3675,22	5823,85	8819,88
	40 960	1,06	2,50	3,92	7,99	22,77	36,83	1711,08	2058,26	4204,81	6652,35	9902,96
	51 200	1,01	2,62	4,09	7,90	22,69	36,85	61,72	3996,27	7874,94	11590,32	16507,67
	57 344	0,95	2,64	4,36	8,05	22,54	36,62	63,79	80,05	19506,95	29462,72	40091,90



таблице с предикатом поля с заданным типом данных. Среднее время выполнения запросов рассчитывалось для каждого из типов данных полей предикатов.

Пример зависимости среднего времени выполнения запросов от объема кэша и размера таблиц приведен в таблице. Запросы осуществлялись по предикату целочисленного поля для зафиксированного значения оперативной памяти из середины диапазона вариации.

Данные в правой части таблицы соответствуют режиму "неэффективного" кэширования (выделено курсивом). Левая часть таблицы содержит значения времени выполнения запроса (выделено жирным) в случае, когда кэширование "эффективное". Это можно заметить при анализе столбцов таблицы сверху вниз — в среднем время выполнения запросов по сравнению с "неэффективным" режимом снижается более чем в 22 раза. Для таблиц больших размеров перехода из режима "неэффективного" кэширования в "эффективный" режим не наблюдается, что объясняется, очевидно, ограниченным экспериментом диапазоном изменения кэш-памяти. При этом отмечается обратный эффект, когда в режиме "неэффективного" кэширования с увеличением кэш-памяти время выполнения запросов не снижается, а, наоборот, лавинообразно возрастает.

На основе данных из таблицы была получена зависимость времени выполнения запросов от объема данных запрашиваемой таблицы для различных значений кэш-памяти (рис. 4). При этом в качестве значения оси ординат выбирались значения времени выполнения запросов, соответствующих режиму "эффективного" кэширования.

Очевидно, что полученные экспериментальные зависимости (рис. 4) хорошо аппроксимируются линейной функцией вида

$$t = a_1 s, \quad (1)$$

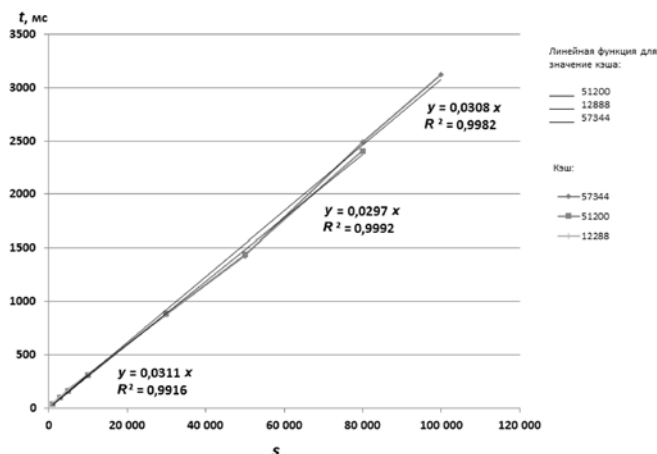


Рис. 4. Зависимость времени выполнения запросов от объема данных запрашиваемой таблицы в случае использования в СУБД "эффективного" кэширования

где  $t$  — время выполнения запроса;  $s$  — размер таблицы (число строк);  $a_1$  — коэффициент линейной регрессии.

При этом угол наклона аппроксимирующих прямых в режиме "эффективного" кэширования остается постоянным для любых значений кэша, что свидетельствует о том, что время запроса преимущественно зависит только от размера таблицы.

В ходе эксперимента выполнялись запросы с предикатами по различным типам полей. Влияние этого фактора на вид зависимости (рис. 4) не отмечалось. Это вполне предсказуемо, если принять за факт предположение о том, что кэшируются, по всей видимости, все данные запрашиваемой таблицы.

Графики времени выполнения запросов из таблицы для всех режимов кэширования представлены на рис. 5 (см. третью сторону обложки) в логарифмическом масштабе из-за сильно отличающихся значений. Они представляют собой зависимости логарифма среднего времени выполнения запроса от размеров таблицы при постоянном значении кэша.

Приведенные графики обнаруживают интересную особенность. Поведение времени выполнения запроса имеет нелинейность в виде скачкообразного перехода от режима "эффективного" кэширования к режиму "неэффективного" кэширования, когда размер таблицы увеличивается выше некоторого критического уровня для заданного объема кэша.

На рис. 5 (см. третью сторону обложки) в результате наложения графиков сформировались асимптоты, соответствующие процессам "эффективного" и "неэффективного" кэширования. Среднее отношение времени выполнения запросов между двумя асимптотами составляет более 22 раз, что доказывает существенное различие между эффективностью указанных режимов.

Для выявления особенностей работы режимов кэширования данных рассмотрим детальнее таблицу. По диагонали таблицы проходит граница между "эффективным" и "неэффективным" режимами кэширования, ячейки которой обведены рамкой. Они соответствуют минимальному размеру кэша для соответствующего размера таблицы, при котором кэширование переходит в "эффективный" режим.

Результаты анализа "перехода" между процессами "эффективного" и "неэффективного" кэширования в относительном виде отображены на рис. 6. На графике представлена экспериментальная кривая, показывающая процентное отношение объема кэша к объему таблицы, при превышении которого таблица соответствующего размера начнет "эффективно" кэшироваться. Объем таблицы  $v$  (кбайт) рассчитывался исходя из ее размера (числа строк) по формуле

$$v = \frac{l_1}{1024} s,$$

где  $l_1$  — средний размер строки таблицы в байтах ( $l_1 = 94$ ).

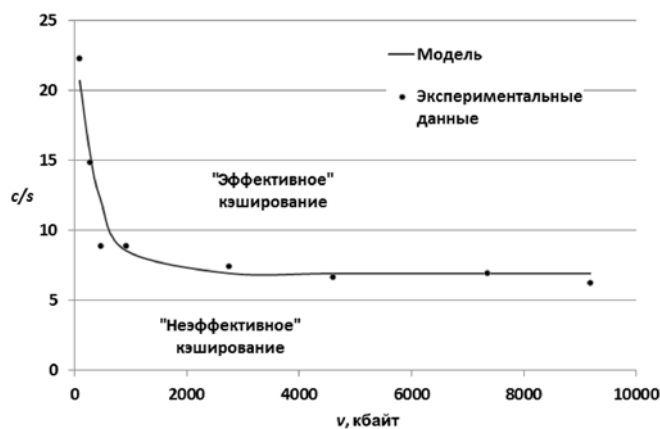


Рис. 6. Критические области режимов "эффективного" и "неэффективного" кэширования для таблиц различных размеров в зависимости от объема кэша

На рис. 6 видно, что при малых объемах таблицы объем кэша составляет до 20 раз от ее объема, но, при увеличении размера таблицы, для обеспечения "эффективного" кэширования доля кэша должна более чем в 7 раз превосходить объем запрашиваемой таблицы.

Проведенный анализ показывает, что для обеспечения режима "эффективного" кэширования и, соответственно, малых времен выполнения запросов размер кэша следует выбирать из соображений, чтобы соответствующее соотношение "объем кэша/объем таблицы" располагалось над указанной на рис. 6 кривой.

Данное наблюдение позволяет ввести понятие показателя относительного объема кэша в виде отношения объема кэша к объему запрашиваемой таблицы (базы данных) в безразмерном виде:

$$q(c, v) = c/v. \quad (2)$$

С учетом формулы (2) можно предложить критерий реализации в СУБД Н2 режима "эффективного" кэширования, когда значение показателя относительного объема кэша  $q(c, v)$  должно быть больше некоторого критического значения  $q_{cr}$ :

$$q(c, v) > q_{cr}. \quad (3)$$

Анализ поведения критического значения показателя  $q_{cr}$  в зависимости от размера таблицы  $s$  позволит определить границу перехода между режимами "эффективного" и "неэффективного" кэширования.

На рис. 6 приведена кривая аппроксимации экспериментальных значений показателя  $q_{cr}$  экспоненциальной зависимостью вида

$$q_{cr}(v) = b_0 + b_1 e^{b_2 v}, \quad (4)$$

где  $b_0, b_1, b_2$  — коэффициенты экспоненциальной регрессии. Значения коэффициентов регрессии  $b_0 = 6,933, b_1 = 17,441, b_2 = -2,581 \cdot 10^{-3}$ . Коэффициент детерминации  $R^2 = 0,986$  и средняя относи-

тельная ошибка в процентах  $MAPE = 8\%$  свидетельствуют об удовлетворительной точности аппроксимации.

### Моделирование времени выполнения запросов

Значимым этапом исследования является построение обобщенной модели зависимости времени выполнения запросов от влияющих факторов, которая позволит сделать выводы о характере процессов и способах увеличения производительности СУБД Н2.

Для аппроксимации экспериментальных данных использовались полиномиальные функции различных степеней. Хорошие результаты были получены в случае полинома 2-, 3-й степеней, при котором значение коэффициента детерминации  $R^2$  лежало в пределах  $[0,9...0,95]$ .

В общем случае для разных типов предикатов были получены следующие выражения зависимостей времени выполнения запроса  $t$  от размера кэша  $c$ , размера оперативной памяти  $r$  и размера запрашиваемых таблиц  $s$ :

*вещественный тип:*

$$t(s, c, r) = 2,352s + 0,002c^2 - 19,803c - 0,002r^3 + 0,813r^2 + 78,037r - 0,011rs + 6591,710;$$

*целочисленный тип:*

$$t(s, c, r) = 2,53s - 16,89c + r^2 - 0,01rs + 12539,4;$$

*текстовой тип:*

$$t(s, c, r) = 2,417s + 0,003c^2 - 45,761c + 0,006r^3 - 3,826r^2 + 1084,718r - 0,01rs + 7173,73;$$

*предикат вида LIKE:*

$$t(s, c, r) = 2,46s - 16,44c + 1,08r^2 - 50,25r - 0,01rs + 17183,37;$$

*предикат по первичному ключу:*

$$t(s, c, r) = 0,0024c + 0,0029r^2 - 1,1031r + 134,677.$$

Полученные модели статистически хорошо описывают исследуемые процессы, но в силу своего полиномиального вида имеют недостатки. Полученные коэффициенты при степенях имеют знаки, которые сложно интерпретировать с позиции физического смысла процессов. В некоторых случаях, при аппроксимации малых значений времен запросов, функция  $t(s, c, r)$  может принимать отрицательные значения.

Возникает задача построения полнофункциональной математической модели, отражающей характер приведенных процессов с учетом вышеуказанных недостатков.

В целях построения обобщенной модели, позволяющей провести графическую интерпретацию процессов, из рассмотрения будет исключен фактор размера оперативной памяти  $r$ . Аргументом для

его исключения служит то обстоятельство, что скорее он является ограничением, нежели фактором влияния. Оперативной памяти может либо хватить для полноценной работы СУБД, либо не хватить. Попытки исследовать влияние пограничного состояния размера оперативной памяти на скорость работы СУБД не имеют большого практического значения.

За основу для построения модели были взяты данные из таблицы.

Наличие скачкообразного перехода между режимами "эффективного" и "неэффективного" кэширования обосновывает проведение аппроксимации соответствующих режимов различными функциями. Таким образом, итоговая модель будет состоять из двух функций, связанных условием перехода.

Для аппроксимации режима "эффективного" кэширования (см. рис. 4) использовалась ранее установленная линейная зависимость времени выполнения запросов от объема данных запрашиваемой таблицы вида (1):

$$t_{ef}(v) = a_1 v, \quad (5)$$

где коэффициент линейной регрессии  $a_1 = 8,401 \cdot 10^{-3}$  и средняя относительная ошибка в процентах  $MAPE = 8,5 \%$ .

При выборе аппроксимирующей функции для режима "неэффективного" кэширования соответствующие данные таблицы были подвергнуты тщательному анализу, в результате которого выяснилось, что наилучшим образом они описываются нелинейной зависимостью вида

$$t_{nef}(c, v) = v(d_0 + d_1 c + d_2 e^{d_3 c}), \quad (6)$$

где коэффициенты нелинейной регрессии  $d_0 = 0,189$ ,  $d_1 = 9,653 \cdot 10^{-7}$ ,  $d_2 = 1,437 \cdot 10^{-6}$ ,  $d_3 = 2,315 \cdot 10^{-4}$ . При этом коэффициент детерминации  $R^2 = 0,929$  и средняя относительная ошибка в процентах  $MAPE = 11,1 \%$ .

Из формул (2)—(4) следует, что критерием перехода из "неэффективного" режима в "эффективный" может служить выполнение неравенства

$$c/v > b_0 + b_1 e^{b_2 v}. \quad (7)$$

Используя формулы (5)—(7), запишем итоговую обобщенную математическую модель зависимости времени выполнения запросов от размера кэша  $c$  и объема запрашиваемых таблиц  $s$  с учетом текущего режима кэширования:

$$t(c, v) = \begin{cases} a_1 v, & \text{если } c/v > b_0 + b_1 e^{b_2 v}; \\ v(d_0 + d_1 c + d_2 e^{d_3 c}), & \text{иначе.} \end{cases} \quad (8)$$

Полученная модель (8) показывает, что время выполнения запросов имеет линейный характер в

режиме "эффективного" кэширования и носит экспоненциальный характер в режиме "неэффективного" кэширования. При этом  $t(c, v)$  будет всегда положительно, будет равняться нулю в случае таблицы нулевого объема ( $v = 0$ ) и будет продолжать экспоненциально расти от размера таблицы при нулевом кэше ( $c = 0$ ).

На рис. 7 (см. третью сторону обложки) приведена зависимость времени выполнения запросов от размера кэша  $c$  и объема запрашиваемых таблиц  $v$ . Экспериментальные данные отображены точками, а аппроксимирующая поверхность (8) представлена контуром.

На аппроксимирующей поверхности хорошо виден разрыв, соответствующий границе между "эффективным" и "неэффективным" режимами кэширования.

Кроме того, из рис. 7 видно, что в "неэффективном" режиме кэширования с увеличением объема кэша время выполнения запросов начинает экспоненциально расти, хотя было ожидаемо, что оно начнет снижаться. Для объяснения этого факта необходимо проводить дополнительные углубленные исследования механизма кэширования, что выходит за рамки данной статьи.

## Заключение

В заключение отметим, что проведенные исследования указывают на довольно значительное влияние кэширования на процесс выполнения запросов, особенно на таблицах относительно небольшого объема. Несмотря на наблюдаемый скачкообразный характер влияния кэша на время выполнения запроса в исследуемой СУБД, вызывает интерес тип зависимостей времени выполнения запросов, проиллюстрированных на рис. 1—7. Выяснилось, что при использовании функции кэширования сам процесс кэширования может осуществляться как в "эффективном", так и в "неэффективном" режиме. Полученная в результате аппроксимации обобщенная модель зависимости времени выполнения запросов от размера кэша и объема запрашиваемых таблиц позволяет, по нашему мнению, провести прогнозирование времени выполнения запросов при детерминированных условиях для любого режима работы кэша. Результаты моделирования обнаруживают, что важным, с практической стороны, является обеспечение работы СУБД с использованием функции кэширования в "эффективном" режиме, когда выделяемого объема кэш-памяти достаточно для обеспечения оперативной работы системы. В качестве критерия обеспечения такого режима можно использовать превышение предложенного безразмерного показателя относительно объема кэша некоторого критического уровня.

### Список литературы

1. **Ефлов В. Б., Питухин Е. А., Стафеев Ф. В.** Анализ выполнения запросов в СУБД H2 // Ученые записки Петрозаводского государственного университета. Сер.: Естественные и технические науки. 2013. № 2 (131). С. 62—65.
2. **Питухин Е. А., Стафеев Ф. В.** Анализ методов оптимизации запросов в реляционных СУБД. М., 2010. Деп. ВИНТИ 16.03.2010, № 167-B2010.
3. **Стафеев Ф. В.** Способы оценки эффективности оптимизаторов запросов в реляционных СУБД // Роль науки в устой-

чивом развитии общества: сборник материалов 2-й Международной научно-практической конф. Тамбов: Тамбовпринт, 2010. С. 150—151.

4. **Стафеев Ф. В.** Методы исследования процесса обработки запросов в реляционных СУБД // Современные инструментальные системы, информационные технологии и инновации: материалы VIII Международной научно-технической конференции. Курск, 2011. С. 250—252.

5. **Tutorial.** H2 Database Engine. 2013. URL: <http://www.h2database.com/html/tutorial.html>

**V. B. Yeflov**<sup>1</sup>, Associate Professor, e-mail: [veflov@gmail.com](mailto:veflov@gmail.com),

**E. A. Pitukhin**<sup>1</sup>, Professor, e-mail: [eugene@petsru.ru](mailto:eugene@petsru.ru),

**F. V. Stafeev**<sup>2</sup>, Consultant, e-mail: [fedor.stafeev@gmail.com](mailto:fedor.stafeev@gmail.com)

<sup>1</sup>Petrozavodsk State University

<sup>2</sup>Petrozavodsk City Administration

## Performance Issues of Query Execution in Relational H2 Database

*To exercise the powers of modern local authorities a wide range of ICT tools is used. A key component is the use of DBMS in various information systems. Despite the long-term development of optimizers of DBMS, in industrial information systems operation an issue of improving the performance of the query modules is becoming quite topical.*

*To achieve this goal it is necessary to solve the problem of development of performance assessment methods of the query execution modules of a freely distributable relational DBMS H2 in case of varying parameters of hardware and software platforms and predicates of SQL expressions.*

*The statistical characteristics of the query execution time have been studied; the features of the work of the Cached exchanged mode have been identified. Using the nonlinear approximation of the experimental data the generalized model of dependence of the performance of query execution on the amount of cache memory and the size of the requested tables was obtained. A dimensionless parameter of the relative amount of cache as the volume ratio of the cache to the database volume was introduced, based on which a criterion for the implementation of efficient caching mode in DBMS H2 was proposed.*

*The present study shows a fairly significant impact of caching on the query execution process, especially in the tables of a relatively small volume. The generalized model based on run-time queries of the cache size and the amount of the requested tables, which was obtained as a result of approximation, allows us to conduct forecasting of the query execution time at the determined conditions for any mode of cache operation. The simulation results demonstrate that from the practical perspective it is important to ensure the operation of DBMS H2 with the use of caching feature in the "effective" mode, when the allocated capacity of the cache memory is sufficient for the provision of efficient operation of the system. As a criterion ensuring this mode, it is possible to use exceeding the proposed dimensionless parameter of the relative volume cache against a certain critical level. The achieved results can be used in practice to significantly reduce the query execution time.*

**Keywords:** query optimization, h2, relational database, math modelling, query execution, caching

### References

1. **Eflov V. B., Pitukhin E. A., Stafeev F. V.** Analiz vypolnenija zaprosov v SUBD H2 (Analysis of query execution in H2 Database), *Uchenye zapiski Petrozavodskogo gosudarstvennogo universiteta. Ser.: Estestvennye i tehicheskie nauki*, 2013, no. 2 (131), pp. 62—65.
2. **Pitukhin E. A., Stafeev F. V.** Analiz metodov optimizacii zaprosov v reljacionnyh SUBD (Method's analysis of query optimization in relational DBMS). Moscow, 2010. Dep. VINITI 16.03.2010, no. 167-B2010.
3. **Stafeev F. V.** Spособы ocenki jeffektivnosti optimizatorov zaprosov v reljacionnyh SUBD (Methods of performance evaluation of query optimizers in relational DBMS), *Rol' nauki v ustojchivom*

*razvitiia obshhestva: sbornik materialov 2-j mezhdunarodnoj nauchno-prakticheskoj konferencii, 24—25 dekabrja 2010 g., g. Tambov.* Tambov, Tambovprint, 2010, pp. 150—151.

4. **Stafeev F. V.** Metody issledovaniya processa obrabotki zaprosov v reljacionnyh SUBD (Research method of query execution process in relational DBMS). *Sovremennye instrumental'nye sistemy, informacionnye tehnologii i innovacii: materialy VIII Mezhdunarodnoj nauchno-tehnicheskoj konferencii, 28 dekabrja 2010 g.* Kursk, 2011, pp. 250—252.

5. **Tutorial.** *H2 Database Engine.* URL: <http://www.h2database.com/html/tutorial.html> (01.03.2016)

Э. Ю. Орехов, канд. физ.-мат. наук, доц., e-mail: emil.orekhov@bk.ru,  
Ю. В. Орехов, канд. техн. наук, доц.,  
Уфимский государственный авиационный технический университет

## Сравнение и оценка качества эвристических алгоритмов на конечной массовой задаче в условиях неравновероятной генерации тестовых примеров

*Предложен подход к сравнению и оценке качества эвристических алгоритмов на основе использования математического ожидания случайной величины — случайной реализации критерия качества данного алгоритма на индивидуальных задачах данной конечной массовой задачи, когда равновероятная генерация индивидуальных задач данной конечной массовой задачи затруднительна либо невозможна. Подход основан на введении вспомогательного множества объектов, которое связано с множеством индивидуальных задач исходной конечной массовой задачи и равновероятная генерация элементов которого может быть легко осуществлена. Приведен пример применения предложенного подхода к статистическому оцениванию качества эвристических алгоритмов решения массовой задачи распределения работ по независимым исполнителям. Сформулированы условия применимости предложенного подхода.*

**Ключевые слова:** эвристический алгоритм, конечная массовая задача, индивидуальная задача, характеристика качества эвристического алгоритма, критерий качества эвристического алгоритма, статистическая оценка математического ожидания критерия качества, равновероятная генерация, распределение работ по независимым исполнителям

### Введение

Множественность эвристических алгоритмов, разрабатываемых для решения сложных задач определенного класса, обуславливает актуальность их сравнения и оценки с точки зрения используемых критериев эффективности. Основные применяемые для этой цели способы описаны в обзоре [1]. Однако реализация этих способов наталкивается на ряд трудностей, которые позволяет преодолеть предложенный в [2] подход к оценке качества эвристических алгоритмов, основанный на понятиях массовой задачи, индивидуальной задачи, параметризации массовой задачи, критерия качества эвристического алгоритма на индивидуальной задаче и, наконец, характеристики качества данного эвристического алгоритма на данной массовой задаче.

Интерпретация характеристики качества данного эвристического алгоритма на данной массовой задаче как функции распределения случайной величины — случайной реализации критерия качества данного алгоритма на индивидуальных задачах данной массовой задачи — позволяет сформулировать задачу ее статистического оценивания и решить эту задачу на основе равновероятной (для конечной массовой задачи) либо равномерной (для непрерывной и смешанной массовых задач) генерации точек области параметрического пространства данной массовой задачи, взаимно однозначно соответствующих индивидуальным задачам этой массовой задачи. Отметим, что наряду с "функциональной" характеристикой качества эвристического алгоритма на данной массовой задаче можно ввести в рассмотрение и более простые числовые харак-

теристики качества, например математическое ожидание упомянутой случайной величины.

Имеющийся опыт показывает, что построение равновероятных либо равномерных генераторов индивидуальных задач данной массовой задачи является, вообще говоря, технически непростой задачей.

Если построение требуемого генератора на основе применения способа интегральной вероятности не представляется возможным, то обычно используется некоторый вариант метода отбора-отказа [3, 4]. Недостаток такого подхода заключается в потенциально низкой эффективности такого генератора, определяемой как обратная величина математического ожидания числа отказов на одну успешную генерацию.

Другой возможный подход к построению требуемого генератора заключается в учете особенностей данной массовой задачи, в частности, возможности введения некоторой известным образом связанной с данной массовой задачей вспомогательной равновероятно распределенной случайной величины [5].

### Постановка задачи и получение статистической оценки характеристики качества

В настоящей работе изложен подход к сравнению и оценке качества эвристических алгоритмов решения конечной массовой задачи без построения равновероятного генератора входящих в нее индивидуальных задач, когда качество данного эвристического алгоритма на данной конечной массовой задаче характеризуется математическим ожиданием заданного критерия качества данного алгорит-

ма, определенного на множестве индивидуальных задач данной массовой задачи.

Пусть  $U$  — конечная массовая задача,  $|U| = K$ ,  $K$  — натуральное и пусть  $A_i, i = 1, \dots, K$ , — событие, заключающееся в извлечении из  $U$   $i$ -й индивидуальной задачи. При равновероятной генерации индивидуальных задач из  $U$  вероятность события  $A_i$  есть  $P(A_i) = \frac{1}{K}, i = 1, \dots, K$ .

Пусть  $H$  — эвристический алгоритм, предназначенный для решения массовой задачи  $U$ , пусть задан некоторый критерий качества алгоритма  $H$ , пусть  $a_i$  — значение данного критерия на  $i$ -й индивидуальной задаче из  $U$  и пусть имеется всего  $L \leq K$  различных значений этого критерия для индивидуальных задач из  $U$ . Введем в рассмотрение случайную величину  $X$  — будущее значение данного критерия качества для индивидуальной задачи из  $U$ , выбираемой путем ее равновероятной генерации из  $U$ .

Так как событие  $\{X = x_j\}$  эквивалентно событию

$$\bigcup_{k=1}^{I_j} A_{i_k}, \text{ где } x_j = a_{i_k}, 1 \leq i_k \leq K, 1 \leq k \leq I_j, \sum_{j=1}^L I_j = K,$$

то закон распределения случайной величины  $X$  можно записать в виде

$$P(X = x_j) = \frac{I_j}{K}, j = 1, \dots, L.$$

Математическое ожидание этой случайной величины  $M[X] = \sum_{j=1}^L x_j P(X = x_j) = \sum_{j=1}^L x_j \frac{I_j}{K} = \frac{1}{K} \sum_{j=1}^L x_j I_j$  может быть представлено как

$$M[X] = \frac{1}{K} \sum_{i=1}^K a_i, \quad (1)$$

что совпадает со средним значением выбранного критерия, вычисленным по всем индивидуальным задачам, входящим в  $U$ , и может рассматриваться как характеристика качества алгоритма  $H$  для массовой задачи  $U$ .

Пусть  $V$  — конечное множество,  $|V| = N \geq K$ ,  $N$  — натуральное, такое что:

- каждому элементу из  $V$  соответствует единственная задача из  $U$ ;
- каждой индивидуальной задаче из  $U$  соответствует хотя бы один элемент из  $V$ .

Пусть  $B_r, r = 1, \dots, N$ , — событие, заключающееся в извлечении  $r$ -го элемента из  $V$ , тогда при равновероятной генерации элементов из  $V$

$$P(B_r) = \frac{1}{N}, r = 1, \dots, N.$$

В силу свойств множества  $V$  для  $i = 1, \dots, K$  имеем

$$A_i = \bigcup_{s=1}^{m_i} B_{r_s}, 1 \leq r_s \leq N, 1 \leq s \leq m_i, \sum_{i=1}^K m_i = N, P(A_i) = \frac{m_i}{N}. \quad (2)$$

В условиях равновероятной генерации элементов из  $V$  поставим в соответствие событию  $A_i$  величину

$\frac{a_i}{m_i}, i = 1, \dots, K$ , т.е. определим некоторую случайную величину  $Y$ , и пусть всего имеется  $M \leq K$  различных значений этой случайной величины.

Так как событие  $\{Y = y_j\}$  эквивалентно событию

$$\bigcup_{k=1}^{q_j} A_{i_k}, \text{ где } y_j = \frac{a_{i_k}}{m_{i_k}}, 1 \leq i_k \leq K, 1 \leq k \leq q_j, \sum_{j=1}^M q_j = K,$$

то закон распределения случайной величины  $Y$  можно записать в виде

$$P(Y = y_j) = \sum_{k=1}^{q_j} \frac{m_{i_k}}{N}, j = 1, \dots, M.$$

Математическое ожидание случайной величины  $Y$

$$M[Y] = \sum_{j=1}^M y_j P(Y = y_j) = \sum_{j=1}^M \left( y_j \sum_{k=1}^{q_j} \frac{m_{i_k}}{N} \right) = \sum_{j=1}^M \sum_{k=1}^{q_j} y_j \frac{m_{i_k}}{N}.$$

Так как  $y_j = \frac{a_{i_k}}{m_{i_k}}$  для  $k = 1, \dots, q_j$ , то получим

$$M[Y] = \sum_{j=1}^M \sum_{k=1}^{q_j} \frac{a_{i_k}}{m_{i_k}} \frac{m_{i_k}}{N} = \frac{1}{N} \sum_{j=1}^M \sum_{k=1}^{q_j} a_{i_k} = \frac{1}{N} \sum_{i=1}^K a_i = \frac{K}{N} \frac{1}{K} \sum_{i=1}^K a_i.$$

Используя (1), получим  $M[Y] = \frac{K}{N} M[X]$ , откуда

$$M[X] = \frac{N}{K} M[Y]. \quad (3)$$

Пусть  $y^1, \dots, y^n$  —  $n$  независимых реализаций случайной величины  $Y$ . Результат  $l$ -й независимой реализации  $y^l$  получается следующим образом:

1) равновероятно генерируется элемент  $v \in V$ , т.е. реализуется некоторое событие  $B_r$ ;

2) определяется соответствующая  $v$  индивидуальная задача  $u \in U$ , т.е. определяется событие  $A_i$  такое, что  $B_r \subset A_i$ ;

3) определяется число  $m_i$  в соответствии с (2);

4) путем прогонки  $H$  на  $u$  определяется значение заданного критерия эффективности  $a_i$  на  $u$ ;

$$5) \text{ полагается } y^l = \frac{a_i}{m_i}.$$

Далее, по результатам  $n$  независимых реализаций случайной величины  $Y$  можно вычислить оценку ее математического ожидания

$$\widetilde{M}[Y] = \frac{1}{n} \sum_{l=1}^n y^l$$

и, следовательно, в соответствии с (3), можно определить оценку математического ожидания случайной величины  $X$  как

$$\widetilde{M}[X] = \frac{N}{K} \widetilde{M}[Y]. \quad (4)$$

Отметим, что полученная оценка сама по себе применима крайне редко, так как величины  $K$  и  $N$  в реальных задачах обычно неизвестны. Однако эта оценка может быть использована для сравнения эвристических алгоритмов.

### Сравнение эвристических алгоритмов

Рассмотрим теперь задачу сравнения двух эвристических алгоритмов по выбранному критерию качества.

Пусть  $H'$  — другой эвристический алгоритм для решения массовой задачи  $U$ . Определив для него аналогично предыдущему случайные величины  $X'$ ,  $Y'$ , получим

$$M[X'] = \frac{N}{K} M[Y'] \quad (5)$$

$$\widetilde{M}[X'] = \frac{N}{K} \widetilde{M}[Y']. \quad (6)$$

Поскольку в соответствии с (3) и (5)

$$\frac{M[X]}{M[X']} = \frac{M[Y]}{M[Y']},$$

$\widetilde{M}[Y]$  и  $\widetilde{M}[Y']$  являются состоятельными и несмещенными оценками соответственно для  $M[Y]$  и  $M[Y']$ ,

то из (4) и (6) следует, что величина  $\frac{\widetilde{M}[X]}{\widetilde{M}[X']} = \frac{\widetilde{M}[Y]}{\widetilde{M}[Y']}$

при достаточно больших объемах выборок дает адекватное представление о сравнительной эффективности алгоритмов  $H$  и  $H'$  для массовой задачи  $U$ ,

а оценки  $\widetilde{M}[Y]$  и  $\widetilde{M}[Y']$  могут быть использованы для построения статистических правил вынесения решения о сравнительной эффективности этих алгоритмов для массовой задачи  $U$ .

Отметим, что полученный результат позволяет установить не абсолютную эффективность эврис-

тического алгоритма в смысле заданного критерия качества, а только эффективность одного алгоритма относительно другого.

### Оценка качества эвристического алгоритма

В условиях равновероятной генерации элементов из  $V$  поставим в соответствие событию  $A_i$  величину  $\frac{1}{m_i}$ ,  $i = \overline{1, K}$ , т.е. определим некоторую случайную величину  $Z$ , и пусть всего имеется  $T \leq K$  различных значений этой случайной величины.

Так как событие  $\{Z = z_j\}$  эквивалентно событию

$$\bigcup_{k=1}^{s_j} A_{i_k}, \text{ где } z_j = \frac{1}{m_{i_k}}, 1 \leq i_k \leq K, 1 \leq k \leq s_j, \sum_{j=1}^T s_j = K,$$

то закон распределения случайной величины  $Z$  можно записать в виде

$$P(Z = z_j) = \sum_{k=1}^{s_j} \frac{m_{i_k}}{N}, j = 1, \dots, T.$$

Математическое ожидание случайной величины  $Z$

$$\begin{aligned} M[Z] &= \sum_{j=1}^T z_j P(Z = z_j) = \\ &= \sum_{j=1}^T \left( z_j \sum_{k=1}^{s_j} \frac{m_{i_k}}{N} \right) = \sum_{j=1}^T \sum_{k=1}^{s_j} z_j \frac{m_{i_k}}{N}. \end{aligned}$$

Так как  $\sum_{j=1}^T s_j = K$ ,  $z_j = \frac{1}{m_{i_k}}$  для  $k = 1, \dots, s_j$ , то по-

лучим

$$M[Z] = \sum_{j=1}^T \sum_{k=1}^{s_j} \frac{1}{m_{i_k}} \frac{m_{i_k}}{N} = \frac{1}{N} \sum_{j=1}^T \sum_{k=1}^{s_j} 1 = \frac{1}{N} \sum_{j=1}^T s_j = \frac{K}{N},$$

откуда

$$\frac{N}{K} = \frac{1}{M[Z]}. \quad (7)$$

Используя формулы (3), (7), получим

$$M[X] = \frac{M[Y]}{M[Z]}. \quad (8)$$

Пусть  $y^1, \dots, y^{n_1}$  —  $n_1$  независимых реализаций случайной величины  $Y$ , а  $z^1, \dots, z^{n_2}$  —  $n_2$  независимых реализаций случайной величины  $Z$ .

Результат  $l$ -й независимой реализации  $y^l$  получается следующим образом:

1) равновероятно генерируется элемент  $v \in V$ , т.е. реализуется некоторое событие  $B_j$ ;

2) определяется соответствующая  $v$  индивидуальная задача  $u \in U$ , т.е. определяется событие  $A_p$ , такое что  $B_r \subset A_p$ ;

3) определяется число  $m_i$  в соответствии с (2);

4) путем прогонки  $H$  на  $u$  определяется значение заданного критерия эффективности  $a_i$  на  $u$ ;

5) полагается  $y^l = \frac{a_i}{m_i}$ .

Результат  $l$ -й независимой реализации  $z^l$  получается следующим образом:

1) равномерно генерируется элемент  $v \in V$ , т.е. реализуется некоторое событие  $B_p$ ;

2) определяется соответствующая  $v$  индивидуальная задача  $u \in U$ , т.е. определяется событие  $A_p$ , такое, что  $B_r \subset A_p$ ;

3) определяется число  $m_i$  в соответствии с (2);

4) полагается  $y^l = \frac{1}{m_i}$ .

Далее, по результатам  $n_1$  независимых реализаций случайной величины  $Y$  можно вычислить оценку ее математического ожидания

$$\tilde{M}[Y] = \frac{1}{n_1} \sum_{l=1}^{n_1} y^l,$$

а по результатам  $n_2$  независимых реализаций случайной величины  $Z$  можно вычислить оценку ее математического ожидания

$$\tilde{M}[Z] = \frac{1}{n_2} \sum_{l=1}^{n_2} z^l,$$

и, следовательно, в соответствии с (8), можно определить оценку математического ожидания случайной величины  $X$  как

$$\tilde{M}[X] = \frac{\tilde{M}[Y]}{\tilde{M}[Z]}. \quad (9)$$

### Пример оценки качества эвристического алгоритма

В качестве примера использования предложенного подхода рассмотрим задачу распределения работ по независимым исполнителям в следующей постановке.

Имеется  $m$  исполнителей и  $n$  работ, каждая из которых может быть выполнена любым исполнителем за определенное время; будем считать, что  $j$ -я работа выполняется  $i$ -м исполнителем за время  $t_{ij}$ ,  $i = 1, \dots, m, j = 1, \dots, n$ , полагая  $t_{ij}$  натуральными числами из отрезка  $[1, T]$ . Все исполнители начинают работать одновременно и выполняют назначенные им работы последовательно. Пусть распределение  $w$  — вектор длины  $n$ , определяющий для каждой работы исполнителя, которым эта работа будет выполнена (т.е.  $w_j$  — номер исполнителя, выполняющего работу  $j$ ), и пусть  $f(w)$  — некоторый заданный на

множестве распределений критерий качества. Фиксировав параметры  $n, m, T$  и критерий  $f$ , получим конечную массовую задачу  $U$ , каждая индивидуальная задача которой может быть задана  $m \times n$ -матрицей  $(t_{ij})$ ,  $t_{ij} \in [1, T], i = 1, \dots, m, j = 1, \dots, n$ .

Отметим, что при этом одной и той же индивидуальной задаче из  $U$  может соответствовать несколько таких матриц, причем число этих матриц для различных индивидуальных задач также, вообще говоря, различно. Пусть, например,  $m = n = 2, T = 4$  при фиксированном критерии  $f$ . Тогда

- матрицы  $\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}, \begin{pmatrix} 2 & 2 \\ 2 & 2 \end{pmatrix}$  определяют каждая "свою" индивидуальную задачу и каждой из этих индивидуальных задач соответствует единственная матрица;

- матрицы  $\begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix}, \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$  соответствуют одной и той же индивидуальной задаче, определяемой любой из этих и только этих матриц;

- матрицы  $\begin{pmatrix} 2 & 1 \\ 1 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 2 \\ 1 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 1 \\ 2 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix}$  соответствуют одной и той же индивидуальной задаче, определяемой любой из этих и только этих матриц;

- матрицы  $\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}, \begin{pmatrix} 3 & 4 \\ 1 & 2 \end{pmatrix}, \begin{pmatrix} 2 & 1 \\ 4 & 3 \end{pmatrix}, \begin{pmatrix} 4 & 3 \\ 2 & 1 \end{pmatrix}$  соответствуют одной и той же индивидуальной задаче, определяемой любой из этих и только этих матриц,

а матрицы  $\begin{pmatrix} 1 & 3 \\ 2 & 4 \end{pmatrix}, \begin{pmatrix} 2 & 4 \\ 1 & 3 \end{pmatrix}, \begin{pmatrix} 3 & 1 \\ 4 & 2 \end{pmatrix}, \begin{pmatrix} 4 & 2 \\ 3 & 1 \end{pmatrix}$  соответствуют другой индивидуальной задаче, определяемой любой из этих и только этих матриц.

Для данной массовой задачи  $U$  (с фиксированными  $n, m, T, f$ ) определим множество  $V$  всех возможных объектов, каждый из которых является  $m \times n$ -матрицей с элементами, представляющими собой пары вида  $(i, t_i)$ ,  $i = 1, \dots, m \cdot n, t_i \in [1, T], t_1 \leq t_2 \leq \dots \leq t_{m \cdot n}, t_i$  — натуральные, среди этих чисел могут быть одинаковые. Отметим следующее.

1. Каждый объект  $v$  множества  $V$  порождает матрицу  $(t_{ij})$ , получающуюся из  $v$  удалением первых элементов пар  $(i, t_i)$ , причем различные объекты множества  $V$  могут породить одинаковые матрицы  $(t_{ij})$ . Например, при  $n = m = 2, T = 4$  и фиксированном  $f$

объекты  $\begin{pmatrix} (1,1) & (3,2) \\ (4,3) & (2,1) \end{pmatrix}$  и  $\begin{pmatrix} (2,1) & (3,2) \\ (4,3) & (1,1) \end{pmatrix}$  порождают

одну и ту же матрицу  $\begin{pmatrix} 1 & 2 \\ 3 & 1 \end{pmatrix}$ . Вообще, если среди  $mn$

чисел  $t_i$  имеется одинаковых  $l_1, \dots, l_L, \sum_{i=1}^L l_i = mn$ , то



одна и та же матрица  $(t_{ij})$  порождается  $m' = \prod_{i=1}^L l_i!$

различными объектами множества  $V$ .

2. Любая из порожденных таким образом матриц  $(t_{ij})$  задает индивидуальную задачу  $u \in U$ , причем данной индивидуальной задаче  $u$  соответствует  $m''$  различных матриц, получаемых из матрицы  $(t_{ij})$  всевозможными перестановками строк и столбцов. Заметим, что в случае, когда все элементы матрицы  $(t_{ij})$  различны,  $m'' = m! \cdot n!$ ; в общем случае число  $m''$  можно определить, например, с помощью любого подходящего переборного алгоритма.

3. Различные наборы чисел  $t_1 \leq \dots \leq t_{mn}$ ,  $t_i \in [1, T]$ ,  $i = 1, \dots, mn$ , определяют непересекающиеся множества индивидуальных задач массовой задачи  $U$ .

4. По определению множества  $V$  с учетом пп. 1—3 заключаем, что каждой  $i$ -й индивидуальной задаче  $u \in U$  соответствует

$$m(u) = m' \cdot m'' \quad (10)$$

элементов множества  $V$ , причем способ определения  $m(u)$  задан.

Определим следующую процедуру генерации объектов из множества  $V$ .

1. В соответствии с [5] генерируем равновероятно набор чисел  $t_1 \leq \dots \leq t_{mn}$ ,  $t_i \in [1, T]$ ,  $i = 1, \dots, mn$ . Событие  $C$ , заключающееся в получении любого такого набора, имеет вероятность [6]

$$P(C) = \frac{(mn)!}{T(T-1)\dots(T+mn-1)}.$$

2. Полученному набору чисел  $t_1 \leq \dots \leq t_{mn}$  поставим в соответствие последовательность пар  $(1, t_1), \dots, (mn, t_{mn})$ .

3. Из множества натуральных чисел  $\{1, \dots, mn\}$  образуем равновероятно упорядоченную выборку без возвращения  $(g_1, \dots, g_{mn})$  объема  $mn$ , поставив каждый элемент  $g_k$  этой выборки в соответствие паре  $(g_k, t_{g_k})$ .

4. Паре  $(g_k, t_{g_k})$  поставим в соответствие пару индексов

$$i = \left\lceil \frac{k}{n} \right\rceil,$$

$$j = k - in,$$

где  $\lceil x \rceil = \begin{cases} x, & x - \text{целое,} \\ \lceil x \rceil + 1 & \text{в противном случае,} \end{cases}$   $\lceil x \rceil$  — целая часть  $x$ .

В результате получим  $m \times n$ -матрицу, элемент которой, стоящий на пересечении  $i$ -й строки и  $j$ -го столбца, есть пара  $(g_k, t_g)$ , т.е. полученный в результате исполнения процедуры объект есть объект из множества  $V$ . Событие  $D$ , заключающееся в получении любого данного объекта из  $V$  при любом

данном наборе чисел  $t_1 \leq \dots \leq t_{mn}$ , полученных на шаге 1 описанной процедуры, имеет вероятность

$$P(D) = \frac{1}{(mn)!}.$$

Так как событие  $B_r$ , заключающееся в получении любого  $r$ -го объекта множества  $V$ , представимо в виде  $B_r = C \cap D$ , то при независимости событий  $C$ ,  $D$  (достигаемой при независимой генерации объектов на шагах 1 и 3 процедуры) получим

$$P(B_r) = P(C)P(D) = \frac{1}{T(T+1)\dots(T+mn-1)},$$

т.е. определенная выше процедура обеспечивает равновероятную генерацию объектов  $v \in V$ .

Определив далее, в соответствии с вышеизложенным, случайные величины  $X$ ,  $Y$ ,  $Z$  и применив описанную выше процедуру получения независимых реализаций  $Y$ ,  $Z$ , где величины  $m_i$  вычисляются в соответствии с формулой (10), получим оценку (9) рассматриваемой характеристики качества тестируемого эвристического алгоритма  $H$  решения данной массовой задачи распределения работ по независимым исполнителям.

## Заключение

Предложенный способ позволяет проводить сравнение и оценку эвристических алгоритмов решения данной конечной массовой задачи  $U$  по заданному критерию качества в следующих условиях:

- эффективный способ равновероятной генерации индивидуальных задач из  $U$  отсутствует;
- имеется эффективный способ равновероятной генерации элементов для некоторого вспомогательного конечного множества  $V$ ;
- для любого элемента  $v \in V$  можно определить соответствующую задачу  $u \in U$  и определить число  $m$  элементов из  $V$ , которым задача  $u$  соответствует.

## Список литературы

1. Мухачева Э. А. Задачи оптимального раскроя и родственная проблематика после 1971 г. / В кн.: Канторович Л. В., Залгаллер В. А. Рациональный раскрой промышленных материалов. 3-е изд., испр. и доп. СПб: Невский диалект, 2012. С. 250—292.
2. Орехов Э. Ю., Орехов Ю. В. Об оценке качества эвристического алгоритма на конечной массовой задаче // Информационные технологии. 2011. № 7. С. 28—33.
3. Ермаков С. М. Метод Монте-Карло и смежные вопросы. М.: Наука, 1971. 328 с.
4. Соболев И. М. Численные методы Монте-Карло. М.: Наука, 1973. 311 с.
5. Orekhov E. Yu., Orekhov Yu. V. A method to compare the heuristic algorithms for the finite problems // Proc. of the 8<sup>th</sup> International Workshop on Computer Science and Informational Technologies. Karlsruhe, Germany. 2006. Vol. 2. P. 91—94.
6. Orekhov E. Yu., Orekhov Yu. V. Equiprobable Generation of the Integer One-Dimensional Cutting-Packing Problem Instances // Proc. of the 5<sup>th</sup> International Workshop on Computer Science and Informational Technologies, Ufa, Russia, 2003. Vol. 2. P. 41—42.

## Comparing and Estimating Quality of Heuristic Algorithms on a Finite Problem under Non-Equiprobable Generation of Test Instances

*Efficiency of heuristic algorithms can be estimated by introducing a "functional" quality characteristic for a given heuristic algorithm on a given finite problem. This characteristic is a distribution function of a random variable that takes on a set of possible values of the algorithm's quality criterion on problem instances. This fact allows statistical estimation of the quality characteristic by a certain random generation of points within a region of parametric space for the given problem. These points correspond one-to-one to the problem instances. However, numerical quality characteristics are more common in practice, such as the expectation of a random variable. Equiprobable generation of problem instances ensures the appropriate statistical estimate for the quality characteristic, but the task of developing equiprobable generators arises here. In this paper we suggest an approach to comparing and estimating quality of heuristic algorithms based on the expectation of the random variable mentioned above, in cases when equiprobable generation of problem instances is difficult or impossible. This approach involves a secondary set of objects associated to the set of problem instances, while equiprobable generation of elements of the secondary set is quite easy. We give an example of applying our approach to statistical estimation of quality for heuristic algorithms that solve a problem of scheduling jobs between unrelated parallel machines. We also define the conditions of application for the suggested approach.*

**Keywords:** heuristic algorithm, finite problem, problem instance, quality characteristic of a heuristic algorithm, quality criterion of a heuristic algorithm, statistical estimate for the expectation of quality criterion, equiprobable generation, scheduling jobs between unrelated parallel machines

### References

1. **Mukhachyova E. A.** Zadachi optimal'nogo raskroya i rodstvennaya problematika posle 1971 g. V kn.: L. V. Kantorovich, V. A. Zalgaller. *Racional'nyj raskroj promyshlennykh materialov*. 3-e izd., ispr. i dop. SPb: Nevskij dialekt, 2012, pp. 250–292.
2. **Orekhov E. Yu., Orekhov Yu. V.** Ob ocenke kachestva ehvristicheskogo algoritma na konechnoj massovoj zadache, *Informacionnye tekhnologii*, 2011, no. 7, pp. 28–33.
3. **Ermakov S. M.** *Metod Monte-Karlo i smezhnye voprosy*, Moscow, Nauka, 1971, 328 p.
4. **Sobol I. M.** *Chislennyye metody Monte-Karlo*, Moscow, Nauka, 1973, 311 p.
5. **Orekhov E. Yu., Orekhov Yu. V.** A method to compare the heuristic algorithms for the finite problems, *Proc. of the 8<sup>th</sup> International Workshop on Computer Science and Informational Technologies*, Karlsruhe, Germany, 2006, vol. 2, pp. 91–94.
6. **Orekhov E. Yu., Orekhov Yu. V.** Equiprobable Generation of the Integer One-Dimensional Cutting-Packing Problem Instances, *Proc. of the 5<sup>th</sup> International Workshop on Computer Science and Informational Technologies*, 2003, vol. 2, pp. 41–42.



### ГАЛУШКИН АЛЕКСАНДР ИВАНОВИЧ

17.02.1940—08.11.2016

Неожиданно для всех нас ушел из жизни очень яркий, энергичный, многогранный человек, который объединял своей плодотворной деятельностью большое число людей у нас в стране и за рубежом.

Доктор технических наук, профессор, заслуженный деятель науки России, лауреат Премии Правительства России, Александр Иванович Галушкин родился 17 февраля 1940 г., в 1963 г. закончил МВТУ им. Баумана. В 1966 г. защитил диссертацию на соискание ученой степени кандидата технических наук, а в 1974 г. — ученой степени доктора технических наук. А. И. Галушкин — автор более 400 научных работ, в том числе 25 монографий.

Работы А. И. Галушкина по теории нейронных сетей являются общепризнанными в России и мире. Они представлены в монографиях: Многослойные системы распознавания образов (1974 г.), Теория нейронных сетей (2000 г.), *Neural network theory*, Springer (2007 г.), *Нейронные сети: основы теории* (2010 г.).

А. И. Галушкин вел активную научно-педагогическую деятельность, которую начал в МИЭМ, продолжил в МФТИ и в Национальном исследовательском университете "Высшая школа экономики".

В последние годы жизни А. И. Галушкин — начальник Лаборатории разработки систем интеллектуального управления промышленными комплексами ФГАНУ ЦИТиС и начальник Центра нейросетевых технологий ИнтерЭВМ.

В лице А. И. Галушкина наука потеряла большого ученого, который внес важный вклад в современную науку и плодотворную деятельность многих организаций.

А. И. Галушкин с момента создания журнала "Информационные технологии" был активным членом редколлегии, а в последние годы — главным редактором журнала в журнале "Нейросетевые технологии".

Выражаем искреннее соболезнование семье и коллегам А. И. Галушкина в связи с тяжелой утратой.

*Редколлегия и редакция журнала  
"Информационные технологии"*

# ЦИФРОВАЯ ОБРАБОТКА СИГНАЛОВ И ИЗОБРАЖЕНИЙ DIGITAL PROCESSING OF SIGNALS AND IMAGES

УДК 004.312.2; 621.3.049.771.14

С. В. Гаврилов<sup>1</sup>, д-р техн. наук, зав. отделом, С. И. Гуров<sup>2</sup>, канд. физ.-мат. наук, доц., e-mail: sgur@cs.msu.ru,  
Т. Д. Жукова<sup>1</sup>, инженер-исследователь, Д. И. Рыжова<sup>1</sup>, мл. науч. сотр., e-mail: ryzhova\_d@ippm.ru

<sup>1</sup>Институт проблем проектирования в микроэлектронике РАН, г. Москва

<sup>2</sup>Московский государственный университет имени М. В. Ломоносова

## Применение теории кодирования для повышения помехозащищенности комбинационных схем

*Исследованы проблемы автоматизации проектирования, направленные на повышение помехозащищенности комбинационных интегральных схем на этапе логического проектирования. Проанализированы различные методы помехоустойчивого кодирования применительно к повышению помехозащищенности комбинационных интегральных схем. Реализованы методы кодирования на основе выбора варианта коммутирования выходов дубликата основной схемы с учетом логических корреляций между выходами схемы. Одним из существенных отличий задачи повышения помехоустойчивости комбинационных схем, по сравнению с кодированием в сетях передачи данных, является дополнительная степень свободы, связанная с реализацией схемы кодирования. Применение методов оптимизации позволяет сокращать размер схемы кодирования, по сравнению с мажоритарным подходом, за счет выбранного порядка коммутаций.*

**Ключевые слова:** комбинационные схемы, проектирование, помехозащищенность, САПР

### Введение

Среди известных подходов к проектированию помехоустойчивых интегральных схем (ИС) широко используется дублирование с выбором результата по мажоритарному принципу [1]. Однако в последнее время все больший интерес вызывают попытки применения для решения данной задачи методов помехоустойчивого кодирования, которые аналогичны по своей сути подходам к защите от помех потоков информации при ее передаче по линиям связи [2].

Традиционным методом повышения отказоустойчивости при информационных сбоях в каналах связи является помехоустойчивое кодирование. Для парирования информационных сбоев, возникших в результате сбоев аппаратных, возможно использование следующих подходов [3].

1. Аппаратное резервирование с дальнейшим применением мажорирования — процедуры коррекции искаженной в результате сбоя информации путем сравнения результатов, полученных параллельным путем, и выдача на выход наиболее совпадающих результатов.

2. Информационное резервирование с применением результатов теории кодов, обнаруживающих и/или исправляющих ошибки.

Поскольку информационные сбои ИС обычно возникают в результате воздействия заряженных частиц или иных помех, в данном случае как ана-

логи понятий "отказоустойчивость" и "надежность" используют термины "помехоустойчивость" и "помехозащищенность" (ИС) соответственно.

Одним из существенных отличий задачи кодирования комбинационных схем по сравнению с кодированием в сетях передачи данных является дополнительная степень свободы, связанная с реализацией схемы кодирования, а именно возможность выбора порядка коммутации в целях сокращения аппаратных затрат на схему кодирования. В данной работе предлагается математический аппарат, позволяющий эффективно оценивать потенциальные возможности оптимизации схемы кодирования за счет выбора того или иного порядка коммутации для схемы кодирования.

В качестве математического аппарата для оценки потенциальных возможностей оптимизации схемы кодирования предлагается использовать формализмы для анализа вероятностей и корреляции логических уровней сигналов в комбинационных схемах.

В разд. 1 данной работы представлен анализ существующих подходов к кодированию информации для решения задачи повышения помехозащищенности комбинационных схем, в разд. 2 приведено описание методов помехоустойчивого кодирования для повышения помехозащищенности комбинационных интегральных схем, а также рассмотрены методы построения оптимальных схем комму-

тирования выходов схемы кодирования на основе анализа вероятностей распространения парных корреляций в схеме.

## 1. Анализ существующих методов блокового кодирования

### 1.1. Общие положения

Опишем сначала основные положения теории кодов, исправляющих ошибки. Помехоустойчивость кодирования достигается введением избыточности в код. Часто возникновение ошибок может быть описано наиболее простыми моделями со случайным некоррелированным информационным потоком, в котором некоторые биты случайно и независимо друг от друга могут оказаться инвертированными. В случае с передачей информации под влиянием помех такие модели называют *двоичными симметричными каналами*, в которых предполагается, что нет добавлений/стираний битов и замены  $0 \rightarrow 1$  и  $1 \rightarrow 0$  равновероятны. При этом могут ставиться задачи автоматического обнаружения и/или исправления ошибок. Отметим, что каждый конкретный корректирующий код не гарантирует исправления любой комбинации ошибок.

Одним из возможных подходов к решению проблемы является разбиение потока информации на *сообщения* — непересекающиеся блоки фиксированной длины  $k$ . Каждый блок можно кодировать независимо от других — *блоковое кодирование* или в зависимости от предыдущих — *сверточное кодирование* [4–6]. В результате вместо сообщений передают *словые* длины  $n > k$  каждое. Естественно требовать построения кода минимальной длины, позволяющего восстановить сообщение, содержащее не более  $r$  ошибок (инверсий).

Например, при кодировании каждого  $k$ -битного сообщения можно добавить один бит, содержащий 0 или 1, так, чтобы число единиц в коде было четным; полученный код называется *кодом с проверкой на четность*. При таком кодировании обнаруживается искажение любого четного числа символов.

Если кодовое слово длины  $n = k + t$  содержит в себе  $k$  бит исходного сообщения (*информационные биты*) и дополнительно еще  $t$  *проверочных бит*, то говорят о *разделимом блоковом кодировании*. В *неразделимых кодах* выделить информационные и проверочные биты невозможно. Увеличение  $t$  ведет, вообще говоря, к увеличению кодового расстояния  $d$  и, следовательно, к увеличению числа ошибок, которые может исправить код. Минимальное расстояние  $d$  между словами кода называется *кодовым расстоянием*. Известно, что у кода, исправляющего  $r$  ошибок, кодовое расстояние должно быть не менее  $2r + 1$ . Разделимый блоковый код описывают тройкой параметров  $(n, k, d)$  или парой  $(n, k)$ .

Определение кодового расстояния  $d$  произвольного кода — сложная задача. Поэтому при создании помехоустойчивых кодов на первый план выходит проблема построения кодов с заданным кодовым расстоянием. Она решается при использовании БЧХ-кодов [7]. Величину  $R = k/n$  называют *скоростью*, а  $t/n = 1 - R$  — *избыточностью кода*. На сегодняшний день уже построены БЧХ-коды с практически значимыми параметрами.

*Блоковое кодирование* — это взаимно-однозначное отображение множества сообщений  $S$  всех векторов из  $2^k$  во множество кодовых слов  $C$  (некоторых векторов из  $2^n$ ) — всегда может быть осуществлено с использованием таблицы размера  $2^k \times n$ . Однако табличное кодирование весьма неэффективно: значения  $n$  и  $k$  на практике могут достигать десятков и сотен тысяч. Известны две конструкции так называемых *совершенных кодов*, плотно заполняющих шарами радиуса  $r$  с центрами в кодовых словах весь куб  $2^n$ : коды Хемминга ( $2^q - 1, 2^q - q - 1, 3$ ), где  $q$  — натуральное число, и код Голея ( $23, 12, 7$ ), исправляющие 1 и 3 ошибки соответственно.

*Декодирование* состоит в определении сообщения по кодовому слову. Декодирование кодов обычно значительно сложнее кодирования. При передаче по каналу с шумом кодовое слово  $v \in 2^n$  превращается в принятое слово  $w = v + e$ , где  $e$  — *вектор ошибок*.

Декодирование  $(n, k, d)$ -кода основано на разбиении единичного куба  $2^n$  на  $k$  областей, содержащих шары радиуса  $r = \left\lceil \frac{d-1}{2} \right\rceil$  с центрами в кодовых словах, в предположении, что при передаче произошло не более  $r$  ошибок. Тогда восстановление переданного сообщения  $v$  состоит в определении ближайшего к полученному  $w$  в метрике Хемминга, другими словами, в нахождении центра соответствующего шара. Для этого надо перебрать все  $2^k$  строк в таблице  $2^k \times n$  кодовых слов. Далее необходимо провести восстановление по словам  $v$  исходного сообщения  $u$  путем удаления проверочных бит. В общем случае, когда неизвестны их позиции, это потребует использования таблицы размера  $2^k \times k$ .

Из сказанного следует, что декодирование блокового  $(n, k)$ -кода общего вида — очень ресурсоемкий процесс, и поэтому использование таких кодов возможно лишь при небольших значениях  $n$  и  $k$ . Однако, приняв ряд дополнительных ограничений на множество кодовых слов, можно перейти от экспоненциальных требований по памяти для хранения кода и по сложности алгоритмов кодирования/декодирования к линейным требованиям по  $n$  и  $k$ . Эти ограничения приводят к использованию блоковых кодов специального вида — групповых, а из групповых кодов — к циклическим.

## 1.2. Групповые (линейные) коды

Большая часть теории блочного кодирования построена на *линейных кодах*, образующих векторное подпространство координатного пространства  $2^n$ . В линейных кодах сумма по mod2 любых кодовых слов — также кодовое слово. Линейные коды позволяют реализовывать эффективные алгоритмы кодирования/декодирования, и в двоичном случае их называют групповыми, так как они образуют группу относительно операции  $\oplus$ . Линейные  $(n, k)$ -коды могут быть заданы матрицами *порождающей*  $G_{n \times k}$  или *проверочной*  $H_{m \times n}$ . Для них выполняются соотношения  $v = Gu$ ,  $Hv = 0$  для любого кодового слова  $v$ , и невыполнение последнего равенства свидетельствует о наличии ошибки.

На практике удобно использовать *систематическое кодирование* [8], при котором  $k$  бит сообщения копируются в фиксированные позиции кодового слова, а затем вычисляются остальные  $m = n - k$  проверочных битов. Такая возможность основана на том, что порождающая и проверочная матрицы определены с точностью до эквивалентных преобразований столбцов и строк соответственно, что эквивалентно переходу к другому базису пространства кодовых слов  $C$  и ортогонального ему  $C^\perp$ . Фиксирование позиций информационных битов задает порождающую и проверочную матрицы однозначно. При этом второй этап декодирования (удаление проверочных битов) становится тривиальным.

Декодирование групповых кодов проводят с использованием *синдромов* — векторов  $s = Hw \in 2^m$ . Вычисление вектора ошибок  $e$  сводится к решению системы линейных алгебраических уравнений (СЛАУ)  $He = s$ , которое ищут в виде суммы частного  $\hat{e}$  решения данной системы и общего  $Gu$  решения соответствующей однородной системы  $e = \hat{e} + Gu$ . После нахождения частного решения  $\hat{e}$  все возможные кодовые слова  $u_1, \dots, u_{2^k}$  входного вектора дадут  $2^k$  вариантов вектора  $e_i = \hat{e} + Gu_i$ . Решение с наименьшим хэмминговым весом дает искомый вектор ошибок.

Для каждого из  $2^m$  синдромов необходимо перебирать  $2^k$  решений очередной СЛАУ, т. е. алгоритм декодирования линейного кода в общем случае имеет экспоненциальную трудоемкость и по памяти, и по числу операций. Получив вектор ошибок  $e$ , декодирование осуществляют по правилу  $v = w + e$ .

Таким образом, кодирование групповыми кодами осуществляется умножением вектора-сообщения на порождающую матрицу. Декодирование также значительно упрощается по сравнению с общим случаем: используются легко вычисляемые синдромы при элементарном этапе удаления проверочных бит в случае систематического кодирования. Однако в общем случае требуется перебрать  $2^k$  решений СЛАУ, т. е. процесс декодирования остается все еще достаточно трудоемким.

Если число единиц во всех комбинациях кода будет постоянным, то такой код будет *кодом с постоянным весом*. Это блочные неразделимые коды. Обнаружение ошибок при таком кодировании сводится к определению веса принятого слова, и если он отличается от заданного, то считается, что произошла ошибка. Код обнаруживает ошибки нечетной кратности и часть ошибок четной кратности. Не обнаруживаются ошибки смещения, при которых несколько единиц превращаются в нули и столько же нулей — в единицы.

## 1.3. Циклические коды

Циклические коды — подкласс линейных [9]. Код  $C$  называется *циклическим (сдвиговым)* (Cyclic Redundancy Code, CRC), если он инвариантен относительно циклических сдвигов, т. е. для любого  $0 \leq s \leq n - 1$  справедливо следующее:

$$\begin{aligned} &(\alpha_0, \dots, \alpha_{n-1}) \in C \Rightarrow \\ &\Rightarrow (\alpha_s, \alpha_{s+1}, \dots, \alpha_{n-1}, \alpha_0, \dots, \alpha_{s-1}) \in C. \end{aligned}$$

В теории конечных полей показывается, что фактор-кольцо многочленов  $\mathbb{F}_2[x]/(x^n - 1)$ , рассматриваемое как векторное пространство размерности  $n$  над полем  $\mathbb{F}_2$ , имеет базис  $\{1, x, \dots, x^{n-1}\}$ . Если  $\varphi(x)$  — неприводимый многочлен из  $\mathbb{F}_2[x]$ , делящий  $x^n - 1$ , то порожденный им идеал  $(\varphi(x))$  — циклическое подпространство в  $\mathbb{F}_2[x]/(x^n - 1)$  и в нем циклический сдвиг равносильен умножению элемента на  $x$ . Поэтому для построения циклического кода выбирают степень  $n$ , порождающий многочлен кода — некоторый делитель  $g(x)$  биннома  $x^n - 1$ , и в кольце  $\mathbb{F}_2[x]/(x^n - 1)$  образуют идеал  $(g(x))$ . При удачном выборе  $g(x)$  коэффициенты многочленов из данного идеала будут давать код с малой избыточностью при большом кодовом расстоянии. Однако известны только несколько конструкций циклических кодов с хорошими параметрами, а в общем случае определение кодового расстояния циклического кода чрезвычайно сложно.

При использовании циклических кодов удобно пользоваться представлением векторов сообщения  $u$  и кодового слова  $v$  в виде полиномов  $u(x)$ ,  $v(x) \in \mathbb{F}_2[x]$ :

$$\begin{aligned} u &= [u_0, u_1, \dots, u_{k-1}]^T \leftrightarrow \\ \leftrightarrow u(x) &= u_0 + u_1x + \dots + u_{k-1}x^{k-1}, \\ v &= [v_0, v_1, \dots, v_{n-1}]^T \leftrightarrow \\ \leftrightarrow v(x) &= v_0 + v_1x + \dots + v_{n-1}x^{n-1}. \end{aligned}$$

Различают *систематическое* и *несистематическое кодирование* циклическими кодами, которое приводит к разделимому и неразделимому кодированию. Несистематическое кодирование осуществляется путем умножения кодируемого вектора на  $g(x) - v(x) = u(x)g(x)$ , а систематическое кодирование — "дописыванием" к кодируемому слову остат-

ка от деления  $x^{n-k}u(x)$  на  $g(x) - v(x) = x^m u(x) + r(x)$ , где  $r(x) \equiv_{g(x)} x^m u(x)$  (мы рассматриваем простейший вариант систематического кодирования, когда полином  $v(x)$  имеет  $k$  коэффициентов полинома  $u(x)$  в  $k$  крайних правых позициях, т. е. при старших степенях  $x$ ).

Синдромом принятого полинома  $w(x)$ , закодированного циклическим  $(n, k)$ -кодом с порождающим полиномом  $g(x)$  (и, возможно, содержащим ошибки), называют остаток  $s(x)$  от деления  $w(x)$  на  $g(x)$ :  $s(x) \equiv_{g(x)} w(x)$ . Ясно, что если  $s(x) \equiv 0$ , то  $w(x)$  — кодовое слово.

Декодирование циклического кода проходит по общей схеме декодирования линейного кода: вычисляется синдром  $s(x)$  принятого слова  $w(x)$ , ищутся решения системы  $e(x) = s(x) + g(x)u(x)$  для всех  $2^k$  возможных полиномов  $u(x)$  степени  $k - 1$ , определяется полином ошибок как решение с минимальным числом ненулевых слагаемых и, наконец, восстанавливается переданное сообщение  $u(x) = w(x) + e(x)$ .

Циклические коды общего вида могут иметь произвольную длину  $n$ , но в отличие от линейного кода общего вида его параметры  $m$  и, следовательно,  $k = n - m$  (число информационных битов) уже не произвольны:  $g(x)|(x^n - 1)$ . При использовании циклических кодов вместо матричных умножений и решения СЛАУ используются более простые операции умножения и деления с остатком полиномов, легко реализуемые на регистрах сдвига с обратными связями. Однако общий алгоритм декодирования по-прежнему имеет экспоненциальную сложность по  $k$ . Существуют и альтернативные методы декодирования циклических кодов общего вида, но и они не имеют удовлетворительных характеристик.

Обнаружение ошибок с помощью циклического кода обеспечивается тем, что в качестве разрешенных комбинаций выбираются такие, которые делятся без остатка на некоторый заранее выбранный порождающий полином. Если принятая комбинация

содержит искаженные символы, то такое деление осуществляется с остатком и при этом формируется сигнал, свидетельствующий об ошибке. Большим преимуществом циклических кодов является простота построения кодирующих и декодирующих устройств, которые по своей структуре представляют регистры сдвига с обратными связями.

## 2. Применение помехоустойчивого кодирования для повышения помехозащищенности комбинационных интегральных схем

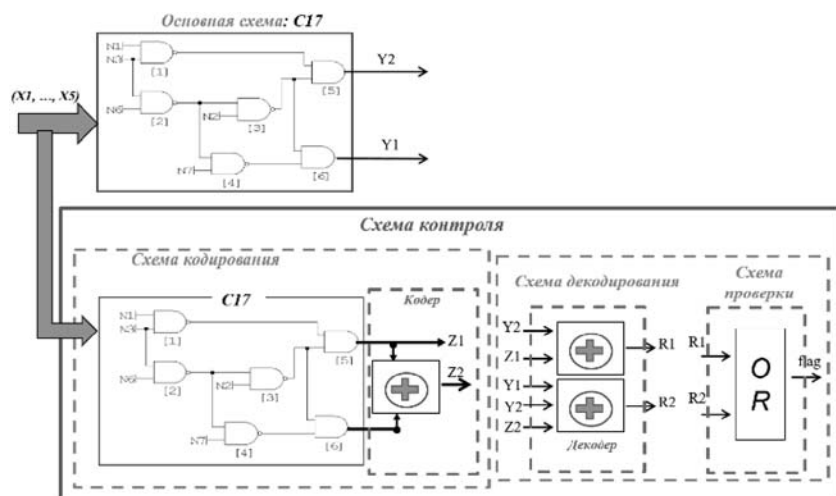
### 2.1. Предлагаемый метод повышения помехозащищенности

Для повышения помехозащищенности комбинационных схем был выбран метод с применением циклических кодов, который позволяет не только обнаружить наличие ошибок в переданном сообщении, но и исправить определенное число ошибок. Число ошибок, которое можно исправить, определяется свойствами образующего многочлена [10—12]. Данный метод не приводит к неоправданной избыточности, обеспечивает возможность оптимального сочетания требований к минимизации аппаратных затрат и достижение требуемого уровня отказоустойчивости [13, 14]. Схема предлагаемого подхода представлена на рисунке.

Для кодирования сообщений применяется операция умножения на образующий многочлен, при этом вектор исходного сообщения  $a_{k-1}, \dots, a_0$  представляется в виде многочлена  $a_{k-1}x^{k-1} + a_{k-2}x^{k-2} + \dots + a_0$ , где  $k$  — длина сообщения. Для декодирования сообщения используется операция деления переданного сообщения на образующий многочлен. Остаток от деления принятого слова на образующий многочлен называется синдромом. Если ошибок в переданном сообщении нет, то синдром равен нулю. Вектор ошибок можно найти по вычисленному синдрому.

В задаче передачи сообщений для реализации операций умножения и деления многочленов используют сдвиговые регистры, так как сигнал передается в виде последовательности нулей и единиц. В случае комбинационных схем сигналы на выходы приходят параллельно, поэтому здесь необходимо использовать в качестве схем кодирования и декодирования логические схемы.

Принципиальное отличие задачи кодирования данных для передачи через канал связи от задачи повышения помехоустойчивости комбинационных схем состоит в следующем. Так как в задаче повышения помехоустойчивости сама комбинационная схема подвержена ошибкам, схема кодирования реализу-



Предлагаемый подход

ется за счет дублирования исходной схемы с последующей оптимизацией совместно со схемой кодирования. Для синтеза схемы кодирования используется операция умножения многочлена выходного сообщения комбинационной схемы  $a_{k-1}x^{k-1} + \dots + a_0$  на образующий многочлен  $g^r(x)$ , где  $r$  — степень образующего многочлена.

Ключевая проблема задачи передачи сообщений — минимизация площади или размера схем кодирования и декодирования. Одним из путей оптимизации схемы является построение логических функций на основе применения аппарата BDD. Предлагаемое при этом использование синтеза булевых функций в конечных полях Галуа на основе редуцированных диаграмм двоичных решений (ROBDD) позволяет снять существующее в настоящее время ограничение на размерность проектируемых комбинационных схем (число входов и выходов). Использование предлагаемой методики обеспечивает управляемость и предсказуемость процесса проектирования схем при достижении оптимального сочетания заданных требований по отказоустойчивости и минимизации структурных затрат. Критерием оценки площади при применении BDD может служить число вершин графа BDD. Оценку задержек можно рассчитать исходя из длины цепи от входа до выхода при предположении мультиплексорной реализации BDD.

Проведенные исследования показали, что на качество результата в терминах занимаемой площади существенное влияние оказывает не только перепорядочение входов, как в случае стандартной BDD, но и порядок коммутации выходов в схеме кодирования.

Полный перебор различных вариантов коммутаций требует анализа  $k!$  вариантов и, следовательно, невозможен для большого числа выходов  $k$ . Выбор вариантов коммутаций будет более эффективен, если заранее оценить потенциальные возможности оптимизации схемы кодирования до проведения самого процесса создания и оптимизации схемы.

Для выбора оптимального варианта коммутации предлагается использовать оценочную функцию, вычисленную на основе расчета взаимных корреляций между выходами [15]. Примеры синтеза схемы контроля на основе предложенного подхода были приведены ранее в работах [10—14]. Наибольший эффект от оптимизации схемы кодирования достигается при условии вхождения в одну формулу схемы кодирования выходов схемы, имеющих взаимные корреляции. Для всех выходов дублирующей схемы применяют предложенные методы анализа логических корреляций в цифровой схеме для получения оценочных функций.

Используя полученные оценочные функции, выбирают порядок коммутации выходов дублирующей схемы на основе анализа вероятностей распространения парных корреляций.

## 2.2. Методы построения оптимальных схем коммутирования входов схемы кодирования с учетом логики работы основной схемы

За счет учета дискретных корреляций можно сократить размер схемы кодирования. Однако дискретные корреляции не описывают полную картину возможных корреляций сигналов в схеме. Для полного учета парных корреляций в качестве альтернативы предлагается модифицировать аппарат анализа парных корреляций сигналов, который использовался при оптимизации мощности и анализе стрессовых состояний транзисторов [16].

Пусть каждый сигнал  $a$  в комбинационной схеме (либо первичный вход, либо выход любого вентиля) характеризуется с помощью величины  $p(a = v)$ , равной отношению времени нахождения сигнала в состоянии ( $a = v$ ) к общему времени моделирования. Для описания корреляции между двумя сигналами введем коэффициент корреляции сигналов по аналогии с работой [17]:

$$SC_{ij}^{ab} = \frac{p(a = i \& b = j)}{p(a = i)p(b = j)}, \quad i, j = 0, 1. \quad (1)$$

Для выбора оптимального варианта коммутации предлагается использовать оценочную функцию, которая вычисляется согласно следующей формуле:

$$f(a, b) = \begin{cases} SC, & SC \leq 1; \\ 1/SC, & SC > 1. \end{cases}$$

Наибольшего эффекта оптимизации можно добиться, сопоставляя минимум оценочной функции с парами связанных сигналов в кодере.

В целях устранения экспоненциальной сложности алгоритма можно воспользоваться предположением о том, что существенны только парные корреляции. Иначе говоря, будем пренебрегать корреляцией любых двух сигналов к третьему и т.д. В этом случае вероятность сложного события может быть приближенно вычислена по формуле

$$p\left(\prod_{k=1}^n (a_k = i_k)\right) = \prod_{k=1}^n p(a_k = i_k) \prod_{1 \leq k < l \leq n} SC_{i_k i_l}^{a_k a_l}. \quad (2)$$

Поскольку

$$p(a = i \& b = j) = p(a = i/b = j)p(b = j) = p(b = j/a = i)p(a = i),$$

где  $p(X/Y)$  обозначает вероятность события  $X$  при условии  $Y$ , то

$$SC_{ij}^{ab} = \frac{p(a = i/b = j)}{p(a = i)} = \frac{p(b = j/a = i)}{p(b = j)}.$$

Следовательно, мы имеем следующие соотношения для четырех коэффициентов  $SC_{ab}$ :

$$\sum_{i=0,1} SC_{ij}^{ab} p(a=i) = 1; \quad \sum_{j=0,1} SC_{ij}^{ab} p(b=j) = 1; j=0,1, i=0,1. \quad (3)$$

Матрица системы (3) имеет ранг 3, поэтому, зная  $SC_{00}^{ab}$ , можно вычислить три других коэффициента по следующим формулам:

$$SC_{01}^{ab} = \frac{1 - SC_{00}^{ab} p_b}{1 - p_b}, \quad SC_{10}^{ab} = \frac{1 - SC_{00}^{ab} p_a}{1 - p_a}, \quad SC_{11}^{ab} = \frac{1 - SC_{10}^{ab} p_b}{1 - p_b}. \quad (4)$$

При вычислении длительности нуля для выхода вентиля должны быть известны вероятности нулевого значения сигналов для всех входов вентиля и коэффициенты  $SC$  для всех пар входов вентиля. Все возможные входные векторы вентиля могут быть разделены на два множества:

- 1)  $V_0$  — множество входных векторов вентиля, для которых выход вентиля равен 0;
- 2)  $V_1$  — множество входных векторов вентиля, для которых выход вентиля равен 1.

Для сложного совместного события, обеспечивающего значение на выходе, равное нулю ( $y=0$ ), можно записать:

$$(y=0) = \sum_{I \in V_0} \prod_{k=1}^n (x_k = i_k), \quad (5)$$

где  $n$  — число входов вентиля;  $x_k$  — сигналы на входах вентиля (переменные);  $I = (i_1, \dots, i_n)$  — входной вектор вентиля. Векторы под дизъюнкцией в выражении (5) являются взаимно исключающими, поэтому

$$p(y=0) = \sum_{I \in V_0} p\left(\prod_{k=1}^n (x_k = i_k)\right).$$

Используя (2) и пренебрегая корреляциями более высоких порядков, получаем следующую формулу для расчета времени нахождения транзистора в стрессовом состоянии:

$$p(y=0) = \sum_{I \in V_0} \prod_{k=1}^n \left( p(x_k = i_k) \prod_{k < l \leq n} SC_{i_k i_l}^{x_k x_l} \right).$$

Для корректной обработки каждого вентиля также необходимо распространять по схеме коэффициенты корреляции  $SC$ . Пусть  $z$  — любой сигнал с известной вероятностью и известными  $SC$  с каждым входом вентиля. Необходимо решить задачу распространения  $SC$  сигнала  $z$  с входов вентиля на

его выход. В соответствии с определением (1) можно записать

$$SC_{00}^{zy} = \frac{p(z=0 \& y=0)}{p(z=0)p(y=0)}.$$

Рассматривая ( $z=0 \& y=0$ ) как сложное событие и вычисляя его вероятность, получаем следующую формулу для распространения  $SC$  через вентиль:

$$SC_{00}^{zy} = \frac{\sum_{I \in V_0} \prod_{k=1}^n \left( p(x_k = i_k) SC_{0i_k}^{zx_k} \prod_{k < l \leq n} SC_{i_k i_l}^{x_k x_l} \right)}{p(y=0)}.$$

Остальные коэффициенты  $SC$  могут быть вычислены по формулам (4).

Формула учета парных корреляций для выбора варианта коммутации может быть модифицирована следующим образом:

$$S^*(a, b) = \sum_{v_a=0,1} \sum_{v_b=0,1} p(a=v_a \& b=v_b) = \sum_{v_a=0,1} \sum_{v_b=0,1} p(a=v_a) p(b=v_b) SC_{v_a v_b}^{ab}.$$

В результате применения предложенных методов получили результаты, которые показывают выигрыш порядка 10 % (в единичных случаях около 58 %) по площади для схем из набора ISCAS85 при построении схемы кодирования на основе полинома второй ( $x^2 + x + 1$ ) и третьей степени ( $x^3 + x + 1$ ) по сравнению с мажорированием.

## Заключение

В работе выполнен анализ существующих методов кодирования для решения задачи повышения помехозащищенности комбинационных схем. Предложен подход к оптимизации схем кодирования за счет выбора варианта коммутирования выходов дубликата основной схемы на основе результатов анализа вероятностей логических корреляций. Применение разработанных методов на основе выбора порядка коммутаций позволяет добиться минимизации размера схемы кодирования примерно на 10 % по сравнению с мажоритарным подходом.

## Список литературы

1. Черкесов Г. Н. Надежность аппаратно-программных комплексов: учеб. пособие. СПб.: Питер, 2004. 479 с.
2. Морелос-Сарагоса Р. Искусство помехоустойчивого кодирования. Методы, алгоритмы, применение / пер. с англ. В. Б. Афанасьева. М.: Техносфера, 2006. 320 с.
3. Матвеевский В. Р. Надежность технических систем: учеб. пособие. М.: Изд. МИЭМ, 2002. 113 с.
4. Вернер М. Основы кодирования: учебник для вузов. М.: Техносфера, 2004. 288 с.
5. Блейхут Р. Теория и практика кодов, контролирующих ошибки. М.: Книга по требованию, 2013. 566 с.
6. Van Lint J. H. Introduction to Coding Theory. Springer Science & Business Media, 2012. 234 p.



7. Poolakparambil M., Mathew J. BCH Code Based Multiple Bit Error Correction in Finite Field Multiplier Circuits // ISQED, 2011. P. 1–6.

8. MacWilliams F. J., Sloane N. J. A. The Theory of Error—Correcting Codes. North—Holland Publishing Company, 1977. 762 p.

9. Касами Т., Токура Н., Ивадари Е. Теория кодирования / пер. с яп. А. В. Кузнецова. М.: Мир, 1978. 576 с.

10. Гаврилов С. В., Иванова Г. А., Рыжова Д. И., Соловьев А. Н., Стемповский А. Л. Методы синтеза помехозащищенных комбинационных блоков // Информационные технологии. 2015. Т. 21, № 11. С. 821–826.

11. Гаврилов С. В., Иванова Г. А., Рыжова Д. И., Стемповский А. Л. Методы повышения надежности комбинационных микросистем на основе мультиинтервального анализа быстрого действия // Системы высокой доступности. 2015. № 4. С. 69–76.

12. Соловьев А. Н., Стемповский А. Л. Методы повышения отказоустойчивости работы устройства управления микросистемы за счет введения структурной избыточности // Информационные технологии. 2014. № 10. С. 17–22.

13. Гаврилов С. В., Иванова Г. А., Соловьев А. Н., Стемповский А. Л. Оптимизация схем кодирования на основе выбора варианта коммутаций с учетом логических корреляций между выходами комбинационной схемы // Известия ЮФУ. Технические науки. 2015. № 6 (167), С. 255–262.

14. Гаврилов С. В., Иванова Г. А., Соловьев А. Н., Щелокоев А. Н. Учет логических корреляций между выходами комбинационной схемы при коммутации с входами схемы кодирования // Труды Международного конгресса по интеллектуальным системам и информационным технологиям — 2015, IS & IT'15. С. 192–197.

15. Гаврилов С. В. Методы анализа логических корреляций для САПР цифровых КМОП СБИС. М.: Техносфера, 2011. 136 с.

16. Стемповский А. Л., Глебов А. Л., Гаврилов С. В., Гудкова О. Н. Вероятности напряженного состояния транзисторов для временного анализа с учетом электротемпературной нестабильности // Информационные технологии. 2009. № 7. С. 32–38.

17. Marculescu R., Marculescu D., Pedram M. Switching Activity Analysis Considering Spatiotemporal Correlations // Proc. ICCAD-1994. P. 294–299.

S. V. Gavrilo<sup>1</sup>, DSc, Head of Department, S. I. Gurov<sup>2</sup>, PhD, Assistant Professor,  
T. D. Zhukova<sup>1</sup>, Research Engineer, D. I. Ryzhova<sup>1</sup>, Junior Research Scientist, ryzhova\_d@ippm.ru  
<sup>1</sup>Institute for Design Problems in Microelectronics of Russian Academy of Sciences (IPPM RAS)  
<sup>2</sup>Lomonosov Moscow State University

## Application of Coding Theory to Improve the Noise Immunity of Combinational Circuits

*The work is devoted to solving problems of design automation to improve the noise immunity of combinational integrated circuits at the logical level design. The different methods of error-correcting coding for improving noise immunity of combinational integrated circuits are analyzed. The coding methods with a selection of duplicate circuit outputs commutation based on the logical correlations between the circuit outputs are implemented. One of the significant difference of the noise immunity of combinational circuits, in comparison with coding in data networks, is an additional degree of freedom associated with the implementation of the coding scheme. The application of effective optimization methods reduces the size of coding circuit by selection of the commutation order, compared to the majority approach.*

**Keywords:** combinational circuits, design, noise tolerance, CAD

### References

1. Cherkesov G. N. *Nadezhnost' apparatno-programmnykh kompleksov*, uchebnoe posobie, Saint Petersburg: Piter, 2004, 479 p.

2. Morelos-Saragosa R. *Iskusstvo pomehoustojchivogo kodirovaniya. Metody, algoritmy, primeneniye*, per. s angl. V. B. Afanas'eva, Moscow: Tehnosfera, 2006, 320 p.

3. Matveevskij V. R. *Nadezhnost' tehniceskikh system, uchebnoe posobie*, Moscow, Moskovskij gosudarstvennyj institut jelektroniki i matematiki, Moscow, 2002, 113 p.

4. Verner M. *Osnovy kodirovaniya, uchebnik dlja VUZov*, Moscow, Tehnosfera, 2004, 288 p.

5. Blejhut R. *Teorija i praktika kodov, kontrolirujushhij oshibki*, Moscow, Kniga po trebovaniju, 2013, 566 p.

6. Lint J. H. *Introduction to Coding Theory*, Springer Science & Business Media, 2012, 234 p.

7. Poolakparambil M., Mathew J. BCH Code Based Multiple Bit Error Correction in Finite Field Multiplier Circuits, *Proceedings of the 12th International Symposium on Quality Electronic Design (ISQED 2011)*, pp. 1–6.

8. MacWilliams F. J., Sloane N. J. A. The Theory of Error—Correcting Codes, North—Holland Publishing Company, 1977, 762 p.

9. Kasami T., Tokura N., Iwadari E. *Teorija kodirovaniya*, per. s jap. A. V. Kuznecova, Moscow, Mir, 1978, 576 p.

10. Gavrilo S. V., Ivanova G. A., Ryzhova D. I., Solov'ev A. N., Stempkovskij A. L. Metody sinteza pomehozashishhennykh kombinacionnykh blokov, *Informacionnye tehnologii*, 2015, vol. 21, no. 11, pp. 821–826.

11. Gavrilo S. V., Ivanova G. A., Ryzhova D. I., Stempkovskij A. L. Metody povysheniya nadezhnosti kombinacionnykh mikroelektronnykh shem na osnove mult'interval'nogo analiza bystrodejstvija, *Sistemy vysokoj dostupnosti*, 2015, no. 4, pp. 69–76.

12. Solov'ev A. N., Stempkovskij A. L. Metody povysheniya otkazoustojchivosti raboty ustrojstva upravlenija mikrosistemy za schet vvedeniya strukturnoj izbytochnosti, *Informacionnye tehnologii*, 2014, no. 10, pp. 17–22.

13. Gavrilo S. V., Ivanova G. A., Solov'ev A. N., Stempkovskij A. L. Optimizacija shem kodirovaniya na osnove vybora varianta kommutacij s uchedom logicheskikh korrelacij mezhdru vyhodami kombinacionnoj shemy, *Izvestija JuFU. Tehnicieskie nauki*, 2015, no. 6 (167), pp. 255–262.

14. Gavrilo S. V., Ivanova G. A., Solov'ev A. N., Shhelokov A. N. Uchet logicheskikh korrelacij mezhdru vyhodami kombinacionnoj shemy pri kommutacii s vhodami shemy kodirovaniya, *Tруды Международного конгресса по интеллектуальным системам и информационным технологиям*, "IS & IT'15", 2015, pp. 192–197.

15. Gavrilo S. V. *Metody analiza logicheskikh korrelacij dlja SAPR cifrovyyh KMOP SBIS*, Moscow, Tehnosfera, 2011, 136 p.

16. Stempkovskij A. L., Glebov A. L., Gavrilo S. V., Gudkova O. N. Veroyatnosti napjzhenno go sostojanija tranzistorov dlja vremennogo analiza s uchedom jelektrotemperaturnoj nestabil'nosti, *Informacionnye tehnologii*, 2009, no. 7, pp. 32–38.

17. Marculescu R., Marculescu D., Pedram M. Switching Activity Analysis Considering Spatiotemporal Correlations, *Proceedings of the International Conference on Computer-Aided Design*, 1994, pp. 294–299.

# ПРОГРАММНАЯ ИНЖЕНЕРИЯ SOFTWARE ENGINEERING

УДК 004:378

**Е. Н. Решетова**, e-mail: [ereshetova@hse.ru](mailto:ereshetova@hse.ru),  
преподаватель Департамента программной инженерии факультета компьютерных наук  
Национальный исследовательский университет "Высшая школа экономики" (НИУ ВШЭ)

## Универсальные компетенции студента образовательного направления "Программная инженерия"

*Рассмотрена одна из актуальных проблем образования на современном этапе — определение универсальных компетенций технических специалистов. Представлены некоторые из современных подходов к этому вопросу и предложена модель универсальных компетенций как обязательная интеллектуальная основа профессионализма программного инженера.*

**Ключевые слова:** программная инженерия, универсальные компетенции, SWECOM

### Введение

Инженерия программного обеспечения — это область компьютерной науки, которая занимается построением больших и/или сложных программных систем. Обычно такие системы существуют долгие годы, развиваясь от версии к версии, претерпевая на своем жизненном цикле множество изменений, таких как устранение ошибок, улучшение существующих версий, добавление новых или удаление устаревших возможностей, адаптация для работы в новой среде. Трудоемкость, долговременность, огромная важность таких высокотехнологичных проектов в настоящее время практически полностью переводят индустрию разработки коммерческого программного обеспечения на проекты с групповой (командной) разработкой.

Статистика утверждает [1]: только треть проектов завершается в срок, не превышает запланированный бюджет и реализует все требуемые функции и возможности. Около половины проектов завершаются с опозданием (среднее превышение сроков составляет 120 %), расходы превышают запланированный бюджет (превышение затрат 100 %), требуемые функции не реализованы в полном объеме. До 20 % проектов полностью проваливаются и

аннулируются до завершения. Примечательным является тот факт, что данная тенденция, как показано в табл. 1, практически сохраняет свои позиции с 2011 г.

Анализируя причины провалов программных проектов, экспертное сообщество утверждает: эта тенденция является результатом недостаточного внимания к человеческому фактору. Теоретически каждый понимает, что люди важны. Почти каждый согласен, что люди важнее, чем средства, методы и процесс. Но большинство продолжает действовать так, как если бы это было неправдой. Линейные руководители считают процесс альфой и омегой разработки: выдвигают инструментальные средства на роль волшебных палочек, собирают разношерстные методы, называют результат методологией и требуют, чтобы сотни программистов читали о ней, посещали курсы по ней, отполировывали знания путем зубрежки и упражнений и затем применяли в ответственных проектах. И все это от имени средств, методов, процессов, стоящих над людьми.

Раньше всех о важности человеческого фактора сказали Бучер [2]: "Больше всего надежность программного обеспечения зависит от отбора и мотивирования персонала, а также от управления им" и Руби [3]: "В конечном счете, главным фактором производительности является потенциал отдельного исполнителя". Эта точка зрения выражена в классической книге "Peopleware" [4], целиком посвященной роли людей в создании программного обеспечения. В ней можно, например, прочитать что "Серьезные проблемы в нашей работе имеют не столько технологическую, сколько социологическую природу" и даже что первостепенное внимание к технологическим вопросам относится к

Таблица 1

Современный анализ результатов программных проектов, выполненных в 2011–2015 гг., с точки зрения удовлетворения сроков и бюджету [1]

Проекты	2011	2012	2013	2014	2015
УСПЕШНЫЕ	29 %	27 %	31 %	28 %	29 %
СПОРНЫЕ	49 %	56 %	50 %	55 %	52 %
ПРОВАЛЬНЫЕ	22 %	17 %	19 %	17 %	19 %

"миражам высоких технологий". Как самую выдающуюся иллюстрацию значимости людей можно назвать обложку классической книги Барри Бозма "Экономика программной инженерии" [5]. На ней представлена гистограмма факторов, способствующих созданию хорошего программного продукта — самый высокий столбец гистограммы представляет профессионализм людей, выполняющих работу. Эта диаграмма свидетельствует, что люди значат намного больше, чем любые средства, методы, языки и процессы, которые они применяют. "Важнейшим человеческим фактором, определяющим производство и качество программных продуктов, являются люди — специалисты с их уровнем профессиональной квалификации, а также с многообразием психологических характеров, знаний, опыта, стимулов и потребностей", — говорит классик отечественной программной инженерии Липаев Владимир Васильевич в своей книге "Человеческие факторы в программной инженерии: рекомендации и требования к профессиональной квалификации специалистов" [6]. "Человеческие факторы — личная профессиональная квалификация, психологические характеристики специалистов и организация крупного коллектива в основном определяют качество и трудоемкость при производстве программных продуктов" [6] — это утверждение вышло из многочисленных научных исследований и докладов за прошедшие годы. Эту точку зрения поддерживает и современный работодатель, активно демонстрируя повсеместный интерес к гибкой (agile) разработке, в Манифесте которой первым принципом провозглашается "Люди и их взаимодействие важнее, чем процессы и инструменты" [7].

Что же сегодня обязательно для признания профессионализма специалиста?

Сегодня и каждому участнику команды и тем более менеджеру программного проекта требуются навыки, далеко выходящие за пределы познаний о принципах программирования. Компетенции продукта, проекта и процесса необходимы, но не достаточны для успешной реализации проекта. Лучшие на планете аналитики процессов и профессионалы в деле постановки требований, проектировщики, кодировщики менеджеры проектов не могут успешно реализовать проект без учета компетенций персонала, без учета феноменов, возникающих при взаимодействии людей. Разработчики и клиенты — это не абстрактные проблемы, а люди с обычными потребностями, любящие, чтобы ими руководили, их инструктировали, с ними советовались, их отбирали с должным достоинством и мотивировали соответственно их потребностям. Программная инженерия — это не только области знаний, в которых изучаются процессы разработки продуктов, рассматривается управление процессами, определяющими выполнение технологических карт и управление проектами. Персонал, рассматриваемый зачастую как более гибкая сторона уп-

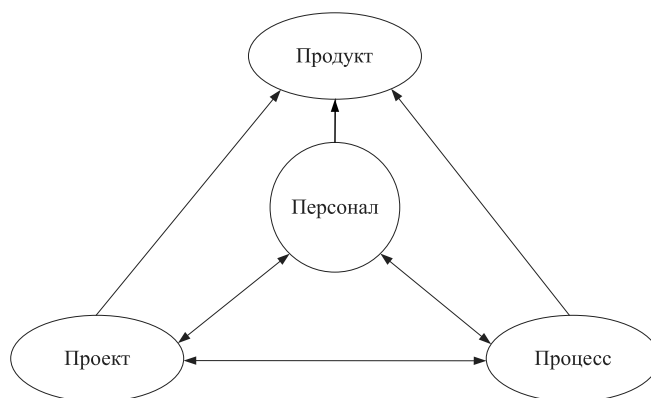


Рис. 1. Четыре "П" в разработке программного обеспечения

равления программными проектами, является фактически наиболее важной частью структуры "Разработка программного обеспечения: персонал, проект, процесс, продукт" (рис. 1). Организация может соответствовать пятому уровню зрелости по СММІ [8], но успешное выполнение проекта будет невозможным без квалифицированной команды. Исследователям может принадлежать превосходное изобретение, но продукт никогда не будет выпущен без организации взаимодействия, эффективного представления и встреч, навыков ведения переговоров и управления изменениями.

### Компетентностный подход — актуальная реальность российского высшего образования

Актуальной задачей высшего профессионального образования на современном этапе становится практическая реализация компетентностного подхода в рамках каждой образовательной программы и каждой учебной дисциплины, что позволит разрешить существующие противоречия между требованиями к подготовке специалистов со стороны государства, общества, работодателей и имеющимися образовательными результатами. Актуальность изменения содержания образования на основе компетентностного подхода наиболее содержательно и мощно стала обсуждаться экспертным сообществом высшего профессионального образования начиная с 2000 г. Аргументация в пользу разработки и внедрения в образовательную практику идей компетентностного подхода началась с анализа нового для России социально-экономического контекста. Социально-экономические изменения 90-х годов в России привели к необходимости модернизации многих социальных институтов и в первую очередь — системы образования, которая напрямую связана с экономическими процессами через подготовку производительных сил. Квалификация как результат профессиональной подготовки, подразумевающая наличие у выпускника определенных профессиональных знаний, умений и навыков, дополнилась компетентностным подходом — центрирован-

ном на студенте, на результатах обучения. Это модель планирования и реализации образовательных программ, альтернативная сложившейся, которая фокусируется на том, что смогут делать студенты в конце дисциплины (модуля) или программы в целом. Его разработкой занимались В. И. Байденко, В. А. Болотов, В. В. Грачев, О. А. Жукова, И. А. Зимняя, А. А. Орлов, В. В. Сериков и многие другие [9–12]. Взгляды, исследования и конкретные разработки этих ученых сильнейшим образом отразились на состоянии современного высшего образования, компетенции, и результаты образования стали рассматриваться как главные целевые установки в реализации ФГОС ВПО, как интегрирующие начала "модели" выпускника.

Именно компетентностный подход стал связующим мостом между образовательным процессом и реальными запросами общества и работодателей: были рассмотрены требования, которые предъявляются по отношению к работнику. При изучении ситуации на рынке труда стало очевидно: современное образование должно формировать профессиональный универсализм — способность человека менять способы и сферы своей деятельности. Хороший сотрудник должен быть не только профессионально подготовлен в своей предметной области, но также от него сейчас требуются умение работать в команде, принимать самостоятельные решения, он должен проявлять инициативу, быть способным к инновациям, быть готовым к перегрузкам, быть психологически устойчивым, устойчивым к стрессовым ситуациям и пр.

Таким образом, под универсальными компетенциями понимают надпрофессиональные умения, деловые навыки и способности личности, позволяющие ей творчески самореализовываться, социально взаимодействовать и адаптироваться к изменяющимся условиям.

В области исследования и применения универсальных компетенций в последнее время произошли серьезные изменения в сторону действительного признания их важности и наполнения актуальной смысловой сущностью. Подтвердим это фактами.

**Факт 1.** *Универсальные компетенции признаны необходимым условием успешной работы современного специалиста вне зависимости от отраслевой специализации.*

В "Атласе новых профессий" [13], как результате проекта, инициированного Агентством стратегических инициатив (АСИ) и Московской школой управления СКОЛКОВО (масштабного исследования "Форсайт компетенций 2030"), в котором приняли участие более 2500 российских и иностранных экспертов, представлены надпрофессиональные навыки и компетенции, которые считаются необходимым условием успешной работы специалиста, вне зависимости от отраслевой специализации. Владение ими позволяет человеку повысить эффективность профессиональной деятельности в

своей отрасли, а также дает возможность переходить из одной отрасли в другую, сохраняя свою востребованность. Среди таких навыков: системное мышление, клиенториентированность, междотраслевая коммуникация, знание нескольких языков и разных типов культур, навыки управления проектами и людьми — от локальных групп до сообществ, навыки программирования, умение работать в режиме многозадачности и постоянно меняющихся условиях, навыки бережливого производства, навыки художественного творчества.

**Факт 2.** *Модели универсальных компетенций меняются, адаптируясь к вызовам мира.*

Экономические спады и подъемы, начавшиеся с 2007 г. и продолжающиеся по сей день, так повлияли на среду, что она фундаментально отличается сегодня от того, с чем мы имели дело 10 лет назад: в последнее десятилетие многие отрасли стремительно меняются, границ больше не существует. Мнение о нынешних условиях как об очень неспокойных подтверждается исследованием IBM, проведенным среди 1500 первых руководителей [14]. Эти директора определили своей главной проблемой растущую сложность окружающей среды, причем большинство из них указали, что их организации не оснащены должным образом для преодоления этой сложности, в том числе и не "оснащены" людьми новой формации. Эта тема неоднократно поднималась многими респондентами данного исследования. Некоторые из них, описывая новые условия, в которых приходится работать, использовали армейскую фразу V. U. C. A [15]:

- Volatile (изменчивость): изменения происходят быстро и масштабно.
- Uncertain (неопределенность): будущее невозможно предсказать сколько-нибудь точно.
- Complex (сложность): проблемы-вызовы усложняются многими факторами; редко бывает, что ситуация имеет одну причину и одно решение.
- Ambiguous (неоднозначность): невозможно однозначно трактовать события и их потенциальное влияние.

Итак, адаптивные организации уже нуждаются в качественно новых сотрудниках, а значит, высшее профессиональное образование нуждается в обновлении модели универсальных компетенций персонала. Помимо традиционных компетенций, таких как логическое, системное и структурированное мышление, коммуникационные навыки, навыки работы в команде, лидерство и пр., в VUCA-мире необходимы VUCA-компетенции: V — Vision (видение), U — Understanding (понимание), C — Creativity/Clarity (креативность/ясность), A — Agility (решимость).

### **Универсальные компетенции специалиста по направлению "Программная инженерия"**

Обозначенные выше факты самым непосредственным образом влияют на систему подготовки

специалистов ИКТ-индустрии, характеризующейся чрезвычайно высокими темпами развития и влиянием на темпы развития экономики, конкурентоспособность и безопасность страны.

Области знаний, относящиеся к нетехнической подготовке специалиста по направлению "Программная инженерия", можно выделить, изучая документы и международные стандарты [16–20].

Кроме того, одним из главных источников для выявления требований к современному ИКТ-специалисту служит модель компетенций програм-



Рис. 2. Элементы SWECOM

Таблица 2

Когнитивные навыки SWECOM

Когнитивные навыки	Примеры
<b>Рассуждение</b> обеспечивает основу для принятия решений логическим и эффективным образом	Индуктивное рассуждение Дедуктивное рассуждение Эвристическое рассуждение Использование абстракций Иерархическое и ассоциативное рассуждение
<b>Аналитические навыки</b> , связанные с методами, которые включают в себя сбор данных, организацию и агрегирование данных, а также анализ и оценку для того, чтобы делать выводы и принимать решения	Применение принципов измерения Статистический анализ и анализ данных Анализ причин Идентификация рисков и анализ, а также анализ воздействия
<b>Решение проблем</b> связано с различными методами, которые используют рассуждение, аналитические методы и приоритизацию информации, чтобы решить проблемы	Разделяй и властвуй Пошаговое уточнение Нисходящий подход Восходящий подход Аналогии и повторное использование Шаблоны и распознавание шаблонов Итерационные и поэтапные подходы
<b>Иновация</b> подразумевает навыки, используемые для создания моделей и абстракций, которые поддерживают анализ и решение проблемы	Мозговой штурм Разработка прототипа Моделирование и симуляция

мною инженерии (SWECOM) [20], охватывающая области навыков, навыки и мероприятия для лиц, которые разрабатывают и поддерживают программное обеспечение.

Данная модель компетенции (рис. 2) добавляет к растущей совокупности знаний то, что характеризует профессию программного инженера и специалистов программной инженерии.

В контексте данной статьи нас будут интересовать не технические навыки, представленные моделью, а так называемая интеллектуальная основа профессии программного инженера: когнитивные навыки, а также поведенческие атрибуты и навыки (в нашей статье названные *универсальными компетенциями*).

Универсальные компетенции (когнитивные и поведенческие навыки) применяются при выполнении практически всей деятельности SWECOM в областях технической квалификации. Уровни универсальных компетенций не включены в SWECOM, но они становятся все более важными на более высоких уровнях технической компетенции, так как объем и сложность работ увеличивается, а деятельность расширяется. Как показано в табл. 2, когнитивные навыки SWECOM включают в себя четыре классификации. Они не являются независимыми: навыки, перечисленные в классификации, пересекаются и объединяются, чтобы поддержать эффективные когнитивные компетенции. Более того, список в табл. 3, предназначенный для иллюстрации когнитивных навыков программных инженеров, не является исчерпывающим.

Поведенческие атрибуты и навыки не являются уникальными для программной инженерии, но позволяют программным инженерам эффективно добиваться желаемых результатов. Некоторые важные для программных инженеров поведенческие атрибуты и навыки приведены в табл. 2.

Анализируя рассмотренную выше информацию, можно предложить модель универсальных компетенций специалистов по направлению "Программная инженерия" (рис. 3). Суть проста: в проекте мы создаем программный продукт и при этом анализируем, моделируем, разрабатываем решения, применяем экспертизу и руководствуемся коммерческой выгодой. В процессе создания программного продукта мы взаимодействуем, устно и письменно коммуницируя, работая в команде, ориентируясь на задачи клиента. При этом каждый из нас исполняет, обеспечивая качество и соблюдение стандартов, планируя и организуя как собственную деятельность, так и деятельность команды. Успешную деятельность поддерживают следующие личностные качества (как основа персональной эффективности в этой профессиональной среде и мире): инициативность, открытость новому/гибкость, нацеленность на результат, развитие.

Модель компетенций представляет собой структурированный набор компетенций и составляющих

Таблица 3

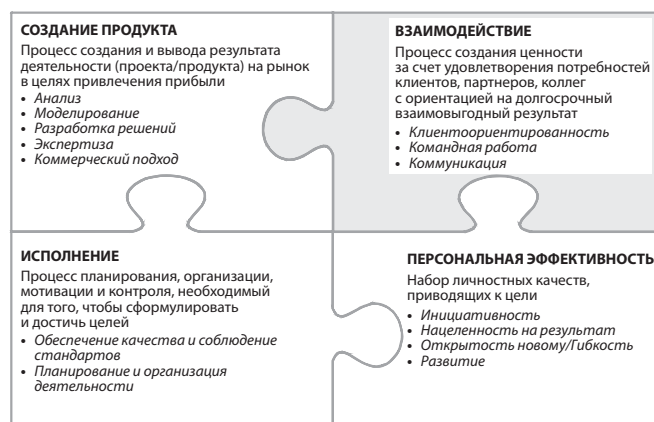
**Поведенческие атрибуты и навыки SWECOM**

Атрибуты	Примеры
Способность	Возможность эффективно выполнять задачи программной инженерии. Способность — это не то же самое, что знания или навыки, она указывает на возможность (интуитивно или эрудированно) умело применять знания
Инициатива	Энтузиазм начала работы и последующего выполнения задач программной инженерии
Энтузиазм	Выражение интереса в выполнении рабочей задачи
Трудовая этика	Быть надежным, приобретать новые навыки и быть готовым выполнять рабочие задачи
Готовность	Начало работы с задачей при запросе и умелое исполнение задачи, даже если человек не в восторге от данных работ
Надежность	Доказанная в течение долгого времени, путем демонстрации этичного поведения, честности, целостности, надежности в индивидуальных решениях и действиях
Культурная чувствительность	Осознание и соответственное поведение в различных стилях коммуникации, социальных взаимодействиях, соблюдение дресс-кода и правил общего поведения, основанного на этнической, религиозной, гендерной ориентации и других поведенческих характеристиках
Коммуникативные способности	Выражение понятий, методов, мыслей и идей в устной и письменной формах в ясной и сжатой манере во время общения с членами команды, менеджерами, участниками проекта и другими; включает в себя эффективное слушание
Навыки работы в команде	Работа с энтузиазмом и готовностью вместе с другими членами команды, сотрудничая на общих задачах
Лидерские качества	Эффективное сообщение видения, стратегии, методов или техник, которые затем принимаются и разделяются членами команды, менеджерами, участниками проекта, и другими

их компонентов. Каждая компетенция из модели детализируется собственным набором компонентов, которые описывают проявления компетенций, являясь требованиями к наличию какого-либо элемента компетенции (табл. 4).

Наравне с навыками, касающимися продукта, процесса и проекта, изучаемыми на протяжении всей образовательной программы специалистов, должно происходить системное обучение универсальным компетенциям. Универсальные компетенции применимы ко всем элементам и на всех уровнях областей технических навыков, умений и деятельностей. Они используются на каждой фазе выполняемого проекта, жизненного цикла разработки программного обеспечения и в ходе постоянной поддержки качества. Трудно четко привязать каждый навык к определенной фазе процесса выполнения проекта. Зачастую во время осуществления любой фазы жизненного цикла требуются многие компетенции. Причем одна компетенция может поддерживать другие компетенции. Приобретение и развитие универсальных компетенций сквозной нитью должно происходить через все учебные дисциплины и профессиональную практику студентов. Безусловно, учебный план может непосредственно влиять лишь на некоторые приобретаемые компетенции студента (например, на командную работу, аналитические навыки, коммуникацию), в то время как приобретение других компетенций (таких как инициативность, открытость новому, уверенность) является предметом более тонкого влияния всего образовательного процесса, корпоративной культуры вуза, факультета, департамента.

На факультете компьютерных наук в Департаменте программной инженерии показанная модель универсальных компетенций как обязательная интеллектуальная основа профессии программиста инженера легла в основу двух нетехнических учебных дисциплин — это обязательные курсы "Психология в ИТ" для студентов 1-го курса и курс "Групповая динамика и коммуникации в профессиональной практике программной инженерии" для студентов 2-го курса. Целью этих дисциплин является непосредственное формирование, развитие и практика использования компонентов универсальных компетенций. Результатом — компетентный специалист с проявленными на практике стремлением и способностью (готовностью) реализовать свой потенциал — знания, умения, опыт, личные качества — для успешной продуктивной деятельности, меняющей статистику Chaos Report [1], при одновременной гибкости, быстрой адаптивной реакции на динамику обстоятельств и среды, ответственности за результаты своей деятельности, понимания необходимости ее постоянного совершенствования и развитым сотрудничеством с коллегами и профессиональной межличностной средой.



**Рис. 3. Модель универсальных компетенций специалистов по направлению "Программная инженерия"**

## Каталог универсальных компетенций специалистов по направлению "Программная инженерия"

№	Название компетенции	Определение компетенции	Компоненты
1	Анализ	Исследования объектов/ситуаций/проблем путем выделения и рассмотрения отдельных составляющих (объекты, ситуации, проблемы) элементов, систематизированные и логичные выводы, основанные на правильно отобранной информации	Умение выбрать стратегию сбора информации Полнота собранной информации Критическая проверка собранной информации Умение анализировать специальную вербальную и числовую информацию Умение идентифицировать источники и элементы проблемы Умение выстраивать причинно-следственные связи, логический вывод Техники и инструменты сбора, анализа и обработки информации Умение вносить ясность в хаос огромного потока быстро меняющейся информации, способность эффективно анализировать большой объем данных Умение рассматривать проблемы с глобальной точки зрения (динамическая система, система взаимодействий и взаимозависимых частей) (системное мышление) Умение определять сложные системы и работать с ними Умение критически мыслить
2	Моделирование	Создание возможных сценариев развития событий (прогнозирование)	Умение создать модель решения Умение разработать комплексное решение Умение внести коррективы в разработанную модель
3	Разработка решения	Выбор рациональных или иррациональных альтернатив, имеющий целью достижение осознанного результата	Умение обнаруживать и структурировать проблемную ситуацию Умение предлагать множественность самых разных решений одной задачи Умение определить требования к решению и выбрать лучший вариант решения Умение работать с совершенно новыми задачами, с которыми раньше не сталкивался Умение предвидеть риски и управлять ими
4	Экспертиза	Накопление и распространение суммы знаний о бизнесе, продуктах, взаимосвязях между ними и способах их применения	Осведомленность о продуктах и решениях Умение выступить в качестве эксперта Стремление развивать уровень профессиональной экспертизы Умение типизировать профессиональные знания Инновационное мышление, создание нового
5	Коммерческий подход	Фокус на экономическую выгоду Компании	Понимание бизнеса Компании/команды Понимание бизнеса клиента Коммерческое обоснование решения
6	Клиентоориентированность	Создание ценности за счет глубокого понимания и удовлетворения потребностей клиентов	Умение прояснить потребности клиента и решить его проблемы Навыки интервьюирования Клиентский фокус Умение оказывать консультационную поддержку клиенту Умение развивать потребности клиента Умение устанавливать контакт Умение работать с возражениями и конфликтами Умение развивать отношения и развивать потребности клиента Умение привлекать новых клиентов
7	Командная работа	Способность работать на общий результат, создавая общее информационное поле, уважая коллег и соблюдая договоренности, подлинное намерение работать совместно с другими, быть частью команды, работать вместе в противопоставление работе по одиночке или конкуренции	Умение понимать и оценивать людей в контексте высокоэффективной команды Умение обмениваться информацией Умение планировать и проводить эффективные встречи Умение оказать поддержку Умение давать развивающую обратную связь Умение сотрудничать друг с другом, высшим руководством и другими командами в процессе решения рабочих задач и развивать отношения Умение конструктивно разрешать конфликты Умение понимать групповую динамику и управлять ею Умение принимать решения в команде: ориентация в проблеме, обсуждение, принятие решения, осуществление Умение использовать креативность для выработки стратегии достижения высоких показателей Ситуационное лидерство
8	Коммуникация	Целенаправленный обмен информацией и ее смысловым содержанием в процессе делового общения	Умение ясно выражать свои мысли и чувства в устной и письменной коммуникации Навыки владения техниками, инструментами подготовки и проведения презентаций, переговоров и создания письменных документов Навыки межотраслевой коммуникации (понимание технологий, процессов и рыночной ситуации в разных смежных и несмежных отраслях) Умение работать в разных культурных средах, с разными поколениями сотрудников Умение слушать, слышать, понимать, регулировать эмоциональное напряжение и поддерживать эмоциональный баланс во взаимодействии с другим человеком (высокий эмоциональный интеллект) Умение учиться друг у друга Управление всеми видами коммуникаций Умение формировать вокруг себя бизнес-сообщества, экспертные сообщества, сообщества единомышленников

№	Название компетенции	Определение компетенции	Компоненты
9	Обеспечение качества и следование стандартам	Следование стандартам работы, выбор требуемого стандарта и его применение	Обеспечение и контроль качества Формирование и следование стандартам деятельности Оптимизация существующих процессов и процедур
10	Планирование и организация деятельности	Оптимальное распределение ресурсов в целях достижения поставленных целей	Навыки эффективной организации собственной деятельности Навыки эффективной организации деятельности проектных команд Управление проектами и процессами
11	Инициативность	Предпочтение в совершении действия, лидерство	Готовность и умение делать больше, чем требуется или ожидается в работе, делать то, чего никто не просил, но что улучшит или увеличит результаты работы и поможет избежать проблем Готовность и умение искать и/или создавать новые возможности Готовность и умение убедить, уговорить, повлиять, оказать воздействие или проинформировать на других определенное впечатление с целью заставить их поддерживать планы говорящего/действующего
12	Нацеленность на результат	Готовность достигать результата вне зависимости от складывающихся внешних условий или новизны задачи, достижение трудных многообещающих целей, ответственность за свои действия и за результат	Решительность Ответственность Драйв, энергия Позитивный настрой
13	Открытость новому/гибкость (agile)	Способность адаптироваться и эффективно работать в разнообразных ситуациях, с отдельными людьми или группами	Готовность и умение работать в режиме высокой неопределенности и быстрой смены условий задач Готовность и умение понимать и ценить различные и противоположные точки зрения на проблему, адаптировать подход по мере изменения требований ситуации Готовность и умение постоянно исследовать новые идеи, возможности; постоянное отражение и поиск конструктивной критики (Открытый ум)
14	Развитие	Систематические усилия по развитию способностей для достижения все более высоких целей	Ориентация на саморазвитие и управление им Широкий междисциплинарный кругозор Умение задавать вопросы (любопытство) Умение учиться на ошибках и постоянно искать новые пути, которые помогут стать лучше в том, что мы делаем

### Заключение

Современные трансформационные процессы в экономике России и мировой экономике в целом повлияли на специфику требований, предъявляемых к высококвалифицированным специалистам в различных сферах бизнеса. Эти изменения непосредственно касаются и всего спектра специалистов направления "Программная инженерия". Работодателям нужна не квалификация, которая, по их представлению, связана с дроблением производственных функций на ряд задач и видов деятельности, а компетентность как соединение навыков, свойственных каждому индивиду, в котором сочетаются квалификация с социальным поведением, способностью работать в группе, инициативностью, умением принимать решения и отвечать за их последствия.

Для системы образования в меняющемся мире задачей номер один должно стать формирование универсальных компетенций как основы любой профессии. Обязательными качествами выпускника должны стать: инициативность, инновационность, мобильность, гибкость, динамизм и конструктивность и т.п. Будущий профессионал должен обла-

дать стремлением к самообразованию на протяжении всей жизни, владеть новыми технологиями и понимать возможности их использования, уметь принимать самостоятельные решения, быстро адаптироваться в социальной и будущей профессиональной сфере, разрешать проблемы и работать в команде, быть готовым к перегрузкам, стрессовым ситуациям и уметь быстро из них выходить.

Безусловно, это требует от современной высшей школы применения активных методов, приемов и форм работы: проектной работы, командной деятельности, проблематизирующих кейсов, геймификации. В аудитории акцент смещается в сторону выполнения практических работ, что существенно влияет на навыки, необходимые преподавателю. Это требует изменения роли преподавателя: он перестает быть "говорящей головой", начитывающей лекции, и становится скорее ментором, наставником, который ведет студенческие команды или конкретных студентов, помогая им реализовывать проекты в рамках семинарских занятий. Он должен быть готов отвечать на вопросы, находиться на передовой практического опыта.



При этих условиях мы сможем "запустить" познавательную, коммуникативную и личностную активность нынешних студентов, сформировать их универсальные компетенции и "выпустить" в профессию и собственно в жизнь компетентного выпускника.

#### Список литературы

1. **Hastie S., Wojewoda S.** Standish Group 2015 Chaos Report — Q & A with Jennifer Lynch [Electronic resource]. URL: <https://www.infoq.com/articles/standish-chaos-2015> (accessed: 30.06.2016)
2. **Bucher.** Maintenance of the Computer Sciences Teleprocessing system // Proceedings of the International Conference on Reliable Software, 1975. P. 260—266.
3. **Rubey R. J.** Higher Order Languages for Avionics Software — A Survey, Summary, and Critique. // Proceedings of the IEEE 1978 National Aerospace and Electronics Conference, NAECON '78. New York: Institute of Electrical and Electronics Engineers, 1978. P. 714—736.
4. **DeMarco T., Lister Timothy R.** Peopleware: Productive Projects and Teams. Addison-Wesley: 3rd Edition, 2013, 249 p.
5. **Boehm Barry W.** Software Engineering Economics. Prentice Hall, 1981. 1317 p.
6. **Липаев В. В.** Человеческие факторы в программной инженерии: рекомендации и требования к профессиональной квалификации специалистов. М.: СИНТЕГ, 2009. 328 с.
7. **Manifesto for Agile Software Development.** [Electronic resource]. URL: <https://www.agilealliance.org/agile101/the-agile-manifesto/> (accessed: 05.07.2016)
8. **Ахен Денис М., Клауз Арон, Тернер Ричард.** CMMI: Комплексный подход к совершенствованию процессов. Практическое введение в модель. М.: "МФК", 2005. 330 с.
9. **Байденко В. И.** Компетенции в профессиональном образовании (к освоению компетентностного подхода) // Высшее образование в России. 2004. № 11. С. 17—22.
10. **Болотов В. А., Сериков В. В.** Компетентностная модель: от идеи к образовательной программе // Педагогика. 2003. № 10. 26 с.
11. **Зимняя И. А.** Ключевые компетенции — новая парадигма результата образования // Высшее образование сегодня. 2003. № 5. С. 22—27.
12. **Орлов А. А., Грачев В. В., Жукова О. А.** Компетентностный подход в высшем профессиональном образовании // Педагогика. 2009. № 2. С. 107—111.
13. **Атлас новых профессий 2.0** / Под ред. Павла Лукши. М.: Олимп-Бизнес, 2016. 288 с.
14. **Петри Ник.** Тенденции в развитии лидерства. [Electronic resource]. URL: <http://www.ccl.org/leadership/pdf/research/futureTrendsRussian.pdf> (accessed: 30.06.2016)
15. **Capitalizing on Complexity: Insights from the Global Chief Executive Officer Study.** [Electronic resource]. URL: <https://public.dhe.ibm.com/common/ssi/ecm/gb/en/gbe03297usen/GBE03297USEN.PDF> (accessed: 30.06.2016)
16. **Guide to the Software Engineering Body of Knowledge Version 3.0 SWEBOK** // IEEE. 2014. 335 p.
17. **Software Engineering 2014. Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering.** Joint Task Force on Computing Curricula // IEEE Computer Society, Association for Computing Machinery. 2015. 134 p.
18. **Software Engineering Competency Model** // IEEE. 2014. 168 p.
19. **SOFTWARE EXTENSION to the PMBOK Guide Fifth Edition** // Project Management Institute. 2013. 247 p.
20. **Руководство к Своду знаний по управлению проектами (Руководство PMBOK).** Пятое издание. Project Management Institute, 2013, 614 с.

**E. N. Reshetova**, Teacher, Software Engineering School,  
Faculty of Computer Science, HSE, [ereshetova@hse.ru](mailto:ereshetova@hse.ru)

## Student Universal Competences of Educational Direction "Software Engineering"

*It is considered one of the most pressing problems of education at the present stage — definition of the universal competence of technicians. We present some of the current approaches to this issue and propose a model of universal jurisdiction as mandatory intellectual basis of professional software engineer.*

**Keywords:** software engineering, universal competences, SWECOM

#### References

1. **Hastie S., Wojewoda S.** Standish Group 2015 Chaos Report — Q & A with Jennifer Lynch [Electronic resource]. URL: <https://www.infoq.com/articles/standish-chaos-2015> (accessed: 30.06.2016)
2. **Bucher.** Maintenance of the Computer Sciences Teleprocessing system, *Proceedings of the International Conference on Reliable Software*, 1975, pp. 260—266.
3. **Rubey R. J.** Higher Order Languages for Avionics Software — A Survey, Summary, and Critique, *Proceedings of the IEEE 1978 National Aerospace and Electronics Conference, NAECON'78*. New York: Institute of Electrical and Electronics Engineers, 1978, pp. 714—736.
4. **DeMarco T., Lister Timothy R.** *Peopleware: Productive Projects and Teams*. Addison-Wesley: 3rd Edition, 2013, 249 p.
5. **Boehm Barry W.** *Software Engineering Economics*. Prentice Hall, 1981, 1317 p.
6. **Lipaev V. V.** *Chelovechskiy faktori v programnoy inzhinerii: rekomendazii i trebovaniya k professionalnoy kvalifikazii spetsialistov*, Moscow: SINTEG, 2009. 328 p.
7. **Manifesto for Agile Software Development.** [Electronic resource]. URL: <https://www.agilealliance.org/agile101/the-agile-manifesto/> (accessed: 05.07.2016)
8. **Ahen Denis M., Klauz Aron, Turner Richard.** *CMMI: Kompleksniy podhod k sovershenstvovaniyu prozessov. Prakticheskoe vvedenie v model*. Moscow, "MFK", 2005. 330 p.
9. **Baydenko V. I.** Kompetentnii v professionalnom obrazovanii (k osvoeniyu kompetentnostnogo podhoda), *Visshee obrazovanie v Rossii*, 2004, no. 11, pp. 17—22.
10. **Bolotov V. A., Serikov V. V.** Kompetentnostny model: ot idei k obrazovatelnoy programme, *Pedagogika*, 2003, no. 10, 26 p.
11. **Zimnyaa I. A.** Kluchevie kompetentzii — novaya paradigma rezultata obrazovaniya, *Visshee obrasovanie segodnya*, 2003, no. 5, pp. 22—27.
12. **Orlov A. A., Grachev V. V., Ghukova O. A.** Kompetentnostny podhod v visshem professionalnom obrazovanii, *Pedagogika*, 2009, no. 2, pp. 107—111.
13. **Atlas novih professiy 2.0.** Pod red. Pavla Lukshi, Moscow: Olimp-Biznes, 2016, 288 p.
14. **Petri Nik.** *Tendenzii v razvitiu liderstva*. [Electronic resource]. URL: <http://www.ccl.org/leadership/pdf/research/futureTrendsRussian.pdf> (accessed: 30.06.2016)
15. **Capitalizing on Complexity: Insights from the Global Chief Executive Officer Study** [Electronic resource]. URL: <https://public.dhe.ibm.com/common/ssi/ecm/gb/en/gbe03297usen/GBE03297USEN.PDF> (accessed: 30.06.2016)
16. **Guide to the Software Engineering Body of Knowledge Version 3.0 SWEBOK**, IEEE, 2014, 335 p.
17. **Software Engineering 2014. Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering.** Joint Task Force on Computing Curricula, *IEEE Computer Society*, Association for Computing Machinery. 2015, 134 p.
18. **Software Engineering Competency Model**, IEEE, 2014, 168 p.
19. **SOFTWARE EXTENSION to the PMBOK Guide Fifth Edition**, *Project Management Institute*, 2013, 247 p.
20. **Rukovodstvo k Svodu znaniyu po upravleniyu projektami (Rukovodstvo PMBOK).** Pyatoe izdanie. Project Management Institute, 2013, 614 p.

УДК 621.39

**Б. И. Соловьев**, канд. техн. наук, доцент, сотрудник,

**Д. Л. Жусов**, канд. техн. наук, сотрудник, e-mail: d.zhusov@mail.ru, **С. А. Просолупов**, сотрудник,  
Академия ФСО России, г. Орел

## Определение мест размещения сетевых ресурсов на квазиоднородной топологической структуре инфокоммуникационной системы

*Представлены формулировка и порядок решения задачи по определению мест размещения ресурсов на структуре инфокоммуникационной сети. Цель работы — разработка способа, обеспечивающего повышение устойчивости доступа пользователей инфокоммуникационной системы. В работе представлена математическая модель топологии системы. Сформулированная цель достигается представлением структуры инфокоммуникационной сети в виде случайного графа и применением схемы пошагового анализа, конструирования и отсева вариантов размещения разнородных ресурсов. Отличительной особенностью предложенного решения является решение по отсеву бесперспективных начальных частей вариантов графа до их полного построения.*

*Представлен порядок расчетов в соответствии с методом последовательного анализа вариантов и предложенной схемой на примере реализации требований по пропускной способности инфокоммуникационной системы.*

*Полученное решение задачи по определению мест размещения сетевых ресурсов на узлах и ребрах инвариантной части топологической структуры инфокоммуникационной системы позволяет организовать устойчивый доступ пользователей независимо от месторасположения по различным маршрутным схемам в любом соотношении.*

**Ключевые слова:** структура, ресурсы, инвариантная часть, двудольный граф, центр графа, медиана графа

### Введение

Цель работы — разработка методики размещения разнородных ресурсов на сетевых элементах, обеспечивающей повышение устойчивости доступа пользователей инфокоммуникационной системы.

Современный этап развития инфокоммуникационных систем связан с повышением производительности средств вычислительной техники, применением высокоскоростных волоконно-оптических систем передачи информации, совершенствованием технологий и средств информационной безопасности. Это обуславливает принципиально новые аспекты содержания, организации и использования сетевых информационных ресурсов (файловых серверов, web-серверов, серверов баз данных и т.п.), а именно:

1. Размещаемые в инфокоммуникационных системах разнородные сетевые ресурсы обладают характеристиками, изменение которых влияет на надежность доступа пользователей к ним. К таким характеристикам относятся:

- пропускная способность;
- совокупность программного обеспечения (специализированного и прикладного), установленного на сетевой ресурс;
- процессорное время;

- размер оперативной памяти;
- размер дискового пространства;
- платформа (комбинация в виде виртуальной машины или сервера, программного обеспечения, процессорного времени, оперативной памяти, дискового пространства);
- информационный фонд.

2. Децентрализация и динамическое распределение ресурсов (распределенный источник ресурсов): структурно разнесенное размещение разнородных ресурсов на множестве сетевых элементов в целях организации устойчивого доступа и их получения по различным маршрутным схемам в любом соотношении. Это определяет отсутствие строго заданной логической структуры системы и в целом наличие заданного масштаба.

3. Инвариантный доступ пользователей: возможность получения услуг заданного качества независимо от его местоположения в сети, что порождает динамичность услуг — изменение их состава на соответствующем узлом элементе системы. При большом числе пользователей (100 и более) и принятой схеме доступа распределение вероятностей поступления требований на предоставление различных услуг не может быть аппроксимировано типовым законом распределения.

## Математическая модель топологии системы

Топологическая структура инфокоммуникационной системы может быть представлена в виде статистического ансамбля связанных неориентированных графов

$$G_T(A, B) = \{(G_1(A, B), p_1), (G_2(A, B), p_2), \dots, (G_q(A, B), p_q), \dots, (G_Q(A, B), p_Q)\},$$

где  $A = \{a_i\}$ ,  $i = \overline{1, N}$ , — множество вершин графа, соответствующее сетевым узлам, или узловая основа сети,  $B = \{b_{ij}\}$ ,  $i, j = \overline{1, N}$ ,  $i \neq j$ ,  $|B| = n$ , — множество ребер, представляющих собой линии, соединяющие данные узлы. Каждый элемент множества может быть представлен случайной величиной, принимающей значения

$$a_i = \begin{cases} 1, & \text{если } i\text{-й узел существует,} \\ & \text{т.е. принадлежит } G_q(A, B), \\ 0, & \text{если } i\text{-й узел не существует,} \\ & \text{т.е. не принадлежит } G_q(A, B), \end{cases}$$

$$b_{ij} = \begin{cases} 1, & \text{если есть непосредственная связь} \\ & \text{между } i\text{-м и } j\text{-м узлами,} \\ 0, & \text{если нет непосредственной связи} \\ & \text{между } i\text{-м и } j\text{-м узлами.} \end{cases}$$

Тем самым, каждый граф  $G_q(A, B)$  имеет определенную вероятность реализации  $p_q$  или каждый элемент ансамбля имеет свой собственный статистический вес.

Исследования в области проектирования и эксплуатации больших сетей [1] показывают, что комбинаторное пространство сетевых структур на заданных множествах  $A$  и  $B$  в общем случае будет неограниченным ввиду существования различного множества графов, представляющих структуру инфокоммуникационных систем. Значительное уменьшение заданного пространства достигается выбором близких к однородным (квазиоднородных или квазирегулярных) структур [2], которые обладают определенной совокупностью таких свойств, как унивалентность, уницентральность, униmodalность и униветвистость. Кроме того, квазиоднородные структуры обладают свойством инвариантности, которое характеризует способность сети обеспечивать распределение собственного ресурса между пользователями  $Z = \{z_k\}$ ,  $z_k = (a_{pk}, a_{gk})$ ,  $k = \overline{1, N(N-1)}$ , в определенных соотношениях. В этом случае минимальное значение сетевого ресурса по узлам и ребрам приведенного покрывающего множества  $\overline{P}_Z^\lambda$ ,  $\lambda = \overline{1, \Lambda_G}$ , будет численно совпадать с минимальным значением сетевого ресурса по узлам и ребрам одного из покрывающих множеств  $\overline{P}_{z_k}^\omega$ ,  $\omega = \overline{1, \Omega_G}$ .

Для получения квазиоднородных структур используется подход, основанный на объединении

цикла Гамильтона и дополнительных ребер, не принадлежащих ему:

$$G_q(A, B) = G_q(A, B = \{b_{ij}\} = [\Gamma_k] \cup b_{kl} \notin \Gamma_k). \quad (1)$$

Данный подход позволяет получить структуру с требуемыми топологическими характеристиками для любого числа узлов. Тем самым формируется базовая структура системы, принадлежащая статистическому ансамблю  $G_T(A, B)$ .

Исследование поведения инфокоммуникационных систем во времени рассматривается как изменение структурных зависимостей элементов множеств  $A$  и  $B$ , приводящее к формированию одного из элементов статистического ансамбля  $G_T(A, B)$ .

Динамическое изменение структуры порождает потребность в обеспечении устойчивого информационного взаимодействия сетевых элементов или организации устойчивого доступа пользователей к различным сетевым ресурсам. Для этого необходимо определить рациональные места размещения разнородных сетевых ресурсов на топологической структуре инфокоммуникационной системы и обеспечить выполнение требований по пропускной способности сетевых элементов.

## Постановка задачи

Начальным этапом решения данной задачи является определение числа и мест расположения сетевых информационных ресурсов (файловых серверов, web-серверов, серверов баз данных и т.п.) на узловой основе инфокоммуникационной системы.

Составляющей устойчивости доступа пользователей является надежность  $K_{\Gamma, d}^{s, q}$ ,  $d = \overline{1, D}$ , функционирования информационного ресурса каждого типа  $q = \overline{1, Q}$ . Если отказы нескольких информационных ресурсов одного типа независимы, система из  $d$  ресурсов одного типа представляет собой составной элемент из  $d$  параллельно включенных элементов. Надежность такого составного элемента  $H^{s, q}$  можно представить в виде:

$$H^{s, q} = 1 - \prod_{d=1}^D (1 - K_{\Gamma, d}^{s, q}). \quad (2)$$

Полагая, что надежности информационных ресурсов  $K_{\Gamma, d}^{s, q}$  одного типа равны между собой, их число  $D$  можно получить согласно следующему соотношению:

$$D \geq \frac{(1 - H_{\text{тр}}^{s, q})}{(1 - K_{\Gamma, d}^{s, q})}, \quad (3)$$

где  $H_{\text{тр}}^{s, q}$  — требуемое значение надежности функционирования системы информационных ресурсов.

Для определения мест размещения информационных ресурсов необходимо выбрать узлы системы с таким расчетом, чтобы расстояние от этого узла до самого удаленного узла и суммарная протяжен-

ность от исходного узла до остальных узлов были минимальными. В первом случае узел системы, на котором размещается информационный ресурс, будет являться центром, а во втором — медианой графа. Таким образом, возникает задача нахождения на  $G_q(A, B)$  вершины графа, которая является одновременно и центром, и медианой. Для определения центра решается минимаксная задача:

$$a = \arg \min_{a \in A} \max_{i, j = \overline{1, N}, i \neq j} \rho_{ij}(a_i, a_j), \quad (4)$$

где  $\rho(a_i, a_j)$  — число ребер в пути между узлами  $a_i$  и  $a_j$ .

Для нахождения медианы решается минисуммарная задача:

$$a = \arg \min_{a \in A} \sum_i \rho(a_i, a_j). \quad (5)$$

Эффективные алгоритмы решения задач (4) и (5) представлены в работах [3, 4]. Ввиду того что топологическая квазиоднородная структура  $G_q(A, B)$  обладает свойством уницентральности, то существует возможность одновременной реализации требований по размещению информационных ресурсов решением задач (4) и (5).

Реализация требований по пропускной способности информационной системы  $V_\Sigma = \sum_{k=1}^m V_{z_k}$  осуществляется в инвариантной части системы, что позволяет выполнить указанные требования независимо от расположения пользователей. Для этого на топологической структуре инфокоммуникационной системы определяется основная система сечений  $S = \{\sigma_1, \sigma_2, \dots, \sigma_n, \dots, \sigma_N\}$ , где  $\sigma_i$  — сечение сети относительно выбранного остова дерева на  $G_q(A, B)$ . В каждом сечении реализуется поток

$$V_\Sigma = \sum_{k=1}^m V_{z_k} \text{ распределением вида:}$$

$$X(V_\Sigma) = \{x_{\sigma_1}(V_\Sigma), x_{\sigma_2}(V_\Sigma), \dots, x_{\sigma_n}(V_\Sigma), \dots, x_{\sigma_{N-1}}(V_\Sigma)\}. \quad (6)$$

При этом значение пропускной способности каждого сечения  $C\{\sigma_n\}$  должно соответствовать системе

$$\begin{cases} C\{\sigma_n\} \geq \sum_{\substack{a_p \in A_s \\ a_q \in A_t}} x_{p,q}; \\ \sum_{\substack{a_p \in A_s \\ a_q \in A_t}} x_{p,q} = V_\Sigma; \\ p, q = \overline{1, N}, n = \overline{1, N-1}, \end{cases} \quad (7)$$

где  $x_{p,q}$  — значения потоков между узлами, разделенных сечением  $\sigma_n$  на два подмножества  $A_s$  и  $A_t$ .

При этом мощность сечения  $h_{\sigma_n}$  (число входящих линий) должна быть:

$$h_{\sigma_n} > H_{CB}, \quad (8)$$

где  $H_{CB}$  — значение связности системы, задаваемое исходя из обеспечения требуемой структурной надежности.

Решение задачи по реализации потока  $V_\Sigma = \sum_{k=1}^m V_{z_k}$  заключается в нахождении множества

$$\{B_{\sigma_1}, B_{\sigma_2}, \dots, B_{\sigma_n}, \dots, B_{\sigma_N}\}, B_{\sigma_n} = \{b_j\}, b_j \in B, j = \overline{1, n},$$

двудольных графов  $G(A_{s_n}, B_{\sigma_n}, A_{t_n})$ , образованных множеством сечений  $S = \{\sigma_1, \sigma_2, \dots, \sigma_n, \dots, \sigma_N\}$ , удовлетворяющего ограничениям (7)–(8) при минимальном значении целевой функции:

$$W = \sum_{b_j \in B_{\sigma_n}} u_j l_j, \quad (9)$$

где  $u_j$  — пропускная способность ребра,  $l_j$  — протяженность ребра  $G(A_{s_n}, B_{\sigma_n}, A_{t_n})$ .

Решение данной задачи выполняется поэтапно. Каждому этапу ставится в соответствие синтез двудольного графа  $G(A_{s_n}, B_{\sigma_n}, A_{t_n})$ ,  $n = \overline{1, N-1}$ , в котором определяются дополнительные дуги  $b_{pq} \in B_{\sigma_n}$ , обеспечивающие реализацию недостающей пропускной способности:

$$\Delta C(\sigma_n) = V_\Sigma - C(\sigma_{n-1}). \quad (10)$$

Каждый этап решения оценивается величиной суммарных затрат, определяемых функцией (9) и рекуррентным соотношением

$$W_n(C(\sigma_n)) = \min_{x(\Delta C_{\sigma_n})} \{W_{n-1}(C(\sigma_{n-1})) + W_n(V_\Sigma - C(\sigma_{n-1}))\}, \quad (11)$$

где  $W_n(C(\sigma_n))$  — затраты на  $n$ -м этапе;  $W_{n-1}(C(\sigma_{n-1}))$  — затраты на  $(n-1)$ -м этапе;  $W_n(V_\Sigma - C(\sigma_{n-1}))$  — затраты перехода от одного этапа к другому.

Таким образом, представленная схема пошагового анализа, конструирования и отсева вариантов заключается в таких способах развития вариантов и построения операторов их анализа, которые позволяют отсеивать невыгодные (бесперспективные) начальные части вариантов до их полного построения — по мере того, как эту бесперспективность удастся обнаружить. Поскольку при отсеивании бесперспективных начальных частей вариантов отсеивается и множество их продолжений, то происходит значительная экономия в вычислительной процедуре. Таким образом, использование предложенного метода позволит эффективно решать задачу реализации требований по пропускной способности инфокоммуникационной системы.

### Решение задачи

Сформулированная задача имеет достаточно большое число вариантов решения, в связи с чем поиск наиболее рационального решения методом

прямого перебора оказывается практически неосуществимым. Учитывая характер и особенности данной задачи, для ее решения можно применить метод динамического программирования [4]. Однако с увеличением масштабов проектируемой сети существенно возрастает размерность поставленной задачи, что ограничивает использование данного метода для ее решения.

В этом случае наиболее приемлемым способом решения задачи по построению двудольных графов является метод последовательного анализа вариантов [1]. В основе этого метода лежит идея представления процесса решения в виде многоступенчатой структуры. Каждая ступень связана с проверкой наличия тех или иных свойств у подмножества вариантов, что ведет либо к непосредственному сокращению исходного множества вариантов, либо подготавливает возможность такого сокращения в будущем. На основе теоретического и практического анализа определяются отличительные свойства каждого варианта и признаки, по которым проводится сравнение полученного множества вариантов. После этого выбор численной схемы решения состоит в выборе рационального порядка проверки признаков, позволяющего провести отсев неконкурентоспособных вариантов и найти оптимальный.

Множество  $V$  возможных вариантов решения  $v = \{B_{\sigma_n}\}$ ,  $n = \overline{1, N-1}$ , задается числом двудольных графов  $G(A_{s_n}, B_{\sigma_n}, A_{t_n})$  и последовательностью их построения. Для пошагового конструирования решения указывается правило и выбора частичных решений  $v = \{B_{\sigma_1}, B_{\sigma_2}, \dots, B_{\sigma_s}\}$ ,  $s \leq N-1$ , подлежащих развитию на каждом шаге, и строится набор тестов  $\varepsilon$ , осуществляющих отсев частичных решений, которые не могут быть достроены до оптимальных. Каждое частичное решение  $v_s$ ,  $s = \overline{1, N-2}$ , образует родовое множество  $R(v_s) \in V$ , состоящее из элементов, у которых  $v_s$  является начальным отрезком. Соответственно, совокупность всех конечных отрезков элементов родового множества  $R(v_s)$  образует множество продолжений  $D(v_s)$ . Размерность частичного решения, подлежащего анализу, соответствует номеру этапа построения двудольного графа. Отсев бесперспективных вариантов решения задачи основан на использовании принципа монотонной рекурсивности: если для двух частичных решений  $v_s^1$  и  $v_s^2$  выполняются соотношения

$$W(v_s^1) < W(v_s^2), D(v_s^1) \supseteq D(v_s^2), \quad (12)$$

то родовое множество  $R(v_s^2)$  не содержит оптимальных решений задачи минимизации монотонно рекурсивной функции  $W$ , определяемой соотношением (11) на множестве возможных вариантов решения  $V$ .

### Пример решения задачи

Рассмотрим пример реализации требований по пропускной способности в инвариантной части на топологической структуре инфокоммуникационной системы, представленной на рис. 1.

Исходными данными задачи является матрица расстояний:

$$L = \|l_{ij}\| = \begin{vmatrix} \infty & 5 & 0 & 0 & 6 \\ 5 & \infty & 7 & 0 & 7 \\ 0 & 7 & \infty & 5 & 6 \\ 0 & 0 & 5 & \infty & 4 \\ 6 & 7 & 6 & 4 & \infty \end{vmatrix}. \quad (13)$$

Необходимо определить распределение требований по пропускной способности  $V_{\Sigma} = 6$  при дискретном значении возможного ресурса  $U = (2, 4)$  и реализации требований по устойчивости доступа  $h_{\sigma_n} = 2$ .

Первоначальным этапом решения задачи является определение минимального остовного дерева на топологической структуре инфокоммуникационной системы. Для его нахождения могут быть использованы алгоритмы Прима или Краскала, представленные в работах [3, 4]. Результатом является остовное дерево минимальной длины, представленное на рис. 2.

Полученное остовное дерево позволяет найти основную систему сечений посредством построения двудольных графов из множества вершин

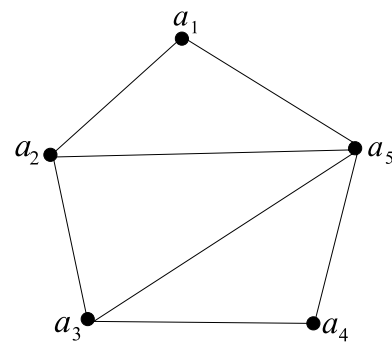


Рис. 1. Топологическая структура инфокоммуникационной системы

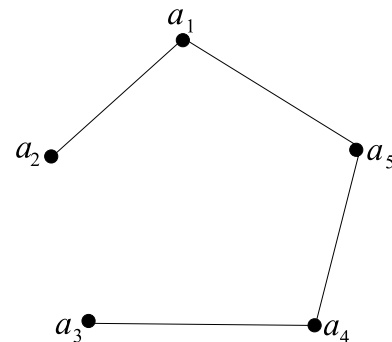


Рис. 2. Остовное дерево минимальной длины

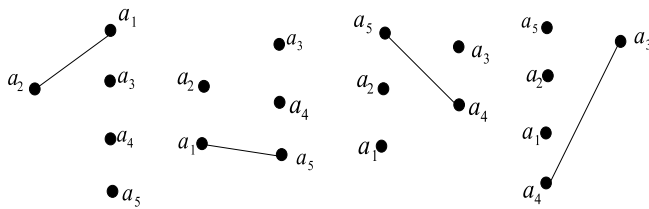


Рис. 3. Совокупность двудольных графов

двух подмножеств  $A_{s_i}$  и  $A_{t_i}$  (рис. 3). Множества  $A_{s_i}$  и  $A_{t_i}$  представлены следующим набором вершин (см. таблицу):

**Множества вершин для формирования двудольных графов**

$A_{s_1} = \{a_2\}$	$A_{t_1} = \{a_1, a_3, a_4, a_5\}$
$A_{s_2} = \{a_1, a_2\}$	$A_{t_2} = \{a_3, a_4, a_5\}$
$A_{s_3} = \{a_1, a_2, a_5\}$	$A_{t_3} = \{a_3, a_4\}$
$A_{s_4} = \{a_1, a_2, a_4, a_5\}$	$A_{t_4} = \{a_3\}$

Далее поэтапно для каждого двудольного графа определяется множество ребер исходного графа, на которых осуществляется реализация требований по пропускной способности  $V_{\Sigma} = 6$ .

На первом этапе для двудольного графа при  $A_{s_1} = \{a_2\}$  и  $A_{t_1} = \{a_1, a_3, a_4, a_5\}$  имеют место следующие допустимые варианты решения с соответствующими значениями целевых функций (9):

- $U_{12} = 4; U_{23} = 2; W_1^1 = 4 \cdot 5 + 2 \cdot 7 = 34.$
- $U_{12} = 4; U_{25} = 2; W_1^2 = 4 \cdot 5 + 2 \cdot 7 = 34.$
- $U_{12} = 2; U_{23} = 4; W_1^3 = 2 \cdot 5 + 4 \cdot 7 = 38.$
- $U_{12} = 2; U_{25} = 4; W_1^4 = 2 \cdot 5 + 4 \cdot 7 = 38.$
- $U_{12} = 2; U_{23} = 2; U_{25} = 2;$   
 $W_1^5 = 2 \cdot 5 + 2 \cdot 7 + 2 \cdot 7 = 38.$

На втором этапе для двудольного графа при  $A_{s_2} = \{a_1, a_2\}$  и  $A_{t_2} = \{a_3, a_4, a_5\}$  имеют место следующие допустимые варианты решения с соответствующими значениями целевых функций:

- $U_{15} = 4; W_2^1 = 4 \cdot 6 = 24.$
- $U_{15} = 2; W_2^2 = 2 \cdot 6 = 12.$
- $U_{15} = 2; U_{23} = 2; W_2^3 = 2 \cdot 6 + 2 \cdot 7 = 26.$

В соответствии с рекуррентным соотношением (11) суммарные затраты на втором этапе будут

$$W_2^{(1)} = \min\{(34 + 26)\} = 60.$$

$$W_2^{(2)} = \min\{(34 + 26), (38 + 12)\} = 50.$$

На третьем этапе для двудольного графа при  $A_{s_3} = \{a_1, a_2, a_5\}$  и  $A_{t_3} = \{a_3, a_4\}$  имеют место следующие допустимые варианты решения с соответствующими значениями целевых функций:

- $U_{54} = 4; U_{53} = 2; W_3^1 = 4 \cdot 4 + 2 \cdot 6 = 28.$
- $U_{54} = 4; W_3^2 = 4 \cdot 4 = 16.$
- $U_{54} = 2; U_{53} = 2; W_3^3 = 2 \cdot 4 + 2 \cdot 6 = 20.$
- $U_{54} = 2; U_{53} = 4; W_3^4 = 2 \cdot 4 + 4 \cdot 6 = 32.$
- $U_{54} = 2; U_{53} = 2; U_{23} = 2;$   
 $W_3^5 = 2 \cdot 4 + 2 \cdot 6 + 2 \cdot 7 = 34.$

В соответствии с рекуррентным соотношением (11) суммарные затраты на третьем этапе будут

$$W_3^{(1)} = \min\{(50 + 28), (60 + 20)\} = 78.$$

$$W_3^{(2)} = \min\{(50 + 32)\} = 82.$$

$$W_3^{(3)} = \min\{(60 + 16)\} = 76.$$

На четвертом этапе для двудольного графа при  $A_{s_4} = \{a_1, a_2, a_4, a_5\}$  и  $A_{t_4} = \{a_3\}$  имеют место следующие допустимые варианты решения с соответствующими значениями целевых функций:

- $U_{34} = 4; U_{35} = 2; W_4^1 = 4 \cdot 5 + 2 \cdot 6 = 32.$
- $U_{34} = 4; W_4^2 = 4 \cdot 5 = 20.$
- $U_{34} = 2; W_4^3 = 2 \cdot 5 = 10.$

Суммарные затраты на четвертом этапе будут определяться

$$W_4^{(1)} = \min\{(76 + 32), (78 + 20), (82 + 10)\} = 92.$$

В результате получаем решение задачи  $U_{12} = 2, U_{15} = 2, U_{25} = 2, U_{23} = 2, U_{35} = 4, U_{34} = 2, U_{45} = 2$  по определению пропускных способностей ребер на топологической структуре инфокоммуникационной системы (рис. 4).

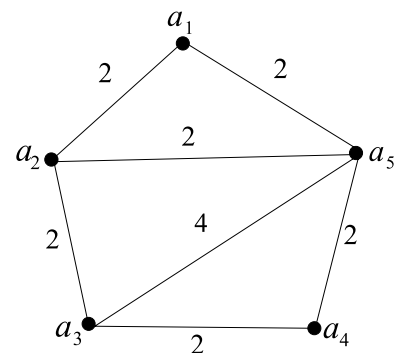


Рис. 4. Реализация требований по пропускной способности в инвариантной части на топологической структуре инфокоммуникационной системы

Экспериментальное исследование эффективности методов упорядоченного перебора свидетельствует об эффективности метода последовательного анализа вариантов для решения поставленной задачи [1]. Так, в результате проведенного эксперимента по решению задачи методом последовательного анализа вариантов при 100 элементах сети и, соответственно, возможных  $10^{14}$  вариантах размещения сетевых ресурсов размерность решений сокращена до 6 вариантов за время, не превышающее 10 мин, а с учетом вычислительных возможностей современных средств вычислительной техники время решения задачи может быть дополнительно сокращено.

### Заключение

Полученное решение задачи по определению мест размещения сетевых ресурсов на узлах и ребрах инвариантной части топологической структуры инфо-

коммуникационной системы позволяет определить рациональные места размещения разнородных сетевых ресурсов и обеспечить выполнение требований по пропускной способности сетевых элементов. Это обеспечивает устойчивый доступ пользователей независимо от месторасположения по различным маршрутным схемам, что определяет достижение цели работы.

### Список литературы

1. **Михалевич В. С., Волкович В. Л.** Вычислительные методы исследования и проектирования сложных систем. М.: Наука, Главная редакция физико-математической литературы, 1982. 283 с.
2. **Артамонов Г. Т., Тюрин В. Д.** Топология сетей ЭВМ и многопроцессорных систем. М.: Радио и связь, 1991. 248 с.
3. **Christofides N.** Graph theory an algorithmic approach. Academic Press Inc. (London) LTD., 1975. 400 p.
4. **Ore O.** Theory of graphs. American Mathematical Society Providence, RHODE ISLAND, 1962. 270 p.
5. **Bellman R. E.** Dynamic Programming. Princeton University Press, 2010. 340 p.

**B. I. Soloviev**, employee, **D. L. Zhusov**, employee, e-mail: d.zhusov@mail.ru, **S. A. Prosolupov**, employee  
The Academy of Federal Security Guard Service of the Russian Federation

## Determining the Placement of Network Resources on a Quasi-Homogeneous Topological Structure of Information and Communication Systems

*The paper presents the formulation and procedure for determining the locations of resources on the network information and communication structure. The aim of this work was the development of the method provides enhancing the stability of user access information and communication systems. The paper presents a mathematical model of the system topology. The formulated goal is achieved by representation of the structure of the info-communication network as a random graph and using step scheme of analysis, design, and elimination properties of heterogeneous resources. A distinctive feature of the proposed solution is solving the drop-out unpromising initial parts of options before their full count of the build.*

*The procedure of calculations in accordance with the method of successive analysis of variants and the proposed scheme on the example of implementation of the requirements on the bandwidth information and communication systems.*

*The solution of a problem of determination of locations of network resources on the nodes and edges of the invariant part of the topological structure of the information communication system allows you to organize sustainable user access regardless of location on different routing schemes in any proportion.*

**Keywords:** structure, resources, invariant part, two-submultiple count, center of the count, median of the count

### References

1. **Mihalevich V. S., Volkovich V. L.** *Vychislitel'nye metody issledovaniya i proektirovaniya slozhnyh sistem*, Moscow: Nauka, Glavnaja redakcija fiziko-matematicheskoy literatury, 1982, 283 p.
2. **Artamonov G. T., Tjurin V. D.** *Topologija setej JeVM i mnogo-processornyh sistem*, Moscow: Radio i svjaz', 1991, 248 p.

3. **Christofides N.** *Graph theory an algorithmic approach*, Academic Press Inc. (London) LTD., 1975, 400 p.
4. **Ore O.** *Theory of graphs*, American Mathematical Society Providence, RHODE ISLAND, 1962, 270 p.
5. **Bellman R. E.** *Dynamic Programming*, Princeton University Press, 2010. 340 p.

УДК 519.872

**В. Н. Тарасов**, д-р техн. наук, проф., зав. каф., e-mail: tarasov-vn@psuti.ru,  
**Л. В. Липилина**, аспирант, e-mail: mila199113@gmail.com,  
**Н. Ф. Бахарева**, д-р техн. наук, проф., зав. каф., e-mail: nadin1956\_04@inbox.ru,  
Поволжский государственный университет телекоммуникаций и информатики, г. Самара

## Автоматизация расчета характеристик систем массового обслуживания для широкого диапазона изменения их параметров

*Приведено описание программы расчета характеристик систем массового обслуживания (СМО) G/G/1, которым соответствуют входные распределения с коэффициентами вариаций интервала между поступлениями требований и временами обслуживания, меньшими и большими 1, т.е. перекрывающими весь интервал (0, ∞). Для этого рассмотрены СМО M/M/1 с запаздыванием во времени, H<sub>2</sub>/M/1 и H<sub>2</sub>/H<sub>2</sub>/1 с гиперэкспоненциальными входными распределениями 2-го порядка. В качестве выходных характеристик использованы среднее время ожидания требований в очереди и дисперсия времени ожидания. Остальные характеристики рассматриваемых систем являются производными от среднего времени ожидания.*

**Ключевые слова:** системы массового обслуживания, среднее время ожидания в очереди, метод спектрального разложения, интегральное уравнение Линдли, преобразование Лапласа

### Введение

В статье представлено описание программы расчета задержки в системах массового обслуживания (СМО) для широкого диапазона изменения параметров трафика. Для этого рассмотрены СМО H<sub>2</sub>/H<sub>2</sub>/1, H<sub>2</sub>/M/1 и M/M/1 с запаздыванием во времени. Здесь использованы следующие обозначения: H<sub>2</sub> — гиперэкспоненциальный закон распределения 2-го порядка, M — экспоненциальный закон распределения. Гиперэкспоненциальный закон распределения H<sub>2</sub> может быть использован для аппроксимации произвольных распределений в случае коэффициента вариации, большего 1. В связи с тем что распределение H<sub>2</sub> является трехпараметрическим, в программе реализован механизм аппроксимации произвольных законов распределений гиперэкспоненциальным распределением как на уровне двух первых моментов, так и на уровне трех первых моментов.

В том случае, когда коэффициент вариации случайной величины меньше 1, для аппроксимации закона распределения можно использовать экспоненциальный закон с запаздыванием во времени. Результаты авторов, полученные для системы M/M/1 с запаздыванием во времени, впервые опубликованы в работе [1]. Также следует отметить тот факт, что оригинальные результаты авторов по СМО H<sub>2</sub>/H<sub>2</sub>/1 и H<sub>2</sub>/M/1 с учетом выявленных возможностей аппроксимации законов распределений как на уровне двух, так и на уровне трех первых моментов случайных величин гиперэкспоненциаль-

ным распределением позволяют их применять для исследования современного телетрафика, чувствительного к задержкам. Это возможно, если учитывать определение вариации задержки (джиттера) в телекоммуникациях как разброс времени ожидания от его среднего значения, тем самым получим возможность определения джиттера через дисперсию [2]. В этом и заключается актуальность проводимых исследований в теории массового обслуживания.

Краткий обзор отечественных и зарубежных источников в области телетрафика позволяет сделать следующие выводы. В работах [3, 4] используются известные классические модели массового обслуживания M/M/1, M/G/1 и верхние границы для времени ожидания требований для системы G/G/1, или же в отсутствие теоретических результатов применяется имитационное моделирование. В зарубежной литературе, например [5], уже давно известны эвристические формулы для систем G/M/1 и G/G/1. В частном случае для системы H<sub>2</sub>/M/1 в работе [5] приведено выражение для среднего времени ожидания:

$$\bar{W} = \frac{\rho}{\mu(1-\rho)} \left\{ 1 + \frac{1}{\rho} \left[ \sqrt{(1-\theta)^2 + (r-1)} - (1-\theta) \right] \right\}, \quad (1)$$

где  $\mu$  — интенсивность обслуживания; промежуточные параметры:  $\theta = \frac{\rho(q-r) - (q-r^2)}{2\rho(r-1)}$ ,  $q = \frac{\sqrt[3]{\tau_\lambda^3}}{6(\bar{\tau}_\lambda)^3}$ ,



$$r = \frac{\overline{\tau_\lambda^2}}{6(\overline{\tau_\lambda})^2}, \text{ а } \overline{\tau_\lambda}, \overline{\tau_\lambda^2}, \overline{\tau_\lambda^3} \text{ — начальные моменты до}$$

3-го порядка интервалов между поступлениями требований в систему;  $\rho$  — загрузка системы. Результаты, полученные из выражения (1) для этой системы, полностью совпадают с авторскими [7].

В предлагаемой программной системе использованы полученные авторами теоретические результаты на основе решения интегрального уравнения Линдли относительно функции распределения времени ожидания известным методом спектрального разложения [6].

### Постановка задачи

Как известно из теории массового обслуживания (ТМО) [6], среднее время ожидания требований в очереди является составной частью задержки в сетях пакетной передачи данных. В классической СМО М/М/1 оно выражается равенством (здесь и далее используется классическое трехпозиционное обозначение Кендалла):

$$\overline{W} = \frac{\rho/\mu}{1-\rho}, \quad (2)$$

для системы М/Г/1 —

$$\overline{W} = \frac{\lambda \overline{X^2}/2}{1-\rho}. \quad (3)$$

Для системы Г/Г/1 это время равно

$$\overline{W} = \frac{D_\lambda + D_\mu + (1-\rho)^2/\lambda^2}{2(1-\rho)/\lambda} - \frac{\overline{I^2}}{2\overline{I}}. \quad (4)$$

В представленных формулах использованы следующие обозначения:  $\rho$  — коэффициент загрузки системы ( $0 < \rho = \lambda/\mu < 1$ );  $\lambda$  — интенсивность входного потока;  $\mu$  — интенсивность обслуживания;

$\overline{X^2}$  — второй начальный момент времени обслуживания;  $D_\lambda, D_\mu$  — соответственно дисперсии интервалов

поступления и времени обслуживания,  $\overline{I}$  и  $\overline{I^2}$  — среднее значение и второй начальный момент периода простоя соответственно. Второе слагаемое в правой части выражения (4) остается неизвестным, и вполне возможно, что оно может содержать моменты интервалов поступления и времени обслуживания более высокого порядка, чем первые два. Поэтому при анализе СМО Г/Г/1 (с произвольными законами поступления и обслуживания требований в системе) необходимо учитывать не только первые два момента случайных интервалов времен поступления и обслуживания, но и моменты более высокого порядка. И наконец, выражения (2)—(4) убедительно демонстрируют зависимость основной характеристики СМО — среднего времени ожидания требований в очереди от вида входных распределений. Ра-

венства (2)—(4) также можно интерпретировать как эволюцию СМО.

### Методы расчета характеристик рассматриваемых систем массового обслуживания

Рассмотрим сначала СМО  $H_2/H_2/1$ . В связи с тем, что система  $H_2/H_2/1$  принадлежит классу систем Г/Г/1, ее точный анализ представляет собой актуальную задачу. При этом СМО с гиперэкспоненциальными входными распределениями, в отличие от систем с другими распределениями, позволяет получить решение задачи в аналитическом виде.

В СМО  $H_2/H_2/1$  интервалы между соседними требованиями входного потока распределены по закону

$$a(t) = p\lambda_1 e^{-\lambda_1 t} + (1-p)\lambda_2 e^{-\lambda_2 t}, \quad (5)$$

а время обслуживания — по закону

$$b(t) = q\mu_1 e^{-\mu_1 t} + (1-q)\mu_2 e^{-\mu_2 t}. \quad (6)$$

Воспользуемся свойством преобразования Лапласа воспроизведения моментов и запишем начальные моменты до 3-го порядка для распределения (5):

$$\overline{\tau_\lambda} = \frac{p}{\lambda_1} + \frac{(1-p)}{\lambda_2};$$

$$\overline{\tau_\lambda^2} = \frac{2p}{\lambda_1^2} + \frac{2(1-p)}{\lambda_2^2}; \quad \overline{\tau_\lambda^3} = \frac{6p}{\lambda_1^3} + \frac{6(1-p)}{\lambda_2^3}. \quad (7)$$

Выражения для начальных моментов позволяют определить неизвестные параметры  $\lambda_1, \lambda_2, p$  распределения (5) методом моментов при аппроксимации произвольных распределений гиперэкспоненциальным распределением. Аналогично определяются неизвестные параметры  $\mu_1, \mu_2, q$  распределения (6).

Для системы  $H_2/H_2/1$  в работе [8] приведено преобразование Лапласа для функции плотности времени ожидания:

$$W^*(s) = \frac{s_1 s_2 (s + \mu_1)(s + \mu_2)}{\mu_1 \mu_2 (s + s_1)(s + s_2)}, \quad (8)$$

где  $-s_1$  и  $-s_2$  — отрицательные вещественные части корней кубического уравнения

$$s^3 - c_2 s^2 - c_1 s - c_0 = 0. \quad (9)$$

Коэффициенты этого уравнения:  $c_0 = a_0 b_1 - a_1 b_0 - a_0(\mu_1 + \mu_2) + b_0(\lambda_1 + \lambda_2)$ ,  $c_1 = -a_1 b_1 - a_0 - b_0 + (\lambda_1 + \lambda_2)(\mu_1 + \mu_2)$ ,  $c_2 = \lambda_1 + \lambda_2 - \mu_1 - \mu_2$ , а параметры:  $a_0 = \lambda_1 \lambda_2$ ,  $a_1 = p\lambda_1 + (1-p)\lambda_2$ ,  $b_0 = \mu_1 \mu_2$ ,  $b_1 = q\mu_1 + (1-q)\mu_2$ . Следовательно, все они однозначно выражаются через параметры распределений (5) и (6), которые, в свою очередь, определяются методом моментов решением систем уравнений типа (7).

Среднее время ожидания в очереди равно значению производной от функции преобразования Лапласа (8) со знаком минус в точке  $s = 0$ . Окон-

чательно среднее время ожидания в очереди для СМО  $H_2/H_2/1$ :

$$\bar{W} = \frac{1}{s_1} + \frac{1}{s_2} - \frac{1}{\mu_1} - \frac{1}{\mu_2}. \quad (10)$$

Выражение (8) на основе свойств преобразования Лапласа позволяет также определить моменты высших порядков для времени ожидания. Например, начальный момент 2-го порядка времени ожидания равен значению второй производной от преобразования (8) в точке  $s = 0$ . Он будет иметь вид

$$\bar{W}^2 = \frac{d^2 W^*(s)}{ds^2} = \frac{2}{\mu_1 \mu_2} - \frac{2}{s_1 s_2} - \frac{2(s_1 + s_2)}{\mu_1 \mu_2 s_1^2 s_2^2} [s_1 s_2 (\mu_1 + \mu_2) - \mu_1 \mu_2 (s_1 + s_2)]. \quad (11)$$

В свою очередь, начальный момент 2-го порядка позволяет определить дисперсию времени ожидания. Учитывая определение джиттера в телекоммуникациях как разброс времени ожидания от его среднего значения, тем самым получим возможность его определения через дисперсию. Этот факт является результатом, важным для анализа трафика, чувствительного к задержкам.

Основной объем вычислений в программе и алгоритме для СМО  $H_2/H_2/1$  приходится на решение систем уравнений вида (7) методом Ньютона и на нахождение отрицательных вещественных частей корней кубического уравнения (9) методом Виета—Кардано.

Рассмотрим теперь систему типа  $H_2/M/1$ . В этом случае функция плотности  $a(t)$  имеет вид (5), а  $b(t) = \mu e^{-\mu t}$ . Неизвестные параметры  $\lambda_1, \lambda_2, p$  распределения  $a(t)$  определяются точно так же, как и в случае СМО  $H_2/H_2/1$ .

Ниже приведены полученные в работах [7, 9] результаты для времени ожидания в системе  $H_2/M/1$ . В этом случае преобразование Лапласа для функции плотности времени ожидания для системы, полученное решением интегрального уравнения Линдли методом спектрального разложения [6], имеет следующий вид:

$$W^*(s) = \frac{s_1(s + \mu)}{\mu(s + s_1)}, \quad (12)$$

где  $-s_1 = -(\sqrt{c_2^2/4 + c_1} - c_2/2)$  — отрицательный корень квадратного уравнения

$$s^2 - c_2 s - c_1 = 0 \quad (13)$$

с коэффициентами:  $c_1 = \mu[\lambda_1(1 - p) + \lambda_2 p] - \lambda_1 \lambda_2$ ,  $c_2 = \lambda_1 + \lambda_2 - \mu$ . Величины  $p, \lambda_1, \lambda_2$  являются параметрами гиперэкспоненциального закона рас-

пределения с функцией плотности  $a(t) = p\lambda_1 e^{-\lambda_1 t} + (1 - p)\lambda_2 e^{-\lambda_2 t}$  для системы  $H_2/M/1$ .

Среднее время ожидания в очереди равно значению производной от функции преобразования Лапласа (12)

$$\frac{dW^*(s)}{ds} = \frac{s_1 \mu (s_1 + s) - s_1 (s + \mu) \mu}{\mu^2 (s + s_1)^2}$$

со знаком минус в точке  $s = 0$ :

$$\bar{W} = -\left. \frac{dW^*(s)}{ds} \right|_{s=0} = \frac{-s_1^2 \mu + \mu^2 s_1}{\mu^2 s_1^2} = \frac{1}{s_1} - \frac{1}{\mu}.$$

Окончательно среднее время ожидания в очереди для СМО  $H_2/M/1$ :

$$\bar{W} = 1/s_1 - 1/\mu. \quad (14)$$

Выражение (12) на основе свойств преобразования Лапласа позволяет также определить моменты высших порядков для времени ожидания. Например, начальный момент 2-го порядка времени ожидания равен значению второй производной от преобразования (12) в точке  $s = 0$ , и он будет иметь вид

$$\bar{W}^2 = \left. \frac{d^2 W^*(s)}{ds^2} \right|_{s=0} = \left. \frac{2s_1(\mu - s_1)}{\mu(s + s_1)^3} \right|_{s=0} = \frac{2(\mu - s_1)}{\mu s_1^2}. \quad (15)$$

В свою очередь, начальный момент 2-го порядка позволяет определить дисперсию времени ожидания:  $D_W = \bar{W}^2 - \bar{W}^2$ .

Основной объем вычислений в программе для СМО  $H_2/M/1$  приходится на решение системы уравнений типа (7) и на нахождение отрицательной вещественной части корня квадратного уравнения (13).

Перейдем теперь к анализу характеристик системы  $M/M/1$  с запаздыванием во времени. Рассмотрим СМО, на вход которой поступают требования, случайные интервалы между которыми распределены по закону

$$a(t) = \begin{cases} \lambda e^{-\lambda(t-t_0)}, & t \geq t_0; \\ 0, & 0 \leq t < t_0. \end{cases} \quad (16)$$

Аналогично распределено и время обслуживания:

$$b(t) = \begin{cases} \mu e^{-\mu(t-t_0)}, & t \geq t_0; \\ 0, & 0 \leq t < t_0. \end{cases} \quad (17)$$

Функции плотности (16) и (17) являются сдвинутыми вправо от нулевой точки на величину  $t_0$  экспоненциальными распределениями с двумя параметрами  $(\lambda, t_0)$  и  $(\mu, t_0)$ , причем  $\lambda < \mu$ . Таким образом, имеем СМО с запаздыванием во времени на величину  $t_0 > 0$ , которая впервые описана в работе [1].

В этой работе показано, что коэффициент вариации случайной величины, распределенной по закону (16) или (17), меньше 1. Заметим, что в отличие от классического экспоненциального закона распределения с одним параметром  $\lambda$  такой закон позволяет аппроксимировать произвольные распределения в случае коэффициента вариации, меньшего 1, на уровне двух первых моментов. В работе [1] получено преобразование Лапласа для функции плотности времени ожидания, который имеет вид

$$W^*(s) = s\Phi_+(s) = \frac{(1 - \lambda/\mu)(\mu + s)}{(s + \mu - \lambda)}. \quad (18)$$

Найдем первую производную от функции (18):

$$\begin{aligned} \frac{dW^*(s)}{ds} &= \\ &= \frac{(1 - \lambda/\mu)(s + \mu - \lambda) - (1 - \lambda/\mu)(\mu + s)}{(s + \mu - \lambda)^2} = \frac{-\lambda(1 - \lambda/\mu)}{(s + \mu - \lambda)^2}. \end{aligned}$$

Тогда значение первой производной со знаком минус при  $s = 0$  будет равно

$$-\left. \frac{dW^*(s)}{ds} \right|_{s=0} = \frac{\lambda(1 - \lambda/\mu)}{(\mu - \lambda)^2} = \frac{\lambda/\mu}{(\mu - \lambda)}.$$

Отсюда среднее время ожидания в очереди будет иметь вид

$$\bar{W} = \frac{\lambda/\mu}{\mu - \lambda}. \quad (19)$$

Неизвестные параметры  $\lambda$ ,  $\mu$ ,  $t_0$  распределений (16) и (17) определяются из системы уравнений по методу моментов:

$$\begin{cases} \lambda^{-1} + t_0 = \bar{\tau}_\lambda; \\ (1 + \lambda t_0)^{-1} = c_\lambda; \\ \mu^{-1} + t_0 = \bar{\tau}_\mu; \\ (1 + \mu t_0)^{-1} = c_\mu, \end{cases}$$

где числовые характеристики в правых частях системы будут входными параметрами для определения неизвестных параметров  $\lambda$ ,  $\mu$ ,  $t_0$ . Здесь приняты следующие обозначения:  $\bar{\tau}_\lambda$  — среднее значение интервалов между соседними требованиями входного потока;  $\bar{\tau}_\mu$  — среднее время обслуживания требований в системе;  $c_\lambda$  и  $c_\mu$  — коэффициенты вариаций интервала между соседними требованиями входного потока и времени обслуживания соответственно.

Алгоритмизация вышеприведенных методов расчета характеристик трех систем позволяет получить эти характеристики для широкого диапазона изменения их параметров. На рис. 1 приведена укрупненная схема алгоритма программы расчета.

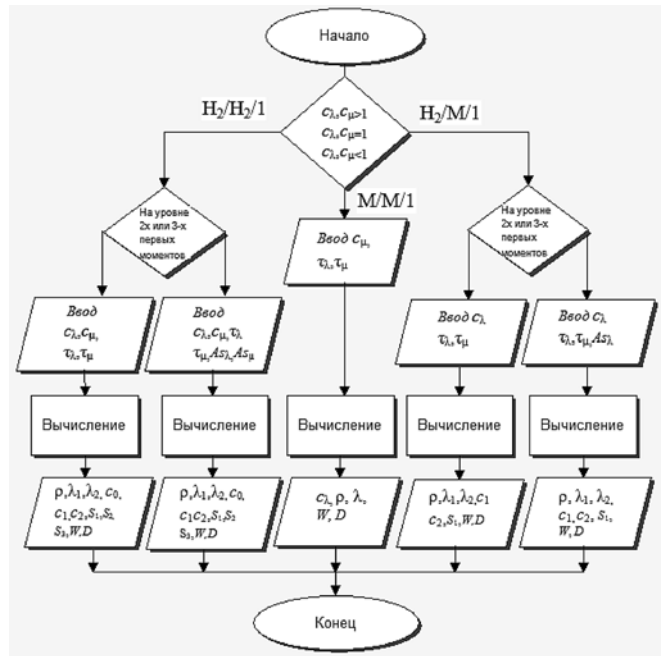


Рис. 1. Укрупненная схема алгоритма

### Примеры автоматизированного расчета характеристик рассмотренных систем

Экранная форма расчета характеристик для системы  $H_2/H_2/1$  как на уровне двух первых моментов, так и на уровне трех первых моментов, представлена на рис. 2.

Здесь приняты следующие обозначения входных параметров:  $\bar{\tau}_\lambda$  — среднее значение интервала между соседними требованиями входного потока;  $\bar{\tau}_\mu$  — среднее время обслуживания требований в системе;  $c_\lambda$  и  $c_\mu$  — коэффициенты вариаций интервала между соседними требованиями входного потока и времени обслуживания соответственно;  $A_{s_1}$  и  $A_{s_2}$  — коэффициенты асимметрии интервала между соседними требованиями входного потока и времени обслуживания.

На уровне двух первых моментов						На уровне трех первых моментов							
Вводите	$\bar{\tau}_\lambda$	$\bar{\tau}_\mu$	$c_\lambda$	$c_\mu$		Вводите	$\bar{\tau}_\lambda$	$\bar{\tau}_\mu$	$c_\lambda$	$c_\mu$	$A_{s_1}$	$A_{s_2}$	
	10/9	1	2	2			10/9	1	2	2	6	6	
Промежуточные параметры						Промежуточные параметры							
$\lambda_1$	1.5871370023	$\lambda_2$	0.2028629976	$\lambda_1$	1.5632628320	$\lambda_2$	0.1884392082	$c_0$	0.0132005432	$c_1$	0.5241919407	$c_2$	0.1957446946
$c_0$	0.0144	$c_1$	0.572	$c_2$	0.2	$s_1$	-0.8173305715	$s_2$	-0.0249793822	$s_3$	0.6465652990		
$s_1$	-0.8517267921	$s_2$	-0.0249838396	$s_3$	0.6797106318	Выходные характеристики							
Выходные характеристики						Выходные характеристики							
$\rho$	0.9	$W$	36.2000	$D$	1583.4490	$\rho$	0.9	$W$	36.1454	$D$	1583.2380		

Рис. 2. Расчет характеристик для СМО  $H_2/H_2/1$

## Заключение

В заключение можем сделать следующие выводы.

- ♦ В том случае, когда коэффициенты вариации интервалов поступления требований  $c_\lambda$  и времени обслуживания  $c_\mu$  больше 1, можно использовать систему  $H_2/H_2/1$ . Закон распределения  $H_2$  позволяет в силу своей специфики аппроксимировать распределения как на уровне двух, так и на уровне трех первых моментов. В этом случае разница в результатах аппроксимации гиперэкспоненциального распределения на уровне двух и трех моментов может получиться существенной. С точки зрения теории вероятностей, результат аппроксимации на уровне трех моментов точнее, так как три первых момента полнее характеризуют случайную величину — интервал между поступлениями или же интервал обслуживания.
- ♦ В случае коэффициента вариации интервалов поступления требований  $c_\lambda$  больше 1 и неопределенного значения коэффициента вариации времени обслуживания можно использовать систему  $H_2/M/1$ .
- ♦ Если коэффициенты вариации интервалов поступления требований  $c_\lambda$  и времени обслуживания  $c_\mu$  меньше 1, можно использовать систему  $M/M/1$  с запаздыванием во времени. Как следствие, имеем немарковскую модель массового обслуживания. Среднее время ожидания требования в очереди в такой системе меньше, чем в классической системе  $M/M/1$  при одинаковом коэффициенте загрузки. Использование функций плотности (16) и (17) позволяет аппроксимировать исходные входные распределения на уровне двух первых моментов, в отличие от классической системы, что с точки зрения теории вероятностей точнее.
- ♦ Таким образом, применение трех предложенных систем позволяет исследовать современный телетрафик в случае широкого диапазона изменения его параметров. Проведенные расчеты по разработанной программе на уровне двух моментов распределений случайных величин хорошо согласуются с данными расчетов [10], а данные расчетов на уровне трех моментов несколько превосходят данные [10].
- ♦ Исходный код программы, написанной на языке C# в среде разработки Microsoft Visual Studio, приведен в работе [11]. В программе реализованы численные методы решения систем нелинейных уравнений Ньютона и кубических уравнений Виета—Кардано. При полном тестировании программы использованы параллельные вычисления в Mathcad. Исполняемый файл занимает 61 кбайт.

Различие между значениями для среднего времени ожидания  $\bar{W}$  при аппроксимации гиперэкспоненциального закона распределения на уровне двух и трех моментов в данном случае несущественно, но относительная погрешность при других входных данных, как показывают многочисленные расчеты, может достигать 10 % [8].

На рис. 3 приведена экранная форма для расчета характеристик системы  $H_2/M/1$ . Здесь приняты аналогичные обозначения для входных параметров, что и на рис. 2.

Различие между значениями среднего времени ожидания  $\bar{W}$  для системы  $H_2/M/1$  при аппроксимации гиперэкспоненциального закона распределения на уровне двух и трех моментов в этом примере также несущественно, но относительная погрешность при других входных данных может превышать 10 %, что уже немаловажно [7].

На рис. 4 приведена экранная форма для расчета характеристик системы  $M/M/1$  с запаздыванием во времени.

Рис. 3. Расчет характеристик для СМО  $H_2/M/1$

Рис. 4. Расчет характеристик для СМО  $M/M/1$  с запаздыванием во времени

## Список литературы

1. **Тарасов В. Н., Бахарева Н. Ф., Блатов И. А.** Анализ и расчет системы массового обслуживания с запаздыванием // Автоматика и телемеханика. 2015. № 11. С. 51–59.
2. **RFC 3393** IP Packet Delay Variation Metric for IP Performance Metrics (IPPM) URL: <https://tools.ietf.org/html/rfc3393>. (дата обращения: 26.02.2015).
3. **Крылов В. В., Самохвалова С. С.** Теория телетрафика и ее приложения. Санкт Петербург: БХВ, 2005. 288 с.
4. **Бочаров П. П., Печинкин А. В.** Теория массового обслуживания. М.: Изд-во РУДН, 1995. 529 с.
5. **Mykja A.** An improved heuristic approximation for the G1/G1/1 queue with bursty arrivals // Teletraffic and datatraffic in a Period of Change, ITC-13. Elsevier Science Publishers. 1991. P. 683–688.
6. **Клейнрок Л.** Теория массового обслуживания: Пер. с англ. / Под ред. В. И. Неймана. М.: Машиностроение, 1979. 432 с.

7. **Тарасов В. Н., Бахарева Н. Ф., Горелов Г. А.** Математическая модель трафика с тяжелохвостным распределением на основе системы массового обслуживания  $H_2/M/1$  // Инфокоммуникационные технологии. 2014. Т. 12, № 3. С. 36–41.

8. **Тарасов В. Н., Картасhevский И. В.** Определение среднего времени ожидания требований в управляемой системе массового обслуживания  $H_2/H_2/1$  // Системы управления и информационные технологии. 2014. № 3. С. 92–96.

9. **Тарасов В. Н., Горелов Г. А., Ушаков Ю. А.** Восстановление моментных характеристик распределения интервалов между пакетами входящего трафика // Инфокоммуникационные технологии. 2014. № 2. С. 40–44.

10. **Кругликов В. К., Тарасов В. Н.** Анализ и расчет сетей массового обслуживания методом двумерной диффузионной аппроксимации // Автоматика и телемеханика. 1983. № 8. С. 74–83.

11. **Тарасов В. Н., Липилина Л. В.** Программный комплекс расчета характеристик систем массового обслуживания типа  $H_2/H_2/1$ ,  $H_2/M/1$  и  $M/M/1$  с запаздыванием во времени. Св-во о гос. регистрации программы для ЭВМ № 2016612657 от 03.03.2016.

**V. N. Tarasov**, Head of Chair, Professor, e-mail: tarasov-vn@psu.ru,  
**N. F. Bahareva**, Head of Chair Professor, **L. V. Lipilina**, Assistant,  
Volga State University of Telecommunications and Informatics, Samara

## Automation for Calculating Characteristics Queuing System for a Wide Range Changing Their Parameters

*The article describes the characteristics of the queuing design software systems (QS)  $G/G/1$ , which correspond to the input distribution with a coefficient of variation of the interval between the receipt of the request and service Times lowest and highest 1, that is, cover the entire range  $(0, \infty)$ . To do this, we consider the QS  $M/M/1$  with delay,  $H_2/M/1$  and  $H_2/H_2/1$  with Hyperexponential input distributions 2nd order. As the output characteristics used by the average waiting time in the queue and its variance. Other characteristics of the systems in question are derived from the average waiting time.*

*With the definition of a delay variation (jitter) in a telecommunications time-variation of its mean value, thus having the opportunity to determine the jitter through dispersion. This result is important for the analysis and description of traffic that is sensitive to delays. For receiving output characteristics the method of spectral expansion of the solution of an integral equation of Lindley is used. For the practical application of the results used the traditional method of moments.*

*It is known that distributed according to the law Hyperexponential  $H_2$  random variable has a coefficient of variation greater than 1. In [1] it is shown that the coefficient of variation of a random variable distributed according to an exponential law with a time lag of less than 1.*

*Given the fact that the distribution of  $H_2$  is a three-parameter in the software system has a mechanism of approximation, arbitrary laws Hyperexponential distributions at the level of the first two moments, and at the level of the first three moments.*

**Keywords:** queuing systems, the average waiting time in the queue, the method of spectral decomposition, the integral equation Lindley, Laplace transform

## References

1. **Tarasov V. N., Bakhareva N. F., Blatov I. A.** Analis i raschet sistemy massovogo obslugivaniya s sapasdyvaniem (Analysis and Calculation of Queuing System with Delay), *Avtomatica i telemehana*, 2015, no. 11, pp. 51–59.
2. **RFC 3393** IP Packet Delay Variation Metric for IP Performance Metrics (IPPM) URL: <https://tools.ietf.org/html/rfc3393>.
3. **Krylov V. V., Samokhvalova S. S.** *Teoriya teletrafika i ee prilozheniya*. Sankt Peterburg: BHV, 2005. 288 p.
4. **Bocharov P. P., Pechinkin A. V.** *Teoriya massovogo obslugivaniya*. Moscow: Isd-vo RUDN, 1995. 529 p.
5. **Mykja A.** An improved heuristic approximation for the G1/G1/1 queue with bursty arrivals, *Teletraffic and datatraffic in a Period of Change, ITC-13*. Elsevier Science Publishers, 1991, pp. 683–688.
6. **Kleinrock L.** *Teoriya massovogo obslugivaniya*. Per. s angl. Pod redakciey V. I. Neimana. Moscow: Mashinostroenie, 1979. — 432 p.
7. **Tarasov V. N., Bakhareva N. F., Gorelov G. A.** Matematicheskaya model trafika s tyazhelohvostnym raspredeleniem na osnove sistemy massovogo obslugivaniya  $H_2/M/1$ , *Ifocommunkazionnye tehnologii*, 2014. no. 3, pp. 36–41.

8. **Tarasov V. N., Kartashevsky I. V.** Opredelenie srednego vremeni ogidanya trebovaniy v upravlyemoy sisteme massovogo obslugivaniya  $H_2/H_2/1$  [Determination of the average waiting time requirements in a controlled queuing system], *Sistemy upravleyiya i informacionnye tehnologii*. 2014, no. 3, pp. 92–96.

9. **Tarasov V. N., Gorelov G. A., Ushakov Yu. A.** Vosstanovlenie momentnykh harakteristik raspredeleniya intervalov mezdu paketami vkhodyashogo trafika [Restoring torque characteristics of the distribution of intervals between packets of incoming traffic], *Infocommunicationnye tehnologii*, 2014, no. 2, pp. 40–44.

10. **Kruglikov V. K., Tarasov V. N.** Analis i raschet setei massovogo obslugivaniya metodom dvumernoi diffuzionnoi approksimacii [Analysis and calculation of queuing networks by method of two-dimensional diffusion approximation], *Avtomatica i telemehana*, 1983, no. 8, pp. 74–83.

11. **Tarasov V. N., Lipilina L. V.** *Programmnyi kompleks rascheta harakteristik sistem massovogo obslugivaniya tipa  $H_2/H_2/1$ ,  $H_2/M/1$  i  $M/M/1$  s sapasduvaniem vo vremeni* [Program complex of calculation of characteristics of queuing systems like  $H_2/H_2/1$ ,  $H_2/M/1$  and  $M/M/1$  with delay in time]. Certificate on the state registration of the computer program no. 2016612657 from 03.03.2016.

# Указатель статей, опубликованных в журнале "Информационные технологии" в 2016 году

## ОБЩИЕ ВОПРОСЫ

**Богатырев П. Ю., Коськин А. В.** Структура информационного пространства организации с применением теоретико-множественного подхода к систематизации информации № 10

## ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

- Ахмедов М. З.** Решение задачи прогнозирования с применением нечеткой импликации . . . . . № 2
- Барский А. Б.** Моделирование индуктивного мышления с помощью языка логического вывода ПРОЛОГ . . . . . № 9
- Бугаков С. В., Мурзинцев С. В., Цхай А. А.** Использование горизонтально масштабируемой инфраструктуры при поиске заимствований в тексте . . . . . № 8
- Ву Вьет Тханг.** Ускорение алгоритма кластеризации DBSCAN за счет использования алгоритма K-Means . . . . . № 2
- Домников А. С., Белоус В. В.** Синтез рационального упорядочения учебных модулей методом Бержа . . . . . № 12
- Зак Ю. А.** Алгоритмы решения транспортной задачи в условиях нечетких данных о стоимости поставки грузов . . . . . № 8
- Кулагин В. П., Дубинин В. Н.** Структурный анализ сетей Петри . . . . . № 1
- Мальгин И. Г., Комашинский В. И.** Информационные технологии и искусственный интеллект — основные двигатели четвертой индустриальной революции (Industrie 4.0). № 12
- Маренко В. А.** Применение когнитивного моделирования для исследования межэтнических отношений . . . . . № 9
- Мохов А. С., Толчеев В. О.** Способы учета структуры научных документов в задачах обработки и анализа текстовой информации . . . . . № 5
- Николаев А. И.** Эффективный подход на основе машинного обучения к решению задачи о максимальной клике . . . . . № 4
- Разоренов А. А., Фомичев В. А.** Компактная формализация входных и промежуточных данных алгоритмов семантического анализа предписаний . . . . . № 12
- Родькина О. Я., Никольская В. А.** К проблеме распознавания психоэмоционального состояния человека по речи с использованием автоматизированных систем . . . . . № 10
- Саак А. Э.** Диспетчеризация массивов заявок кругового и гиперболического типа в Grid-системах . . . . . № 5
- Саак А. Э.** Кольцевые алгоритмы диспетчеризации массивами заявок в Grid-системах . . . . . № 3
- Тарасов А. Д.** Эффективность работы генетического алгоритма в задаче проектирования систем физической защиты . № 4
- Черняховская Л. Р., Галиуллина А. Ф., Сабитов И. И.** Оценка эффективности предоставления государственных услуг с использованием системы нечеткого логического вывода № 8

## МОДЕЛИРОВАНИЕ И ОПТИМИЗАЦИЯ

- Барский А. Б., Желенков Б. В.** Средства оптимизации информационного взаимодействия ресурсных процессоров для минимизации времени "облачных" вычислений . . . . . № 1
- Бронштейн Е. М., Давлетбаев А. А.** Задача маршрутизации с учетом зависимости стоимости транспортировки от погодных условий . . . . . № 12
- Валеева А. Ф., Гончарова Ю. А., Валеев Р. С.** Об одном подходе к решению задач операционного планирования по доставке однородной продукции различным клиентам. Часть 1. . . . . № 10
- Гейдаров П. Ш.** Алгоритм кратчайшего маршрута на основе выделенного набора маршрутов . . . . . № 9
- Дмитриев А. В., Мальцева С. В., Цуканова О. А.** Моделирование и качественный анализ социальной микроблогинговой сети как динамической системы . . . . . № 4
- Доморацкий Е. П.** Алгоритм синтеза геометрических характеристик проекционных изображений трехмерных объектов № 8
- Дягилев В. И., Коковин В. А., Увайсов С. У., Увайсова С. С.** Компьютерное моделирование работы силового преобразователя с выходным синусоидальным напряжением . . . . . № 4

- Ефлов В. Б., Питухин Е. А., Стафеев Ф. В.** Вопросы производительности при выполнении запросов в реляционной СУБД H2 . . . . . № 12
- Зак Ю. А.** Математические модели эффективного подбора персонала для выполнения инновационных проектов . . . . . № 10
- Зеленко Г. В., Иванников А. Д., Коган Д. Ю., Рошин А. В.** Проблемы функциональной устойчивости сенсорной локальной сети . . . . . № 7
- Инютин С. А.** Метод вычисления количественной характеристики модулярной величины . . . . . № 5
- Исаев Д. В.** Имитационное моделирование сложных проектов № 8
- Кухаренко Б. Г., Солнцева-Чалей М. О.** Моделирование центров для пучков многомерных траекторий . . . . . № 2
- Левин В. И.** Модели производной и их применение для моделирования динамических неполностью определенных систем . . . . . № 8
- Левин В. И.** Непрерывно-логическая модель решения комбинаторных задач . . . . . № 11
- Львович И. Я., Львович Я. Е., Преображенский А. П., Чопоров О. Н.** Особенности методов вычислительной гидродинамики для моделирования турбулентности . . . . . № 12
- Мандрикова О. В., Фетисова (Глушкова) Н. В., Полозов Ю. А.** Моделирование временного хода параметров ионосферы и выделение аномалий . . . . . № 1
- Матвеев М. Г., Михайлов В. В., Сирота Е. А.** Комбинированная прогностическая модель нестационарного многомерно-временного ряда для построения пространственного профиля атмосферной температуры . . . . . № 2
- Мельников Г. А., Мельников Т. А., [Убарев В. В.]** Алгоритмы упрощения деревьев регрессии: обзор и эмпирическое сравнение . . . . . № 9
- Моисеев А. А.** Критериальное моделирование в виртуализации квалификационных испытаний . . . . . № 8
- Мухутдинов А. Р., Вахидова З. Р., Ефимов М. Г.** Компьютерное моделирование бризантного действия взрыва . . . . . № 5
- Новикова Н. М., Борискин А. В.** Математическая модель поиска влиятельных объектов социальной сети на основе априорной информации . . . . . № 2
- Орехов Э. Ю., Орехов Ю. В.** Сравнение и оценка качества эвристических алгоритмов на конечной массовой задаче в условиях неравновероятной генерации тестовых примеров . . . . . № 12
- Ошкало Д. В.** Разработка процессов трансформации моделей с помощью биграфов . . . . . № 3
- Светушков Н. Н.** Применение интегрального метода струн для расчета тепловых нагрузок на охлаждаемые стенки сопловой части ракетного двигателя . . . . . № 1
- Светушков Н. Н.** Упрощенная математическая модель расчета температурных полей при детонационном горении . . . . . № 9
- Струченков В. И.** Динамическое программирование в задачах планирования реализации частично возобновляемых ресурсов . . . . . № 2
- Струченков В. И.** Динамическое программирование с использованием множеств Парето в задачах планирования реализации возобновляемых ресурсов . . . . . № 3
- Федосов В. В., Федосова А. В.** О параметризации сетки в алгоритмах полубесконечной оптимизации для задач промышленной экологии . . . . . № 7
- Филиппова А. С., Дямина Э. И., Валиахметова Ю. И.** Метод ограниченной декомпозиции для решения комплексной задачи геометрического покрытия и раскроя . . . . . № 3

## ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ И СЕТИ

- Богатырев В. А., Богатырев А. В.** Модель резервированного обслуживания запросов реального времени в компьютерном кластере . . . . . № 5
- Богатырев В. А., Богатырев А. В.** Надежность функционирования кластерных систем реального времени с фрагментацией и резервированным обслуживанием запросов . . . . . № 6
- Бритвин Н. В., Карпунин Е. О.** Разработка имитационной модели атаки "человек посередине" для исследования эффективности протоколов информационного взаимодействия . . . . . № 6
- Дворников А. А., Восков Л. С., Саксонов Е. А., Ефремов С. Г.** Метод построения оптимального наложенного канала для беспроводной сенсорной сети . . . . . № 11
- Микова С. Ю., Оладько В. С.** Сравнение алгоритмов выявления сетевых аномалий с помощью меры Ван Ризбергера . . . . . № 5

- Неустроев М. Ю.** Анализ показателей эффективности и скорости обслуживания в центрах обработки вызовов . . . № 6
- Печенко И. С.** Способы представления спецификаций вычислительных систем: проблемы и возможности машинной обработки . . . № 9
- Поршнев С. В., Божалкин Д. А., Копосов А. С.** Опыт использования суперкомпьютера для обработки дампов сетевого трафика магистрального интернет-канала . . . № 1
- Романов А. Ю.** Исследование сетей на кристалле с топологией *mesh* с помощью модели *NoCTweak* . . . № 7
- Саак А. Э.** Диспетчеризация заявок кругового типа в Grid-системах . . . № 1
- Саак А. Э.** О качестве диспетчеризации массивов точных форм с заявками параболического типа в Grid-системах . . . № 8
- Тарасов В. Н., Бахарева Н. Ф., Липилина Л. В.** Математическая модель телетрафика на основе системы G/M/1 и результаты вычислительных экспериментов . . . № 2
- Холопов Ю. А., Ле Ба Чунг, Нгуен Тхань Чунг, Чан Ван Хань.** Особенности реализации распределенных систем управления на основе аппаратных сетевых решений . . . № 7

## СИСТЕМЫ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ

- Амирханов А. В., Гладких А. А., Глушко А. А., Макаруч В. В., Шахнов В. А.** Автоматизированная система формирования топологической информации для изготовления комплекта фотошаблонов сверхбольших интегральных схем . . . № 2
- Бибило П. Н., Логинова И. П.** Формирование энергоемких тестов для комбинационных логических схем по результатам оценки их энергопотребления . . . № 4
- Гридин В. Н., Анисимов В. И., Дмитриевич Г. Д., Ларистов А. И., Аль-Шамери Я. М.** Организация информационного обеспечения web-ориентированных схемотехнических САПР . . . № 2
- Загидуллин Р. Р.** Автоматизация разработки альтернативных технологических процессов . . . № 3
- Иванников А. Д.** Анализ методов декомпозиции задачи отладки проектов цифровых систем . . . № 10
- Ильясов Б. Г., Гвоздев В. Е., Бежаева О. Я., Курунова Р. Р.** Сравнительный анализ проектных решений по комплексу потребительских свойств и технических характеристик объектов . . . № 10
- Кузнецова Е. В., Кравченко Т. К., Кряжев С. А.** Комплексная корпоративная система управления проектами в строительном бизнесе . . . № 10
- Олейник П. П., Гурьянов В. И.** UML-профиль для метамодельно-ориентированного проектирования программных приложений баз данных . . . № 4
- Петров Ю. И.** Система поддержки принятия решений по созданию конкурентоспособной технической сложной продукции . . . № 6
- Стемпковский А. Л., Тельпухов Д. В., Соловьев Р. А.** Повышение сбоеустойчивости логических схем на основе частичного ресинтеза схемы . . . № 7
- Сурков А. В.** Использование статического временного анализа для проектирования самосинхронных схем . . . № 2
- Яковлев А. А., Камаев В. А., Сорокин В. С., Мишустина С. Н.** Поискное конструирование систем охлаждения на основе инженерно-физического подхода . . . № 11

## ЦИФРОВАЯ ОБРАБОТКА СИГНАЛОВ И ИЗОБРАЖЕНИЙ

- Бимаков Е. В.** Метод виртуального зондирования сцены на базе воксельной технологии обработки сцен . . . № 11
- Гаврилов С. В., Гулов С. И., Жукова Т. Д., Рыжова Д. И.** Применение теории кодирования для повышения помехозащищенности комбинационных схем . . . № 12
- Гиацинтов А. М., Мамросенко К. А.** Методы отображения трехмерных объектов при применении отложенной визуализации . . . № 7
- Дворников С. С., Дворников С. В.** Формирование сигналов с непрерывной фазой для передачи дискретной информации . . . № 6
- Кухаренко Б. Г., Солнцева-Чалей М. О.** Использование методов выделения ярких областей при сегментировании телеметрических изображений для идентификации групп объектов . . . № 7
- Лютюкова Л. А., Шматова Е. В.** Анализ и синтез алгоритмов распознавания образов с использованием переменного-значной логики . . . № 4

- Мальцев А. В., Михайлюк М. В., Тимохин П. Ю., Торгашев М. А.** Технология разделения видеопотоков в дистанционном обучении с помощью нескольких устройств Kinect . . . № 9
- Мамедов Р. К., Алиев Т. Ч., Муталлимова А. С.** Распознавание бинарных изображений объектов, подвергнутых аффинным преобразованиям . . . № 10
- Наумов А. С.** Метод автоматизированной реконструкции фрагментированных изображений на основе совокупности признаков фрагментов . . . № 1
- Поперечный П. С.** Применение цифровой фильтрации для реализации кодера Рида—Соломона . . . № 3
- Поперечный П. С.** Сравнительная оценка корректирующей способности кодов Рида—Соломона и Боуза—Чоудхури—Хоквингема . . . № 6
- Садыхов С. С., Кульков Я. Ю.** Распознавание отдельных тестовых и реальных плоских объектов по безразмерным признакам контуров их бинарных изображений . . . № 10
- Филимонов А. Б., Фам Фьюнг Кыонг.** Динамическое распознавание воздушных целей по радиолокационным дальностным портретам . . . № 6

## БЕЗОПАСНОСТЬ ИНФОРМАЦИИ

- Авдошин С. М., Лазаренко А. В.** Методы деанонимизации пользователей Tor . . . № 5
- Авдошин С. М., Лазаренко А. В.** Технология анонимных сетей . . . № 4
- Асратян Р. Э., Лебедев В. Н., Орлов В. Л.** Эффективность защищенных каналов взаимодействия на основе применения протокола HTTPS в прокси-серверах . . . № 7
- Грибова В. В., Иванова А. В.** Концепция программного комплекса для управления безопасностью информационных систем № 7
- Димов Э. М., Маслов О. Н., Раков А. С.** Управление информационной безопасностью корпорации с применением критериев риска и ожидаемой полезности . . . № 8
- Еременко А. В., Сулавко А. Е., Волков Д. А.** Современное состояние и пути модернизации преобразователей биометрия-код . . . № 3
- Еременко А. В., Сулавко А. Е., Толкачева Е. В., Левитская Е. А.** Метод защиты текстовых документов на электронных и бумажных носителях на основе скрытого биометрического идентификатора субъекта, получаемого из подписи . . . № 8
- Машкина И. В., Сенцова А. Ю.** Обеспечение информационной безопасности системы облачных вычислений . . . № 11
- Червяков Н. И., Дерябин М. А.** Новый метод порогового разделения секрета, основанный на системе остаточных классов № 3
- Щеглов К. А., Щеглов А. Ю.** Вопросы корректности и универсальности подхода к моделированию надежностных параметров и характеристик угроз атак . . . № 11
- Щеглов К. А., Щеглов А. Ю.** Интерпретация и моделирование угрозы атаки на информационную систему. Часть 2. Моделирование угрозы атаки . . . № 1

## ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ОБРАЗОВАНИИ

- Бершадский А. М., Бождай А. С., Мкртчян В. С.** Принципы построения общедоступной самоадаптирующейся системы дистанционного обучения на основе модели изменчивости и сервис-ориентированной архитектуры . . . № 2
- Вишнеков А. В., Иванова Е. М.** Автоматизация выбора траектории обучения в среде Smart-образования . . . № 9
- Груздева Л. М., Лобачев С. Л.** Набор на дистанционное обучение в РУНЕТ: предпосылки, опыт и анализ . . . № 10
- Подбельский В. В., Максименкова О. В., Бабич К. С.** Об обеспечении интероперабельности программных средств поддержки тестирования в образовании . . . № 7

## ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В БИОМЕДИЦИНСКИХ СИСТЕМАХ

- Абдуллаев Н. Т., Гасанкулиева М. М.** Информативные параметры и флуктуационная динамика электрокардиограмм при фликкер-шумовой спектроскопии . . . № 1
- Абдуллаев Н. Т., Мамедов Н. Я., Агаева Г. С.** Применение метода спектрального анализа для дифференциальной диагностики заболевания органов желудочно-кишечного тракта . . . № 9
- Авдошин С. М., Песоцкая Е. Ю.** Экосистемы мобильной медицины . . . № 10

- Агеева У. О., Агеева В. Г., Барский А. Б.** Бионическое интеллектуальное протезирование конечностей и логические нейронные сети . . . . . № 5
- Знайко Г. Г., Прохоров Н. Л., Красовский В. Е.** Опыт проектирования и внедрения компьютеризированной ультразвуковой медицинской техники . . . . . № 10
- Кухтичев А. А., Кленов Е. А., Скородумов С. В.** Разработка архитектуры информационной системы "ЦифроМед" цифровой медицины в авиации и космонавтике. . . . . № 2
- Полосин В. Г., Бодин О. Н.** Энтропийно-параметрический критерий проверки адекватности модели распределения ионных токов миокарда . . . . . № 1
- Прасолова А. Е.** Коллективный нейросетевой алгоритм диагностики инфаркта миокарда . . . . . № 5

#### БАЗЫ ДАННЫХ

- Алыгулиев Р. М., Алекперова И. Я.** Проблема BIG DATA в нефтегазовой отрасли: состояние и перспективы . . . . . № 11
- Жураковский В. Н., Силин С. И.** Выбор алгоритмов обработки данных в современной вычислительной технике на основе системного анализа . . . . . № 3
- Коровин А. С., Скирневский И. П.** Система динамической визуализации больших массивов данных сложных физических экспериментов . . . . . № 5
- Кузнецова М. В., Стрижов В. В.** Локальное прогнозирование временных рядов с использованием инвариантных преобразований . . . . . № 6
- Родионов А. Н.** Системная интерпретация ключевых категорий концептуальных моделей баз данных . . . . . № 6
- Чернышов А. В.** К вопросу о применении оптических дисков для создания долговременных электронных архивных хранилищ информации небольших организаций . . . . . № 8
- Четырбоцкий А. Н.** Автоматическая классификация объектов многомерной выборки рекурсивными методами построения кривых Пеано (на примере выборки Р. Фишера) . . . . . № 6

#### ГЕОИНФОРМАЦИОННЫЕ СИСТЕМЫ

- Беляков С. Л., Белякова М. Л., Глушков А. А.** Метапреобразования образов при поиске достоверных решений в интеллектуальных геоинформационных системах . . . . . № 11

#### ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

##### В ОРГАНИЗАЦИОННЫХ И СОЦИАЛЬНО-ЭКОНОМИЧЕСКИХ СИСТЕМАХ

- Зуев А. С., Фадеев И. С.** Виртуальные ситуационные центры — новый инструмент управления социально-экономическими системами . . . . . № 3
- Мамедова М. Г., Джабраилова З. Г.** Методологический подход к многокритериальному принятию решений в задачах управления человеческими ресурсами . . . . . № 6

#### СТРУКТУРНЫЙ СИНТЕЗ И ПРИНЯТИЕ РЕШЕНИЙ

- Захаров А. В., Рамазанова Р. Р., Курилова И. С.** Управление системой, состоящей из набора типовых подсистем, с помощью системы показателей . . . . . № 2

#### СИСТЕМЫ СВЯЗИ И ТЕЛЕКОММУНИКАЦИИ

- Соловьев Б. И., Жусов Д. Л., Просолупов С. А.** Определение мест размещения сетевых ресурсов на квазиоднородной топологической структуре инфокоммуникационной системы . № 12

#### ПРОГРАММНАЯ ИНЖЕНЕРИЯ

- Решетова Е. Н.** Универсальные компетенции студента образовательного направления "Программная инженерия" . № 12

#### ПРИКЛАДНЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

- Гриняк В. М., Иваненко Ю. С., Девятисильный А. С.** Погрешность измерения координат компьютеризированными РЛС, обусловленная скоростью передачи данных в распределенных информационных системах . . . . . № 5
- Савченко В. В.** Распознавание речевых команд методом фонетического декодирования слов с подавлением фонового шума . . . . . № 1
- Скоробогатов Р. Ю.** Расширение интерактивности компьютерной модели в телевизионной среде . . . . . № 5
- Тарасов В. Н., Липилина Л. В., Бахарева Н. Ф.** Автоматизация расчета характеристик систем массового обслуживания для широкого диапазона изменения их параметров. . . . . № 12

#### ДИСКУССИОННЫЙ КЛУБ

- Вяткин В. Б.** Интегративно-кодовая взаимосвязь комбинаторного, вероятностного и синергетического подходов к определению количества информации . . . . . № 7
- Голубев В. И.** Информация как отображение объектов окружающего мира в коре головного мозга . . . . . № 3
- Цветков В. Я.** Полисемия информации . . . . . № 4

#### Журнал в журнале

#### НЕЙРОСЕТЕВЫЕ ТЕХНОЛОГИИ

- Аведьян Э. Д., Ле Т. Ч. Л.** Двухуровневая система обнаружения DoS-атак и их компонентов на основе нейронных сетей СМАС . . . . . № 9
- Галушкин А. И.** Новые технологии микроэлектроники и разработки перспективных нейрокомпьютеров . . . . . № 7
- Галушкин А. И., Пантюхин Д. В.** СуперЭВМ и мемристоры . № 4
- Дам В. Н.** Автоматическое распознавание цифровой модуляции радиосигналов с помощью многослойной нейронной сети по кумулянтным признакам . . . . . № 7
- Задаянчук А. И., Попова М. С., Стрижов В. В.** Выбор оптимальной модели классификации временных рядов . . . № 4
- Кулагин В. П., Иванов А. И., Серикова Ю. И.** Корректировка методических и случайных составляющих погрешностей вычисления коэффициентов корреляции, возникающих на малых выборках биометрических данных . . . . . № 9
- Кургалин С. Д., Туровский Я. А., Борзунов С. В., Адаменко А. А.** Теоретические аспекты оптимизации эволюционного обучения нейронных сетей с использованием "изолятов" . . . . № 11
- Пекунов В. В.** Компактное описание вариативных полей физических величин в повторяющихся задачах моделирования атмосферного переноса загрязнителей . . . . . № 11

#### Адрес редакции:

107076, Москва, Стромьинский пер., 4

Телефон редакции журнала (499) 269-5510

E-mail: it@novtex.ru

Технический редактор *Е. В. Конова.*

Корректор *З. В. Наумова.*

Сдано в набор 06.10.2016. Подписано в печать 22.11.2016. Формат 60×88 1/8. Бумага офсетная.

Усл. печ. л. 8,86. Заказ IT1216. 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 "Авансед солюшнз". Отпечатано в ООО "Авансед солюшнз".

119071, г. Москва, Ленинский пр-т, д. 19, стр. 1.