



НАУЧНО-ТЕХНИЧЕСКИЕ ВЕДОМОСТИ

САНКТ-ПЕТЕРБУРГСКОГО ГОСУДАРСТВЕННОГО
ПОЛИТЕХНИЧЕСКОГО УНИВЕРСИТЕТА

Информатика. Телекоммуникации.
Управление

Том 12, № 2
2019



ПОЛИТЕХ-ПРЕСС
Санкт-Петербургский
политехнический университет
Петра Великого

Санкт-Петербург
2019

НАУЧНО-ТЕХНИЧЕСКИЕ ВЕДОМОСТИ САНКТ-ПЕТЕРБУРГСКОГО
ГОСУДАРСТВЕННОГО ПОЛИТЕХНИЧЕСКОГО УНИВЕРСИТЕТА
ИНФОРМАТИКА. ТЕЛЕКОММУНИКАЦИИ. УПРАВЛЕНИЕ

РЕДАКЦИОННЫЙ СОВЕТ ЖУРНАЛА

Председатель

Юсупов Р.М., чл.-кор. РАН;

Редакционный совет:

Абрамов С.М., чл.-кор. РАН;

Арсеньев Д.Г., д-р техн. наук, профессор;

Воеводин В.В., чл.-кор. РАН;

Заборовский В.С., д-р техн. наук, профессор;

Козлов В.Н., д-р техн. наук, профессор;

Фотиади А.Э., д-р физ.-мат. наук, профессор;

Черноруцкий И.Г., д-р техн. наук, профессор.

РЕДАКЦИОННАЯ КОЛЛЕГИЯ ЖУРНАЛА

Главный редактор

Коротков А.С., д-р техн. наук, профессор, Санкт-Петербургский политехнический университет Петра Великого, Россия;

Редакционная коллегия:

Ицыксон В.М., канд. техн. наук, доцент, Санкт-Петербургский политехнический университет Петра Великого, Россия;

Prof. Dr. *Philippe Ferrari*, Head of the RF and Millimeter-Wave Lab IMEP-LAHC Microelectronics, Electromagnetism and Photonic Institute, Grenoble Alpes University, France;

Prof. Dr. *Wolfgang Krautschneider*, Head of Nanoelectronics Institute, Hamburg University of Technology, Germany;

Кучерявый Е.А., канд. техн. наук, профессор, Tampere University of Technology, Finland.

Prof. Dr. *Fa-Long Luo*, Affiliate Full Professor University of Washington, USA, Chief Scientist Micron Technology, Inc., Milpitas, USA, Chairman IEEE SPS Industry DSP Technology Standing Committee;

Макаров С.Б., д-р техн. наук, профессор, Санкт-Петербургский политехнический университет Петра Великого, Россия;

Prof. Dr. *Emil Novakov*, IMEP-LAHC Microelectronics, Electromagnetism and Photonic Institute, Grenoble, France;

Прокопенко Н.Н., д-р техн. наук, профессор, Донской государственный технический университет, г. Ростов-на-Дону, Россия;

Путря М.Г., д-р техн. наук, профессор, Национальный исследовательский университет «Московский институт электронной техники», Москва, Россия;

Пышкин Е.В., канд. техн. наук, доцент, School of Computer Science and Engineering, University of Aizu, Japan;

Трифонов П.В., д-р техн. наук, доцент, Санкт-Петербургский политехнический университет Петра Великого, Россия;

Устинов С.М., д-р техн. наук, профессор, Санкт-Петербургский политехнический университет Петра Великого, Россия;

Уткин Л.В., д-р техн. наук, профессор, Санкт-Петербургский политехнический университет Петра Великого, Россия;

Цикин И.А., д-р техн. наук, профессор, Санкт-Петербургский политехнический университет Петра Великого, Россия;

Шкодывев В.П., д-р техн. наук, профессор, Санкт-Петербургский политехнический университет Петра Великого, Россия.

Журнал с 2002 года входит в Перечень ведущих рецензируемых научных журналов и изданий, в которых должны быть опубликованы основные результаты диссертаций на соискание ученой степени доктора и кандидата наук.

Сведения о публикациях представлены в Реферативном журнале ВИНТИ РАН, в международной справочной системе «Ulrich`s Periodical Directory», в базах данных Российский индекс научного цитирования (РИНЦ), Google Scholar, EBSCO, Math-Net.Ru, ProQuest, Index Copernicus

Журнал зарегистрирован Федеральной службой по надзору в сфере информационных технологий и массовых коммуникаций (Роскомнадзор). Свидетельство о регистрации ПИ № ФС77-51457 от 19.10.2012 г.

При перепечатке материалов ссылка на журнал обязательна.

Точка зрения редакции может не совпадать с мнением авторов статей.

Адрес редакции и издательства: Россия, 195251, Санкт-Петербург, ул. Политехническая, д. 29.
Тел. редакции (812) 552-62-16.

Подписной индекс **47517** в объединенном каталоге «Пресса России».

Санкт-Петербургский политехнический университет Петра Великого, 2019

MINISTRY OF SCIENCE AND HIGHER EDUCATION OF THE RUSSIAN FEDERATION



**ST. PETERSBURG STATE
POLYTECHNICAL UNIVERSITY
JOURNAL**

Computer Science.
Telecommunications and Control Systems

**Vol. 12, No. 2
2019**



POLYTECH-PRESS
Peter the Great
St.Petersburg Polytechnic
University

Saint Petersburg
2019

ST. PETERSBURG STATE POLYTECHNICAL UNIVERSITY JOURNAL COMPUTER SCIENCE. TELECOMMUNICATIONS AND CONTROL SYSTEMS

EDITORIAL COUNCIL

Head of the editorial council

Prof. Dr. *Rafael M. Yusupov* (corresponding member of the Russian Academy of Sciences)

Members:

Prof. Dr. *Sergey M. Abramov* (corresponding member of the Russian Academy of Sciences),

Prof. Dr. *Dmitry G. Arseniev*,

Prof. Dr. *Vladimir V. Voevodin* (corresponding member of the Russian Academy of Sciences),

Prof. Dr. *Vladimir S. Zaborovsky*,

Prof. Dr. *Vladimir N. Kozlov*,

Prof. Dr. *Alexandr E. Fotiadi*,

Prof. Dr. *Igor G. Chernorutsky*.

EDITORIAL BOARD

Editor-in-chief

Prof. Dr. *Alexander S. Korotkov*, Peter the Great St. Petersburg Polytechnic University, Russia;

Members:

Assoc. Prof. Dr. *Vladimir M. Itsykson*, Peter the Great St. Petersburg Polytechnic University, Russia;

Prof. Dr. *Philippe Ferrari*, Head of the RF and Millimeter-Wave Lab IMEP-LAHC Microelectronics, Electromagnetism and Photonic Institute, Grenoble Alpes University, France;

Prof. Dr. *Yevgeni Koucheryavy*, Tampere University of Technology, Finland.

Prof. Dr. *Wolfgang Krautschneider*, Head of Nanoelectronics Institute, Hamburg University of Technology, Germany;

Prof. Dr. *Fa-Long Luo*, Affiliate Full Professor University of Washington, USA, Chief Scientist Micron Technology, Inc., Milpitas, USA, Chairman IEEE SPS Industry DSP Technology Standing Committee;

Prof. Dr. *Sergey B. Makarov*, Peter the Great St. Petersburg Polytechnic University, Russia;

Prof. Dr. *Emil Novakov*, IMEP-LAHC Microelectronics, Electromagnetism and Photonic Institute, Grenoble, France;

Prof. Dr. *Nikolay N. Prokopenko*, Don State Technical University, Rostov-on-Don, Russia;

Prof. Dr. *Mikhail G. Putrya*, National Research University of Electronic Technology, Moscow, Russia;

Sen. Assoc. Prof. Dr. *Evgeny Pyshkin*, School of Computer Science and Engineering, University of Aizu, Japan;

Prof. Dr. *Viacheslav P. Shkodyrev*, Peter the Great St. Petersburg Polytechnic University, Russia;

Prof. Dr. *Peter V. Trifonov*, Peter the Great St. Petersburg Polytechnic University, Russia;

Prof. Dr. *Igor A. Tsikin*, Professor, Peter the Great St. Petersburg Polytechnic University, Russia;

Prof. Dr. *Sergey M. Ustinov*, Peter the Great St. Petersburg Polytechnic University, Russia;

Prof. Dr. *Lev V. Utkin*, Peter the Great St. Petersburg Polytechnic University, Russia.

The journal is included in the List of Leading Peer-Reviewed Scientific Journals and other editions to publish major findings of PhD theses for the research degrees of Doctor of Sciences and Candidate of Sciences.

The journal is indexed by Ulrich's Periodicals Directory, Google Scholar, EBSCO, ProQuest, Index Copernicus, VINITI RAS Abstract Journal (Referativnyi Zhurnal), VINITI RAS Scientific and Technical Literature Collection, Russian Science Citation Index (RSCI) database © Scientific Electronic Library and Math-Net.ru databases.

The journal is registered with the Federal Service for Supervision in the Sphere of Telecom, Information Technologies and Mass Communications (ROSKOMNADZOR). Certificate ПИ № ФС77-51457 issued Oct. 19, 2012.

No part of this publication may be reproduced without clear reference to the source.

The views of the authors can contradict the views of the Editorial Board.

The address: 195251 Polytekhnicheskaya Str. 29, St. Petersburg, Russia.

Содержание

Информационные технологии

Лазарев В.Л. Информационное шкалирование вариаций законов распределения параметров в приложениях к задачам мониторинга и управления	7
Ермолова Т.К., Иващенко П.Д., Лаптев В.В. Изучение эффективности визуализации статических структур данных с помощью брусковых и секторных диаграмм методом ай-трекинга	16
Ядгарова Ю.В., Таратухин В.В. Методика выбора шаблона программной архитектуры на основе анализа параметров качества системы	28

Программное обеспечение вычислительных, телекоммуникационных и управляющих систем

Костенко Д.А., Онуфриев В.А., Шкодырев В.П. Многокритериальная оптимизация ректификационного процесса по алгоритму SPEA2	39
Попов С.Г., Пурий А.А. Исследование алгоритмов декомпозиции и выполнения запросов на выборку в гетерогенных системах управления реляционными базами данных	50

Моделирование вычислительных, телекоммуникационных, управляющих и социально-экономических систем

Ботнев В.А., Устинов С.М. Методика определения расстояния от точки до отрезка в задачах навигации	68
--	----

Contents

Information Technologies

- Lazarev V.L.** Informational scaling of variations in the distribution laws of parameters. Applications to the tasks of monitoring and management 7
- Ermolova T.K., Ivaschenko P.D., Laptev V.V.** Studying the efficiency of visualization of static data structures by means of bar and pie charts by eye-tracking method 16
- Yadgarova Yu.V., Taratukhin V.V.** General approach for selection software architecture pattern based on system quality attributes 28

Software of Computer, Telecommunications and Control Systems

- Kostenko D.A., Onufriev V.A., Shkodyrev V.P.** Multicriterial optimisation of rectificational process based on SPEA2 algorithm 39
- Popov S.G., Purii A.A.** Study decomposing and executing algorithms of data extraction queries in heterogeneous relationship database management systems 50

Computer Simulations of Telecommunication, Control, and Social Systems

- Botnev V.A., Ustinov S.M.** Distance finding method between a point and a segment in navigation 68

DOI: 10.18721/JCSTCS.12201
УДК 53.08

ИНФОРМАЦИОННОЕ ШКАЛИРОВАНИЕ ВАРИАЦИЙ ЗАКОНОВ РАСПРЕДЕЛЕНИЯ ПАРАМЕТРОВ В ПРИЛОЖЕНИЯХ К ЗАДАЧАМ МОНИТОРИНГА И УПРАВЛЕНИЯ

В.Л. Лазарев

Санкт-Петербургский национальный исследовательский университет
информационных технологий, механики и оптики,
Санкт-Петербург, Российская Федерация

Исследовано влияние изменений закона распределения на состояние неопределенности параметра на основе информационных оценок. Предложена методология по созданию информационных шкал для оперативной оценки изменения этих состояний. Для иллюстрации предлагаемого подхода рассмотрен пример информационной шкалы, основанной на базе пяти типовых законов распределения, получивших широкое распространение в различных сферах. В результате использования предложенных решений представляется возможным получать оценки трансформации законов распределений на основе порождаемого при этом количества информации. Полученные результаты являются наглядными, а предложенные методы и технологии достаточно «простыми» и удобными для практического использования. Реализация подхода осуществлена на основе методов и разработок теории энтропийных потенциалов, имеет перспективы применения для организации мониторинга и управления.

Ключевые слова: законы распределения, состояния неопределенности, энтропийные потенциалы, информационное шкалирование, мониторинг.

Ссылка при цитировании: Лазарев В.Л. Информационное шкалирование вариаций законов распределения параметров в приложениях к задачам мониторинга и управления // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2019. Т. 12. № 2. С. 7–15. DOI: 10.18721/JCSTCS.12201

INFORMATIONAL SCALING OF VARIATIONS IN THE DISTRIBUTION LAWS OF PARAMETERS. APPLICATIONS TO THE TASKS OF MONITORING AND MANAGEMENT

V.L. Lazarev

St. Petersburg National Research University
of Information Technologies, Mechanics and Optics,
St. Petersburg, Russian Federation

The paper studies the effect of changes in the distribution law on the parameter uncertainty state based on informational estimates. The proposed methodology for creating information scales for the rapid assessment of changes in these states. To

illustrate the proposed approach, we consider an example of an information scale based on the basis of five model distribution laws that are widely used in various fields. As a result of using the proposed solutions, it is possible to obtain estimates of the transformation of the laws of distributions based on the amount of information generated in this case. The results obtained are illustrative, and the proposed methods and technologies are fairly «simple» and convenient for practical use. The implementation of the approach was carried out on the basis of the methods and developments of the theory of entropy potentials; it has prospects of application for the organization of monitoring and control.

Keywords: laws of distribution, states of uncertainty, entropy potentials, information scaling, monitoring.

Citation: Lazarev V.L. Informational scaling of variations in the distribution laws of parameters. Applications to the tasks of monitoring and management. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2019, Vol. 12, No. 2, Pp. 7–15. DOI: 10.18721/JCSTCS.12201

Введение

Эволюция закона распределения какого-либо параметра объекта является одной из составляющих процесса изменения его состояния неопределенности. Природа эволюции может быть обусловлена различными причинами: действием естественных возмущений, внесением управляющих воздействий по различным каналам и др. Состояния неопределенности любого параметра g в общем случае могут рассматриваться в n -мерном пространстве различных координат – \mathbf{R}^n . Частным случаем, например, могут являться координаты объема рабочего пространства x, y, z ($n = 3$) камеры для термической обработки различных изделий и заготовок. Если в качестве параметра g рассматривать температуру среды, то функция $g(x, y, z)$ будет описывать неоднородность или состояние неопределенности температурного поля в рабочем объеме камеры. Другим частным случаем может являться временная координата t . Тогда функция $g(t)$ будет описывать состояние неопределенности температуры в какой-либо точке во время обработки. Увеличение размерности пространства, например, может быть осуществлено за счет объединения упомянутых координат в единую систему. При данных условиях функция $g(x, y, z, t)$ будет описывать состояние неопределенности температурного

поля камеры в пространственно-временных координатах ($n = 4$). Возможны и другие варианты [1, 2].

Исследование эволюции состояний неопределенности позволяет «глубже» уяснить природу протекающих в анализируемом объекте процессов. Изменение состояния неопределенности параметра g проявляется, в основном, изменением вида его закона распределения и изменением характеристик его разброса или рассеяния. Для описания разброса используются следующие характеристики: размах выборки, дисперсия – σ^2 , среднее квадратическое отклонение (СКО) – σ , коэффициент вариации и др., поддающиеся достаточно простому численному определению на основании результатов наблюдений [3, 4]. Актуальна задача количественной оценки трансформации закона распределения в процессе эволюции, полностью не решенная до настоящего времени.

Обзор существующих решений.

Постановка задачи

Известен подход к описанию свойств законов распределений с помощью набора начальных и центральных моментов различных порядков, каждый из которых характеризует отдельные свойства этих законов [3, 5]. Возможности такого подхода ограничены и не позволяют получить еди-

ную комплексную оценку «вариативных» свойств законов распределений. Альтернативным является подход, основанный на использовании величины вероятностной энтропии параметра H_g для описания свойств этих законов [6–10]:

$$H_g = - \int_{-\infty}^{\infty} p(g) \ln p(g) dg, \quad (1)$$

где $p(g)$ – закон плотности распределения вероятности параметра g , зависящий только от вида этого закона и характеризующий его «вариативные» свойства.

При таком подходе количество информации I_g , порожденное изменением состояния неопределенности параметра на каком-либо этапе эволюции, будет определяться изменением величины энтропии [11–15]:

$$I_g = H_{g1} - H_{g2}, \quad (2)$$

где H_{g1} и H_{g2} – величины энтропий в начале и в конце рассматриваемого этапа эволюции системы.

Недостаток изложенного подхода – необходимость проведения больших объемов измерений, используемых для построения гистограмм распределений параметра и последующего вычисления величин энтропий. Особенно остро этот недостаток проявляется в случаях, когда проведение измерений требует значительных затрат времени, средств, использования дорогостоящей аппаратуры, реактивов и др. Такие ситуации, например, имеют место при измерении состава и свойств различных веществ и изделий и характерны для многих процессов и производств пищевой, химической, металлургической и др. отраслей промышленности. Замена реальных законов распределений какими-либо типовыми законами (например, нормальным законом) приводит к искажению оценок состояний неопределенности параметров и, следовательно, к снижению эффективности мониторинга процессов их эволюций.

В данной статье излагается подход к решению стоящей задачи, основанный на применении методов и разработок теории энтропийных потенциалов (ТЭП) [14, 15].

С его использованием представляется возможным оценивать результаты трансформации законов распределений на основе порождаемого в этом процессе количества информации. Полученные результаты являются наглядными, а предложенные методы и технологии оценивания – несложными и удобными для практического использования.

Основные положения и возможности ТЭП

В ТЭП описание состояний неопределенности осуществляется с использованием набора понятий энтропийных потенциалов, каждый из которых имеет свои особенности и возможности для описания этих состояний. Базовым является понятие энтропийного потенциала (ЭП) – Δ_e . Другие понятия из этого набора вводятся на основе понятия ЭП с привлечением дополнительных характеристик, усложняя и развивая тем самым модель состояний неопределенности. Таковыми, например, являются комплексный ЭП, многомерный комплексный ЭП. Решение стоящей задачи, возможно, осуществить, используя понятие ЭП. Идея решения состоит в следующем.

Возможности ТЭП позволяют осуществить «уход» от величины энтропии, с присущими ей указанными выше недостатками, к величине ЭП – Δ_e , функционально с ней связанной. Эта величина, в свою очередь, может быть выражена через другие характеристики рассеяния параметра, получение оценок которых требует меньших объемов результатов наблюдений. В ряде случаев соответствующие оценки могут быть получены аналитическим путем, на основе мягких измерений и вычислений, на основе аналогий с другими объектами и явлениями и т. д. Указанная взаимосвязь описывается следующей моделью [14–17]:

$$\Delta_e = \frac{1}{2} e^{H_g} = K_e \sigma, \quad (3)$$

где K_e – энтропийный коэффициент закона распределения параметра, характеризующий его «вариативные» свойства; σ – величина СКО.

Теперь, с учетом (3), исходная информационная модель изменения состояния неопределенности параметра (2) может быть представлена в следующем виде:

$$I_g = H_{g1} - H_{g2} = \ln \Delta_{e1} - \ln \Delta_{e2} = \ln \frac{K_{e1} \sigma_1}{K_{e2} \sigma_2} = \ln \frac{K_{e1}}{K_{e2}} + \ln \frac{\sigma_1}{\sigma_2} = I_I + I_P. \quad (4)$$

Если состояние неопределенности объекта рассматривается по множеству параметров G , $G = \{g_l; l \in L = \{1, \dots, m\}\}$, то количество информации, порожденной на каждом отдельном этапе эволюции, может быть определено из выражения

$$I_\Sigma = \sum_{l=1}^m c_l I_{g_l}. \quad (5)$$

В выражении (5) величины $c_l (l = 1, \dots, m)$, $\left(\sum_{l=1}^m c_l = 1\right)$ называются весовыми коэффициентами. Они характеризуют «вес» или значимость состояния неопределенности каждого из параметров для описания состояния неопределенности « m -мерного» объекта.

В выражении (4) все величины с индексом 1 соответствуют началу, а с индексом 2 – концу рассматриваемого этапа эволюции системы. Величины

$$I_I = \ln \frac{K_{e1}}{K_{e2}} = \ln k_{ke}, \quad (6)$$

$$I_P = \ln \frac{\sigma_1}{\sigma_2} \quad (7)$$

называются *интеллектуальной и энергетической составляющей информации*, соответственно [15]. Величина k_{ke} называется *коэффициентом преобразования закона распределения*, рекомендации по ее определению в различных ситуациях приведены, например, в [15].

Технологии и методы проведения анализа, синтеза, организации управления для систем различной природы, основанные на «информационном» подходе, получают все большее распространение [18–20].

Методология и пример построения информационной шкалы

Так как в работе исследуется влияние вида закона распределения параметра на изменение его состояния неопределенности, то для получения информационной оценки этого явления будем полагать, что $\sigma_1 = \sigma_2$. Тем самым выделяется исследуемое явление в «чистом» виде. Следовательно, согласно (4) и (7), $I_P = \ln \frac{\sigma_1}{\sigma_2} = 0$, что и обуславливает использование величины I_I для проведения исследований. Также необходимо определить следующие позиции.

1. Для понимания количественной сути изменений соответствующих состояний следует определить реально возможный диапазон варьирования величины I_I . Это можно сделать исходя из следующих соображений. Реально возможный диапазон варьирования величины энтропийного коэффициента находится в пределах [6–8]:

$$1 \leq K_e \leq 2,07. \quad (8)$$

Поэтому максимально возможное значение величины $I_I - I_I(\max)$ находится из условия:

$$I_I(\max) = \ln \frac{K_{e1}(\max)}{K_{e2}(\min)} = \ln 2,07 \approx 0,73 \text{ [нит]}. \quad (9)$$

Аналогично, минимально возможное значение величины $I_I - I_I(\min)$ находится из условия:

$$I_I(\min) = \ln \frac{K_{e1}(\min)}{K_{e2}(\max)} = -\ln 2,07 \approx -0,73 \text{ [нит]}. \quad (10)$$

Единица измерения количества информации, в данном случае – нит, определяется основанием логарифма в выражениях (9) и (10), т. н. натуральные логарифмы. Использование других логарифмов, например двоичных или десятичных, породит соответствующие единицы: бит и дит. Соотношение между различными единицами измерения информации выражается через основания соответствующих логарифмов с использованием известных зависимостей. Исходя из полученных результатов (9) и (10) можно указать реально возможный диапазон варьирования величины I_I в виде:

$$-0,73 \leq I_I \leq 0,73 \text{ [нит]}. \quad (11)$$

Отрицательные значения величины I_l соответствуют ситуациям, когда в результате завершения рассматриваемого этапа эволюции системы имело место возрастание величины энтропийного коэффициента, т. е. $K_{e1} < K_{e2}$ и наоборот.

2. Также необходимо определить «реперные» точки для величины I_l из диапазона (11), которые целесообразно применять при мониторинге процесса эволюции закона распределения. В качестве таковых предлагается использовать этапы переходов между типовыми, известными и хорошо изученными законами распределений, например, приведенными в табл. 1. Большинство реальных законов распределений параметров, встречающихся на практике, может быть достаточно точно «округлено» до этих законов. При необходимости исходное множество законов может быть пополнено другими законами, что позволит повысить «качество» дискретизации состояний неопределенности параметра на основе информационных оценок. При этом методология и технология проведения самих исследований не изменится. Состоятельность предложенного набора пяти законов распределения подтверждается тем, что значения величин соответствующих энтропийных коэффициентов полностью и равномерно «накрывают» весь реально возможный диапазон их изменения (8). Эти законы расположены в порядке возрастания соответствующих значений величины K_e , их нумерация приведена в столбце 1 табл. 1. Количество информации, порожденное при переходе от i -го к j -му закону распределения параметра, согласно (6), будет определяться из выражения:

$$I_l(i, j) = \ln \frac{K_{ei}}{K_{ej}} = \ln k_{ke}(i, j);$$

$$(i = 1, 2, \dots, 5; j = 1, 2, \dots, 5). \quad (12)$$

В соответствии с (12), значения величин $I_l(i, j)$ для возможных сочетаний индексов i и j вычислены и приведены в табл. 2. Особенности представленных результатов состоят в следующем.

В ячейках главной диагонали, содержаемое которых выделено жирным шрифтом, находятся нули. Это следует из того,

что для всех элементов этой диагонали выполняется условие $i = j$. Другими словами, в результате соответствующих переходов вид закона распределения не изменился. Следовательно, справедливо соотношение $K_{ei} = K_{ej}$ и значение величины $k_{ke} = 1$. Поэтому, согласно (12), значение величины I_l в этих ячейках будет равно нулю.

Вторая особенность состоит в том, что при смене очередности следования индексов, согласно (12), дробь под знаком логарифма «перевернется» (т. е. станет обратной относительно исходной). Это обстоятельство приведет к смене знака результата. Поэтому значения величин I_l в ячейках, расположенных симметрично относительно главной диагонали, равны по модулю, но имеют противоположные знаки.

Использование предложенного подхода для организации мониторинга и управления

Прикладная значимость полученных результатов имеет перспективы применения в следующих аспектах.

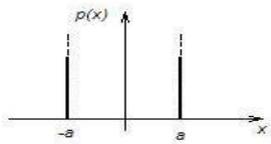
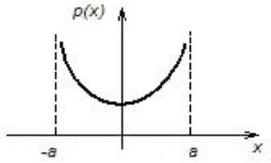
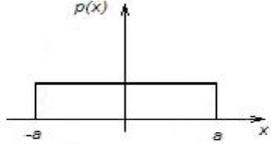
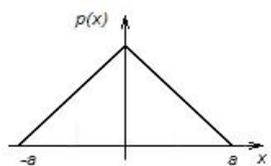
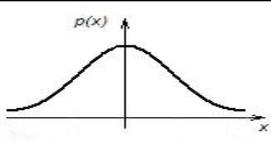
1. Возможность проведения оперативного мониторинга изменения состояний неопределенности, обусловленных изменением закона распределения параметра. Как отмечено выше, условно, реальные законы распределения могут быть «округлены» до какого-либо типового закона, из множества априори выбранных «реперных» законов, например, приведенных в табл. 1. Поэтому влияние изменения закона распределения параметра на каком-либо этапе эволюционного процесса на изменение его состояния неопределенности может быть оценено в виде соответствующей величины $I_l(i, j)$ с использованием данных табл. 2. Так, например, если изначально закон распределения был аппроксимирован законом равномерной плотности распределения ($i = 3$), а впоследствии трансформировался к виду, аппроксимируемому треугольным законом распределения ($j = 4$), то значение величины $I_l(3, 4)$ будет равно $-0,15$ [бит]. Знак минус свидетельствует, что состояние неопределенности, обусловленное таким процессом трансформации закона распределения, возросло.

Таблица 1

Типовые законы распределения вероятностей, использованные для построения «реперных» точек на информационной шкале

Table 1

Typical laws of probability distribution used to construct the «reference» points on the information scale

Тип закона распределения	Плотность вероятности $p(x)$	Вид кривой плотности вероятности	Значение K_e
1. Дискретное двухзначное Распределение	$p(x) = \begin{cases} 0,5 & \text{при } x = a; \\ 0 & \text{при } x \neq a \end{cases}$		1,00
2. Арксинусоидальный закон	$p(x) = \frac{1}{\pi a \sqrt{1 - \left(\frac{x}{a}\right)^2}}; x \leq a$		$\frac{\pi}{2\sqrt{2}} \approx 1,11$
3. Закон равномерной плотности	$p(x) = \begin{cases} \frac{1}{2a} & \text{при } x \leq a; \\ 0 & \text{при } x > a \end{cases}$		$\sqrt{3} \approx 1,73$
4. Треугольный закон распределения	$p(x) = \begin{cases} 0 & \text{при } x > a; \\ \frac{a - x }{a^2} & \text{при } x \leq a \end{cases}$		$\sqrt{\frac{3e}{2}} \approx 2,02$
5. Нормальный закон	$p(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{x^2}{2\sigma^2}}$		$\sqrt{\frac{\pi e}{2}} \approx 2,07$

2. Возможность обоснованно и оперативно выбрать канал внесения управляющего воздействия для изменения состояния неопределенности объекта по рассматриваемому параметру. Идея такого выбора основана на анализе соотношения величин I_I и I_P в выражении (4) для конкретного объекта и конкретной ситуации. Если имеет место доминирование какой-либо из компонент, то наличием другой компоненты можно пренебречь и выбирать канал внесения управляющего воздействия только для доминанты.

Так, например, если состояние неопределенности температуры в рабочем объеме упомянутой выше термокамеры в основном определяется величиной I_P , то есть $I_P \gg I_I$, то предпочтительнее выбирать канал, через который можно оказать влияние на процесс формирования величины СКО. Другими словами, имеет место ситуация, когда характеристика разброса температуры в анализируемом объеме, в виде величины σ , изменяется более значительно, чем величина энтропийного коэффициента. Изменение величины СКО (в данном случае

Таблица 2

Значения величин $I_i(i, j)$, порождаемых при переходах между типовыми законами распределения параметра

Table 2

The values of the quantities $I_i(i, j)$ generated during transitions between the typical laws of parameter distribution

$i \backslash j$	1	2	3	4	5
1	0,00	-0,10	-0,55	-0,70	-0,73
2	0,10	0,00	-0,44	-0,60	-0,62
3	0,55	0,44	0,00	-0,15	-0,18
4	0,70	0,60	0,15	0,00	-0,02
5	0,73	0,62	0,18	0,02	0,00

уменьшение) может быть осуществлено различными способами, направленными на «подавление» основных возмущений, определяющих ее формирование. Таковыми, в основном, являются, перетечи наружного, более теплого воздуха, возникающие при открывании люков или дверей в процессе загрузки и выгрузки продукции. Подавление или компенсация таких воздействий может быть осуществлена, например, путем вдувания холодного воздуха в створ люка в период его открывания. Другой способ – создание буферного, шлюзового отсека перед загрузочным люком. Реализация подобных решений позволит уменьшить дестабилизирующее влияние указанных возмущений на состояние неопределенности температурного поля камеры, что приведет к уменьшению величины σ .

Наоборот, если в силу условий и специфики эксплуатации камеры величина σ изменяется менее значительно, чем величина K_e , то есть $I_p \ll I_j$, то наиболее эффективным вариантом организации управления состоянием неопределенности температурного поля будет внесение воздействий, направленных на изменение закона распределения температуры в камере.

Здесь основным возмущением будет являться изменение условий циркуляции потоков охлаждающего воздуха вследствие, например, изменения объемов хранящейся продукции и схемы или конфигурации расположения соответствующих блоков и контейнеров. Управление процессом циркуляции потоков может быть осуществлено за счет целенаправленного изменения положений направляющих заслонок и экранов внутри камеры, что, в конечном счете, приведет к требуемым изменениям величины энтропийного коэффициента.

В общей ситуации, когда значения величин I_l и I_p соизмеримы, то есть $I_p \approx I_l$, необходимо использовать весь комплекс изложенных выше решений по организации управления состоянием неопределенности температурного поля на основе информационной модели (4).

Следует отметить, что методы и технологии организации мониторинга и управления состояниями неопределенности различных объектов, основанные на использовании энтропийных и информационных характеристик, являются востребованными, что находит отражение в планах подготовки кадров высшей квалификации и формировании компетентностных моделей выпускников вузов [21].

Заключение

В статье изложены методология, алгоритм и пример создания информационной шкалы для получения количественных оценок изменений состояний неопределенности, обусловленных трансформацией законов распределений параметров. Использование информационных шкал позволяет оперативно получать оценки изменений соответствующих состояний без применения сложных вычислительных процедур. Предложенные решения базируются на методах и технологиях ТЭП и имеют перспективы применения для организации мониторинга и управления состояниями неопределенности объектов и систем различной природы.

СПИСОК ЛИТЕРАТУРЫ

1. Лазарев В.Л., Богданов П.А., Петров М.М., Томсон К.Ю. К вопросу оценки неоднородности свойств и состава сред // Вестник Международной академии холода. 2018. № 1. С. 74–80.
2. Лазарев В.Л. Исследование и проектирование многокомпонентных систем в биотехнологической промышленности в условиях неопределенности на основе энтропийных потенциалов показателей качества // Вестник Международной академии холода. 2017. № 2. С. 84–90.
3. Боровков А.А. Теория вероятностей. М.: КД Либроком, 2016. 656 с.
4. Геворкян П.С. Теория вероятностей и математическая статистика. М.: Физматлит, 2016. 176 с.
5. Тутубалин В.Н. Теория вероятностей. М.: Academia, 2018. 210 с.
6. Ho S.W., Yeung R.W. The interplay between entropy and variational distance // IEEE Trans. Inform. Theory. 2010. Vol. 56. No. 12. Pp. 5906–5929.
7. Shirokov M.E. Tight uniform continuity bounds for the quantum conditional mutual information, for the Holevo quantity, and for capacities of quantum channels // J Math. Phys. 2017. Vol. 58. No. 10. Pp. 102–202.
8. Wilde M.M. Quantum information theory. Cambridge, UK; New York: Cambridge Univ. Press, 2017.
9. Keevash P., Long E. On the normalized Shannon capacity of a union // Combin. Probab. Comput. 2016. Vol. 25. No. 5. Pp. 766–767.
10. Isermann R. Fault diagnosis application. Heidelberg: Springer, 2011. 354 p.
11. Колмогоров А.Н. Теория информации и теория алгоритмов. М.: Наука, 1987. 304 с.
12. Lazarev V.L. Processing of observations on the basis of information criteria // Proc. of the 19th Internat. Conf. on Soft Computing and Measurements. St. Petersburg, Russia, 2016. Publisher: IEEE. Pp. 48–50. DOI: 10.1109/SCM.2016.7519679.
13. Чернавский Д.С. Синергетика и информация: Динамическая теория информации. М.: Ленанд, 2017. 304 с.
14. Лазарев В.Л. Теория энтропийных потенциалов. СПб.: Изд-во Политехнического университета, 2012. 127 с.
15. Теоретические основы и методы // Мягкие измерения и вычисления. В 3-х т. Т. 1. М.: ИД «Научная библиотека», 2017. 420 с.
16. Hu Wenhua, Jeffrey C. Carver, Vaibhav Anu, Gursimran Wilia, Gary Bradshaw. Defect prevention in requirements using human error information: An empirical study. Springer International Publishing. Ab 2017. LNCS 10153. Pp. 61–76.
17. Туричин А.М., Новицкий П.В., Левшина Е.С. и др. Электрические измерения неэлектрических величин. Л.: Энергия, 1975. 576 с.
18. Krishnamoorthy S. Linguistic features for review helpfulness prediction // Expert Syst. With Appl. 2015. Vol. 42. No. 7. Pp. 3751–3759.
19. Papageorgiou E. (Ed.) Fuzzy cognitive maps for applied science and engineering: From fundamentals to extensions and learning algorithms. Springer-Verlag, 2014.
20. Le Ba Chung, Holopov Y.A. Information environment for neural-network adaptive control system. Springer, 2017. Vol. 736. Pp. 59–64.
21. Lazarev V.L. Epistemological foundations for generation of perspective competencies in the training of personnel for industrial and economic complex // 4th Forum Strategic Partnership of Universities and Enterprises of Hi-Tech Branches (Science, Education, Innovations). St. Petersburg, Russia. 2015. Publisher: IEEE. Pp. 26–28. DOI: 10.1109/IV Forum.2015.7388242

Статья поступила в редакцию 24.05.2019.

REFERENCES

1. Lazarev V.L., Bogdanov P.A., Petrov M.M., Tomson K.Yu. Evaluating heterogeneity of composition and properties for different environments. *Vestnik Mezhdunarodnoi akademii kholoda [Bulletin of the International Cold Academy]*, 2018, No. 1, Pp. 74–80. (rus)
2. Lazarev V.L. Analysis and design of multicomponent systems in biotechnology industry in the context of uncertainty based on entropy potential quality indicators. *Vestnik Mezhdunarodnoi akademii kholoda [Bulletin of the International Cold Academy]*, 2017, No. 2, Pp. 84–90. (rus)
3. Borovkov A.A. *Teoriya veroyatnostey [Probability theory]*. Moscow: KD Librokom Publ., 2016, 656 p. (rus)
4. Gevorkyan P.S. *Teoriya veroyatnostey i matematicheskaya statistika [Theory of Probability and Mathematical Statistics]*. Moscow: Fizmatlit Publ., 2016, 176 p. (rus)
5. Tutubalin V.N. *Teoriya veroyatnostey [Probability theory]*. Moscow: Academia Publ., 2018, 210 p. (rus)
6. Ho S.W., Yeung R.W. The interplay between entropy and variational distance. *IEEE Trans. Inform. Theory*, 2010, Vol. 56, No. 12, Pp. 5906–5929.

7. **Shirokov M.E.** Tight uniform continuity bounds for the quantum conditional mutual information, for the Holevo quantity, and for capacities of quantum channels. *J. Math. Phys.*, 2017, Vol. 58, No. 10, Pp. 102–202.
8. **Wilde M.M.** *Quantum information theory*. Cambridge, UK; New York: Cambridge Univ. Press, 2017.
9. **Keevash P., Long E.** On the normalized Shannon capacity of a union. *Combin. Probab. Comput.*, 2016, Vol. 25, No. 5, Pp. 766–767.
10. **Isermann R.** *Fault Diagnosis Application*. Heidelberg: Springer, 2011, 354 p.
11. **Kolmogorov A.N.** *Teoriya informatsii i teoriya algoritmov [Information Theory and Algorithm Theory]*. Moscow: Nauka Publ., 1987, 304 p. (rus)
12. **Lazarev V.L.** Processing of observations on the basis of information criteria. *Proceedings of the 19th International Conference on Soft Computing and Measurements*. St. Petersburg, Russia, 2016. Publisher: IEEE. Pp. 48–50. DOI: 10.1109/SCM.2016.7519679.
13. **Chernavskiy D.S.** *Sinergetika i informatsiya: Dinamicheskaya teoriya informatsii [Synergetics and Information: Dynamic Information Theory]*. Moscow: Lenand Publ., 2017, 304 p. (rus)
14. **Lazarev V.L.** *Teoriya entropiynykh potentsialov [Theory of Entropy Potentials]*. St. Petersburg: Izd-vo Politekhnicheskogo un-ta, 2012, 127 p. (rus)
15. **Prokopchinoy S.V. (Ed.)** *Teoreticheskiye osnovy i metody. Myagkiye izmereniya i vychisleniya [Soft measurements and calculations]*. Moscow: ID «Nauchnaya biblioteka» Publ., 2017, Vol. 1, 420 p. (rus)
16. **Hu Wenhua, Jeffrey C. Carver, Vaibhav Anu, Gursimran Wilia, Gary Bradshaw.** *Defect prevention in requirements using human error information: An empirical study*. Springer International Publishing, 2017, LNCS 10153, Pp. 61–76.
17. **Turichin A.M., Novitskiy P.V., Levshina Ye.S., et al.** *Elektricheskiye izmereniya neelektricheskikh velichin [Electrical measurements of non-electrical quantities]*. Leningrad: Energiya Publ., 1975, 576 p. (rus)
18. **Krishnamoorthy S.** Linguistic features for review helpfulness prediction. *Expert Syst. with Appl.*, 2015, Vol. 42, No. 7, Pp. 3751–3759.
19. **Papageorgiou E. (Ed.)** *Fuzzy cognitive maps for applied science and engineering: From fundamentals to extensions and learning algorithms*. Springer-Verlag, 2014.
20. **Le Ba Chung, Holopov Y.A.** *Information environment for neural-network adaptive control system*. Springer, 2017, Vol. 736, Pp. 59–64.
21. **Lazarev V.L.** Epistemological foundations for generation of perspective competencies in the training of personnel for industrial and economic complex. *4th Forum Strategic Partnership of Universities and Enterprises of Hi-Tech Branches (Science, Education, Innovations)*. St. Petersburg, Russia, 2015. Publisher: IEEE. Pp. 26–28. DOI: 10.1109/IV Forum.2015.7388242

Received 24.05.2019.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

ЛАЗАРЕВ Виктор Лазаревич
LAZAREV Victor L.
 E-mail: holod25@yandex.ru

DOI: 10.18721/JCSTCS.12202
УДК 004.932.2:003.63

ИЗУЧЕНИЕ ЭФФЕКТИВНОСТИ ВИЗУАЛИЗАЦИИ СТАТИЧЕСКИХ СТРУКТУР ДАННЫХ С ПОМОЩЬЮ БРУСКОВЫХ И СЕКТОРНЫХ ДИАГРАММ МЕТОДОМ АЙ-ТРЕКИНГА

Т.К. Ермолова, П.Д. Иващенко, В.В. Лаптев

Санкт-Петербургский политехнический университет Петра Великого,
Санкт-Петербург, Российская Федерация

Понимание механизмов декодирования информации во время создания диаграмм во многом гарантирует корректность и эффективность интерпретации заложенной в них информации во время считывания пользователем. В статье описаны основные подходы к представлению статических структур данных при помощи различных графических образов (секторных, кольцевых, столбиковых и полосовых диаграмм), рассмотрены исследования по определению эффективности визуализации при помощи систем отслеживания глазодвигательной активности (метода ай-трекинга). На основе статистического анализа, в том числе с использованием кластерного анализа координат точек фиксации глаз испытуемых, получено подтверждение гипотезы о существовании различных паттернов и подходов к восприятию диаграмм, выявлены недостатки визуализации информации при помощи секторных диаграмм. Сформулированы рекомендации по визуализации статических структур данных.

Ключевые слова: инфографика, визуализация данных, секторные диаграммы, кольцевые диаграммы, брусковые диаграммы, ай-трекинг.

Ссылка при цитировании: Ермолова Т.К., Иващенко П.Д., Лаптев В.В. Изучение эффективности визуализации статических структур данных с помощью брусковых и секторных диаграмм методом ай-трекинга // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2019. Т. 12. № 2. С. 16–27. DOI: 10.18721/JCSTCS.12202

STUDYING THE EFFICIENCY OF VISUALIZATION OF STATIC DATA STRUCTURES BY MEANS OF BAR AND PIE CHARTS BY EYE-TRACKING METHOD

T.K. Ermolova, P.D. Ivaschenko, V.V. Laptev,

Peter the Great St. Petersburg Polytechnic University,
St. Petersburg, Russian Federation

The understanding of the mechanisms used for decoding information ensures efficient and correct interpretation of the information if it is taken into consideration during the design process. The article describes the most popular approaches to the visual presentation of static data structures with the help of graphic images (pie charts, donut charts, vertical and horizontal bar charts), an overview different eye-tracking studies of the effectiveness of visualization is given. Based on the statistical analysis, including cluster analysis of the

coordinates of the fixation points of the subjects' eyes, a confirmation of the hypothesis of the existence of various patterns and approaches to the perception of diagrams was obtained. The drawbacks of visualization of information using pie charts were discovered. Certain recommendations for the visualization of static data structures were formulated.

Keywords: infographics, data visualization, pie charts, donut charts, bar charts, eye-tracking.

Citation: Ermolova T.K., Ivaschenko P.D., Laptev V.V. Studying the efficiency of visualization of static data structures by means of bar and pie charts by eye-tracking method. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2019, Vol. 12, No. 2, Pp. 16–27. DOI: 10.18721/JCSTCS.12202

Введение

Визуализация в части представления совокупностей числовых данных различного рода и управление их восприятием – важная прикладная задача. Ее решение непосредственно связано с выбором графического образа, соответствующего семантически и перцепционно конкретному набору данных. Во время создания диаграммы информация кодируется в геометрические или изобразительные фигуры, а во время считывания – декодируется для дальнейшего анализа. Соответственно корректное построение диаграмм является залогом правильного решения визуализации числовой информации, при этом эффективное представление количественных показателей требует понимания чисел и способности показать их в виде графического сообщения для точной и эффективной интерпретации читателем [1, с. 9]. Следует также учитывать характер сравнения, определяющего назначение статистических графиков и являющегося одним из основополагающих критериев типологии количественной инфографики [2, с. 106]. Среди прочих – это сравнение структур явлений, состоящих из нескольких компонентов. Цель данной работы – определение влияния типа визуализации единичной статической структуры данных на эффективность ее восприятия.

В настоящее время имеются разнообразные инструменты информационной графики для представления структурных диаграмм. Они служат для выявления размеров отдельных компонентов, их отноше-

ния к целому и сравнения между собой. Речь идет о пропорциях значений отдельных элементов или структуре совокупности, состоящей из нескольких долей. Одна и та же структура данных может быть визуализирована с помощью различных диаграмм: брусковых стековых (полосовых, столбиковых), секторных (круговых и кольцевых), фигурных и потоковых, а также используя древовидные карты и процентные квадраты. Правильный выбор формы зависит от целого ряда факторов: контекста, семантических связей формы диаграммы и контента, удобочитаемости, композиции и простоты восприятия [3].

К распространенным инструментам визуализации относятся секторные диаграммы и их производные, например, кольцевые диаграммы. Они наиболее удачно демонстрируют структуру одиночной совокупности, поскольку круг является олицетворением целого (100 %), а его сектора – процентных долей. Очевидным конкурентом по визуализации данного типа числовых данных признаны брусковые стековые диаграммы горизонтального и вертикального расположения. В данной статье сравнивается эффективность восприятия различных типов диаграмм визуализации структуры данных, изучаются особенности формирования паттерна рассматривания при помощи метода ай-трекинга.

Анализ проблемы

Каждый тип представления структур имеет свои проектные правила и рекомендации к построению, ограничивающие об-

ласти применения [4]. Секторные диаграммы принято формировать не более чем из пяти секторов, которые не должны ни резко отличаться по величине, с одной стороны, ни быть примерно равнозначными, с другой. Данные должны быть упорядочены, поэтому сектора располагаются от большего к меньшему от верхнего вертикального радиуса. К особенностям построения брусковых стековых диаграмм относится запрет на искривление координатных осей и соответствующего искажения формы столбиков или полос. Кроме того, столбики и полосы изображаются полностью, т. к. в первую очередь сравниваются линейные размеры их долей и целого, поэтому необходимо обеспечить обязательность нулевой отметки и отсутствие разрывов в диаграмме, т. е. запрещается разрывать ось значений X для полосовой, а ось Y для столбиковой диаграммы.

Для визуализации большого количества структур данных существует возможность использования и секторных, и брусковых диаграмм. Исследователи неоднократно затрагивали проблему инвариантности использования круговых и брусковых диаграмм. Так, американский ученый Карл Карстен выступил с критикой восприятия секторных диаграмм. Он опирался на предположение, что отдельные сектора диаграммы представляются человеком в виде набора площадей, плохо оцениваемых человеческим взглядом. На основании чего Карстен сделал вывод, что визуализация при помощи секторных диаграмм нежелательна [5].

Первым вопрос сопоставления эффективности секторных и брусковых диаграмм поднял Вальтер Иллс в 1926 г. Изучив считывание двух типов диаграмм, он пришел к выводу, что секторные диаграммы считываются так же быстро, как и брусковые. Более того, было отмечено, что секторные диаграммы позволяют прочесть данные более точно [6]. Согласно опросу, большая часть испытуемых оценивала значения секторов при помощи дуг, а остальные руководствовались площадью секторов или

размером углов. По итогам экспериментов Иллс установил, что секторные диаграммы привлекательны и привычны.

Дальнейшее сравнение секторных и брусковых диаграмм показало, что, несмотря на критические замечания по процедуре эксперимента Иллса, подтвердилась эффективность секторных диаграмм: они действительно считывались быстрее и точнее [7].

Дискуссия относительно формы структурных диаграмм продолжилась Уильямом Кливлендом. В своих работах он неоднократно выступал с критикой секторных диаграмм, указывая на то, что по сравнению с брусковыми они менее эффективны [8, 9]. И это несмотря на то, что секторные диаграммы считаются более интуитивными для человека (круг — целое, сектор — доля). Он обращал внимание на абстрактность и ограниченность такого типа визуализации, который должен применяться только в редких случаях. Кливленд опубликовал данные эксперимента, в котором предлагал сравнить сегмент секторной диаграммы и брусковую диаграмму [8]. Это исследование показало, что брусковые диаграммы были в два раза эффективнее секторных. Однако, стоит отметить, что секторные диаграммы, которые Кливленд применял в своем эксперименте, были построены не по общепринятым правилам, тогда как брусковые столбиковые диаграммы были построены по всем правилам, и к ним также предлагались шкалы для более эффективного считывания.

В 1991 г. Ян Спенс и Стефан Левандовски провели эксперимент на базе компьютера и установили, что секторные диаграммы более эффективны, чем брусковые в тех случаях, когда испытуемым требовалось сравнить несколько графиков. Более того, они отметили, что во время восприятия секторных диаграмм испытуемые визуально обозначали для себя отметки 25, 50 и 100 процентов и считывали диаграммы, исходя из этих отметок [10].

В настоящее время отсутствует единое мнение о том, в каком случае какая из диаграмм — брусковая или секторная — спо-

собны визуализировать данные наилучшим образом. На протяжении долгого времени считалось, что люди визуальнее лучше оценивают длину, чем углы, дуги и площади. В этом заключается преимущество брусковых диаграмм. Однако часть экспериментов, представленных выше, свидетельствуют об обратном. На данный момент исследователи склоняются к анализу процесса восприятия каждой диаграммы по отдельности. Так, Роберт Косара и Дрю Скау провели многочисленные эксперименты в поисках значимого аспекта восприятия секторной диаграммы (центр, угол или внешняя дуга), влияющего на считывание информации [11]. В другой работе рассмотрено влияние размера диаграмм и наличия визуальных ключей на корректность восприятия секторных и кольцевых диаграмм, исследована точность оценки пропорций отдельных частей структуры [12].

Однако вопрос о том, какой способ визуализации структурных данных является наиболее эффективным, так и остается открытым. Здесь под эффективностью визуального восприятия информации понимается точность и скорость решения задачи [13, с. 139]. Неопределенности выбора геометрического образа и соответствующего типа структурной диаграммы, секторной или брусковой, побуждает продолжить исследования в данном направлении с использованием метода ай-трекинга. Это позволит оценить особенности формирования паттерна рассматривания отдельных видов диаграмм и сформулировать рекомендации по выбору способов визуализации структур данных.

Методы исследования

Механизмы восприятия визуализации структур данных в большинстве указанных выше исследований оценивались при помощи субъективных метрик, таких как опросы и анкетирование. Степень эффективности визуализации оценивалась в зависимости от количества правильных ответов.

В данной статье предлагается методика оценки эффективности выбора формы визуализации данных при помощи кластерного анализа глазодвигательной активности [14], т. н. видеоокулографии или ай-трекинга. Данные, полученные таким образом, позволяют оценить не только правильность восприятия, но и когнитивную нагрузку испытуемых во время решения поисковых задач. Система ай-трекинга передает данные о передвижениях зрачка испытуемого на обработку в компьютер и, исходя из полученных данных, позволяет определить т. н. «зоны интереса» и стратегии восприятия визуальной информации.

В предыдущих исследованиях доказано, что система отслеживания глазодвигательной активности может применяться для решения различных прикладных задач: для оценки влияния композиции макета на понимание диаграмм [15], для анализа подходов к восприятию элементов графов (взаиморасположению узлов и дуг) [16] и для оценки различий в методах восприятия линейных и брусковых диаграмм [17].

Принято считать, что движения глаз отражают три визуальные стратегии при рассматривании: определение формы, распознавание формы и реинтерпретацию формы [18]. Система оценки глазодвигательной активности использует две основные метрики: фиксации и саккады [19]. Фиксации отвечают за момент восприятия информации и интерпретации форм. Саккады показывают процесс поиска информации на представленном стимульном материале [19, 20]. Как следствие, длительность фиксации, количество саккад и их амплитуды могут использоваться в качестве индикаторов интерпретации формы во время поисковых задач различного характера [18, 21].

Таким образом, система анализа глазодвигательной активности позволяет определить степень эффективности и механизмы восприятия инфографики испытуемым, а также обозначить области когнитивной нагрузки во время решения задач [17], что позволит сделать выводы о приоритетности формы визуализации.

Постановка эксперимента

Используемый в данной работе научный метод исследования подразумевает проведение эксперимента на основе айтрекера (электронного видеоокулографа). С его помощью было принято решение провести экспериментальное сравнение визуального восприятия секторных и брусковых диаграмм. В рамках эксперимента подготовили следующий стимульный мате-

риал: пять видов секторных, кольцевых, столбиковых и полосовых диаграмм. Для того, чтобы определить влияние насыщенности цвета на восприятие диаграмм, подготовили второй набор стимульного материала: имеющиеся графики окрасили в инверсированном порядке: от светлого к темному, таким образом, чтобы наибольший сектор был самым светлым, а наименьший – темным (рис. 1).

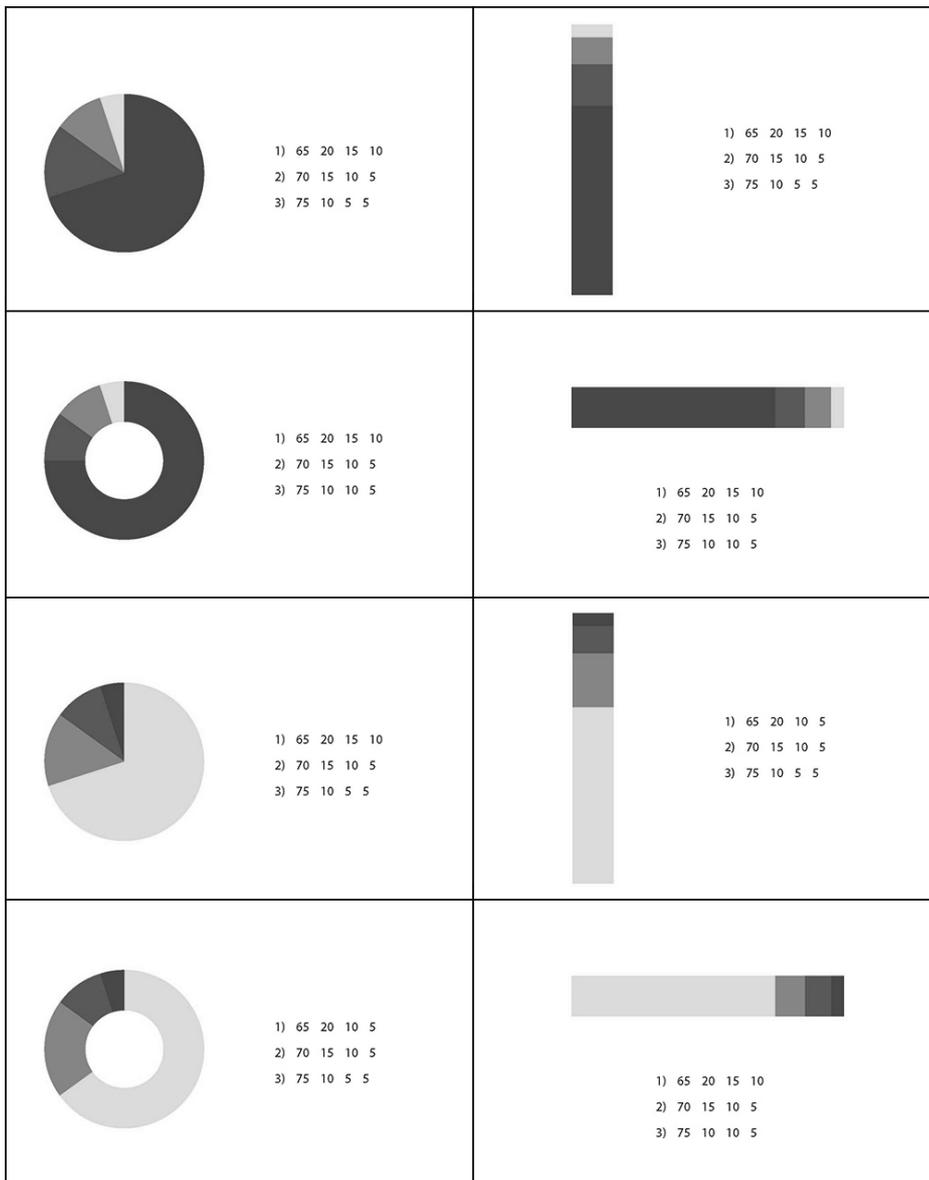


Рис. 1. Стимульный материал: секторная, кольцевая, столбиковая брусковая и полосовая брусковая диаграммы в двух вариантах: от темного большого к светлому меньшему и инверсированный

Fig. 1. Stimulus material: pie chart, donut chart, vertical and horizontal bar chart in two versions: from dark to light smaller and inverted

Диаграммы построили в соответствии с общепринятыми рекомендациями, на каждом графике визуализировалось процентное соотношение из четырех значений: от большего к меньшему, и соответствующий порядок оттенков цвета с изменением насыщенности и светлоты. Справа от каждого графика было представлено три варианта процентных соотношений. Графики окрасили в оттенки одного цвета, что позволило исключить психологическое влияние различных цветов на восприятие информации.

В эксперименте приняло участие 44 испытуемых в возрасте от 19 до 28 лет, студенты различных курсов Санкт-Петербургского политехнического университета Петра Великого. В ходе эксперимента каждому из испытуемых демонстрировалось в случайном порядке по пять графиков из каждой группы. Они рассматривали каждую диаграмму, после чего переходили на страницу опросника, где указывали, какое из предложенных процентных соотношений соответствует рассмотренной диаграмме. Время просмотра диаграмм ограничено не было.

Стимульный материал был представлен на персональном компьютере Intel Core 2 Duo CPU E8400, с оперативной памятью 3,25 Гб и операционной системой Microsoft Windows XP на мониторе BENQ XL2411 24" (53×30 см), 1920×1080 px, 144 Гц. Данные о глазодвигательной активности собирались при помощи системы SMI RED250 и программного обеспечения SMI iViewX.

Калибровка оборудования проводилась в соответствии с методиками и техническими условиями, предъявляемыми к экспериментам с использованием айтрекингового оборудования в лаборатории человеко-компьютерного взаимодействия Санкт-Петербургского политехнического университета Петра Великого, согласно утвержденному порядку [3].

В ходе эксперимента проверялась основная гипотеза: тип графика (секторный, кольцевой, столбиковый брусковый и полосовой брусковый) влияет на процесс восприятия визуальной информации при отсутствии семантического контекста данных. Исходя из этой гипотезы сформулировано шесть основных вопросов исследования.

Влияет ли тип и порядок следования оттенков цвета графиков:

(RQ 1) на время решаемой задачи;

(RQ 2) на правильность решения задачи;

(RQ 3) на продолжительность фиксации;

(RQ 4) на количество фиксаций;

(RQ 5) на количество саккад;

(RQ 6) на паттерн рассматривания.

Результаты

Во время анализа результатов рассматривались (1) средняя продолжительность фиксации, (2) среднее количество фиксаций, (3) среднее количество саккад, (4) общая продолжительность решения задачи и (5) количество правильных ответов. Отдельно рассматривались паттерны рассматривания и выявленные пространственные кластеры – зоны интереса (AOI).

Программное обеспечение SMI iViewX сохраняет координаты точек и визуализирует различные карты активности (фокусные карты, тепловые карты, карты движений). Все последующие расчеты проводились на основании данных калибровки системы при помощи библиотек языка программирования Python: Pandas, SciPy и NumPy для обработки данных и Matplotlib для построения диаграмм. Кластерный анализ данных проводился при помощи оригинального программного обеспечения. Кластеризация проводилась на основе координатных данных точек фиксаций с выделением зон интереса (AOI). Статистическая значимость оценивалась при помощи однофакторного дисперсионного анализа и теста Шапиро-Уилка при $p < 0,05$.

Было установлено, что секторные диаграммы имели наибольшее среднее количество фиксаций (One Way ANOVA $F = 5,92$, $p = 0,01$) (рис. 2 а). При считывании секторных диаграмм возрастало и среднее количество саккад (One Way ANOVA $F = 6,03$, $p = 0,01$) (рис. 2 б). Интересно, что в ходе эксперимента не была установлена статистически значимая разница между средними длительностями фиксаций для всех типов основного набора диаграмм (One Way ANOVA $F = 1,82$, $p = 0,18$) (рис. 2 в). Анализ правильности ответов показал, что все типы диаграмм одинаково хорошо считывались испытуемыми и статистически значимых отличий нет (рис. 2 г).

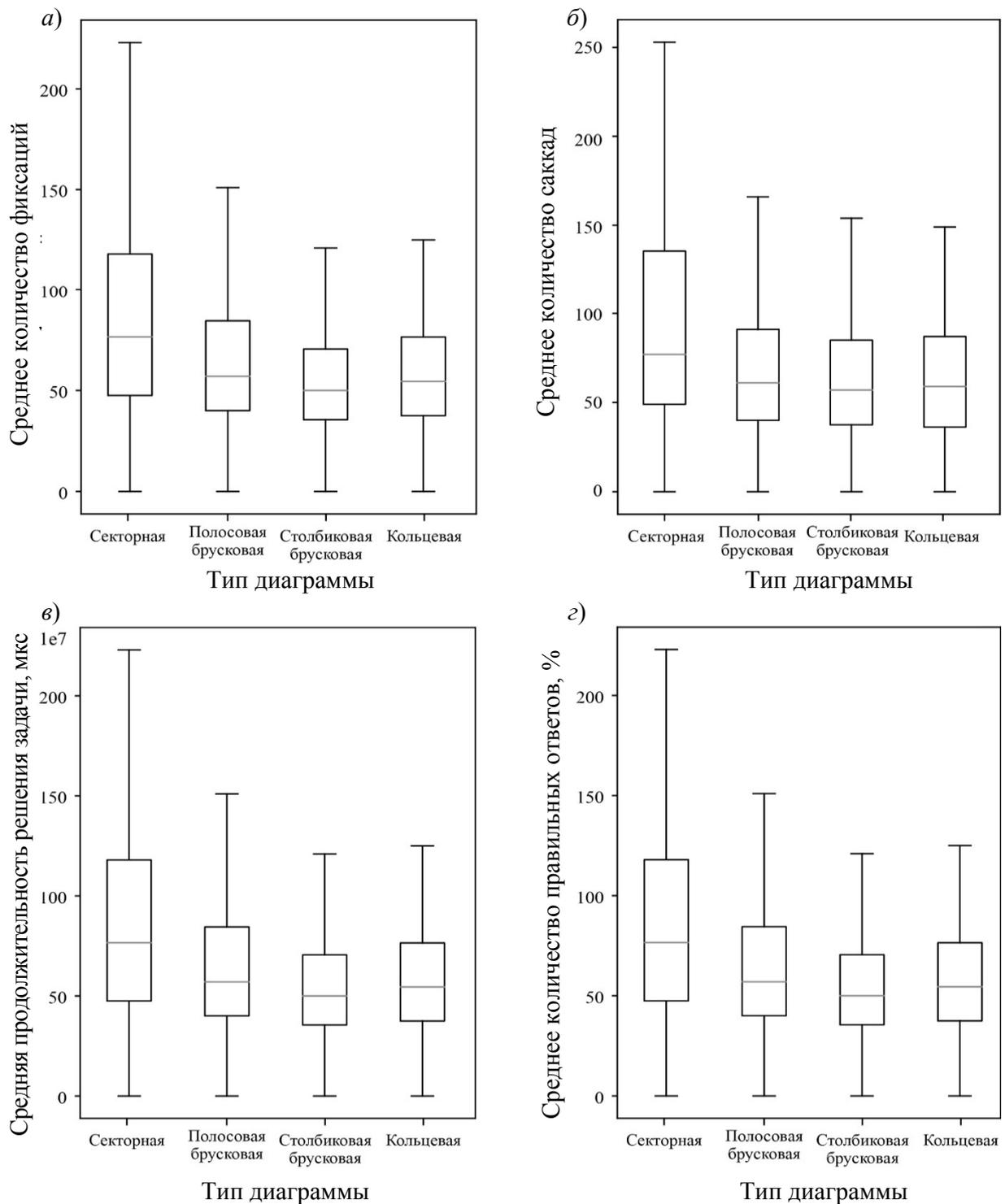


Рис. 2. Распределение значений в зависимости от фактора «Тип диаграммы»:
a – среднее количество фиксаций; *б* – среднее количество саккад; *в* – средняя продолжительность решения задачи, мкс; *г* – среднее количество правильных ответов

Fig. 2. The distribution of values depending on the «Chart Type» factor: *a* – the average number of fixations; *б* – the average number of saccades; *в* – the average duration of the solution of the problem (microseconds); *г* – average number of correct answers

Таким образом, можно предположить, что во время считывания секторных диаграмм испытуемые дольше и с большим вниманием анализировали информацию, однако при выполнении данной задачи не испытывали больших когнитивных нагрузок. Можно также предположить, что брусковые и кольцевые диаграммы проще оцениваются при быстром рассмотрении, тогда как секторные диаграммы приводят к более детальному изучению.

При сопоставлении темных и светлых графиков было обнаружено, что цвет влияет только на среднюю длительность фиксации у брусковых диаграмм (One Way ANOVA $F = 5,5$, $p = 0,01$): светлые диаграммы имели в среднем меньшую длительность фиксации. У секторных и кольцевых диаграмм такой закономерности обнаружено не было.

Для определения паттернов рассматривания и визуализации зон интересов испытуемых (AOI) построили тепловые карты.

На них цветом выделили те фрагменты стимульного материала, на которых происходили фиксации. Чем насыщенней цвет, тем больше фиксаций приходилось на определенное место на стимульном материале. Такие карты позволяют визуально оценить механизмы восприятия диаграмм. Анализ тепловых карт показал, что для всех типов диаграмм испытуемые применяли разные механизмы глазодвигательной активности: у секторных диаграмм и столбиковых брусковых рассматривались только меньшие фрагменты (рис. 3 а, в), тогда как в полосовых брусковых диаграммах считывание информации происходило при помощи визуального охвата всей диаграммы (рис. 3 б). В брусковых и кольцевых диаграммах испытуемые рассматривали варианты всех ответов полностью, тогда как во время решения задач с секторными диаграммами основное количество фиксаций приходилось на большее значение у каждого из вариантов ответов.

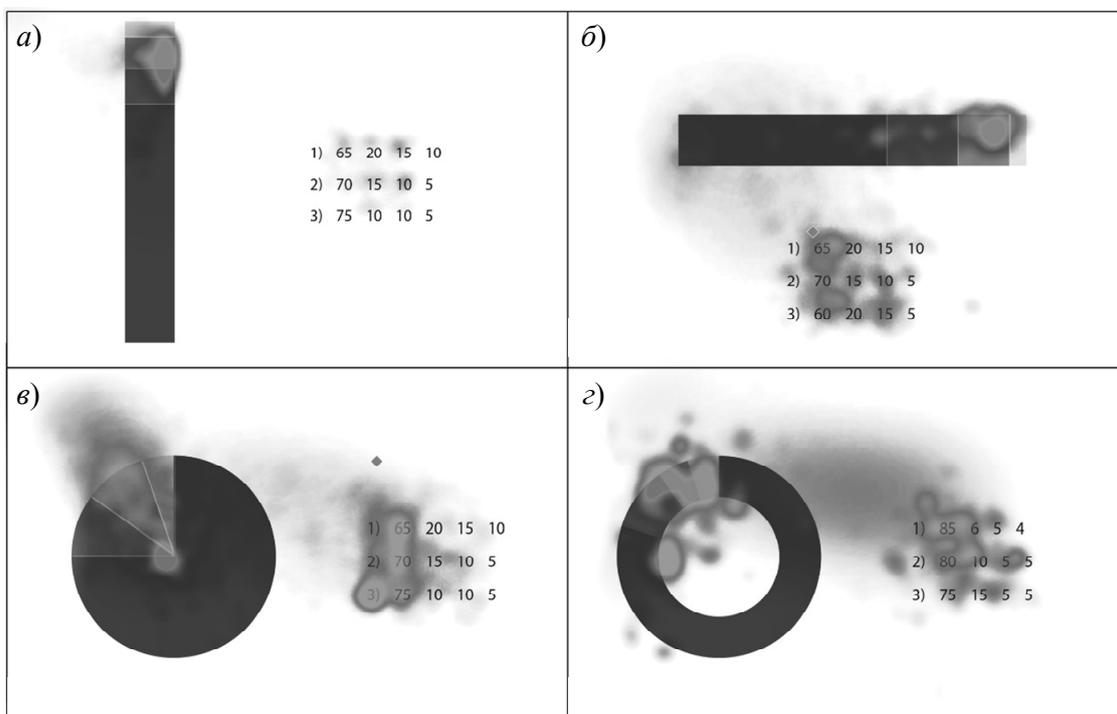


Рис. 3. Тепловые карты: а – столбиковой брусковой диаграммы; б – полосовой брусковой диаграммы; в – секторной диаграммы; з – кольцевой диаграммы

Fig. 3. Heat maps: а – vertical bar chart; б – horizontal bar chart; в – pie chart; з – donut chart

Следует отметить, что оценка тепловых карт показала, что в секторных и кольцевых диаграммах испытуемые ищут визуальные ключи, а иногда даже визуально отмеряют доли на диаграмме, чтобы ее считать и декодировать. Чем сложнее диаграммы с точки зрения близости визуализированных процентных соотношений, тем большее количество испытуемых прибегало к такой стратегии. Испытуемые взглядом обозначали нижний вертикальный радиус и, отталкиваясь от него, отмеряли значения по часовой стрелке.

Интерпретация результатов и выводы

В ходе эксперимента были получены ответы на поставленные вопросы:

RQ 1. Тип графика влияет на время решаемой задачи. Секторные диаграммы в среднем считываются дольше. Порядок следования оттенков цвета графиков влияет на время решаемой задачи только в случае брусковых диаграмм: графы, где большая область обозначена светлым, считываются быстрее.

RQ 2. Тип и порядок следования оттенков цвета графика на правильность решения задачи не влияет.

RQ 3. Тип и порядок следования оттенков цвета графиков на продолжительность фиксации не влияет.

RQ 4. Количество фиксаций возрастает во время считывания информации с секторных диаграмм. В других случаях тип и порядок следования оттенков цвета графиков на количество фиксаций не влияет.

RQ 5. Тип графика влияет на количество саккад: во время работы с секторными диаграммами среднее количество саккад намного больше, чем у других типов диаграмм. Порядок следования оттенков цвета графика на количество саккад не влияет.

RQ 6. Тип графика влияет на паттерн рассматривания. Для каждого типа диаграмм применяется свой паттерн рассматривания. Порядок следования оттенков цвета не влияет на паттерн рассматривания диаграмм.

На основе полученных результатов можно сделать следующие выводы.

Для визуализации одиночной статической структуры данных стоит отдавать предпочтение брусковому типу диаграмм. Секторные диаграммы следует выбирать только в том случае, если идет обращение к ассоциативным формам. Использование такого графического образа для визуализации структуры данных диктуется семантическими связями и контекстом использования.

Выбирая между вертикальным и горизонтальными брусковыми диаграммами, дизайнер должен обращаться к семантике контента и композиционным решениям размещения информационной графики, т. к. разница в восприятии информации в этих диаграммах не является статистически значимой. Однако, следует отметить, что паттерн рассматривания полосовых и столбиковых брусковых диаграмм имеет значительные отличия: полосовые диаграммы рассматриваются полностью, от начальной точки до конца, тогда как в столбиковых внимание приходится только на малую область, а остальная часть графика оказывается вне зоны внимания. В обоих случаях глазодвигательная активность концентрируется на меньших долях структуры. Если большая доля окрашена светлее, чем остальные, то такая диаграмма считывается быстрее.

На основе тепловых карт можно предположить, что в секторных диаграммах оценка значений происходит путем вычитания: испытуемые анализировали меньшую область графика и сравнивали с большими значениями в ответах, вычисляя оставшуюся от этих больших значений часть. Об этом свидетельствуют две явные зоны на тепловой карте: внимание на графике приходится на меньшие сегменты, тогда как в вариантах ответов испытуемые рассматривают большее значение. Соответственно, можно предположить, что эффективность секторных диаграмм падает из-за того, что, воспринимая их, пользователи совершают математические вычисления в уме.

Анализ тепловых карт показал, что в секторных диаграммах испытуемые обращают основное внимание на углы, в меньшей степени на дуги. Это утверждение основывается на наличии центральной зоны интереса, в которой происходит считывание угловых значений. Попытки определить угловые параметры долей можно увидеть и в паттерне рассматривания кольцевой диаграммы, хотя и не так выражено. Из этого следует рекомендация оставлять свободным от дополнительных экспликационных или декоративных элементов центр секторной диаграммы и ее кольцевой модификации.

За рамками настоящего исследования остались проблемы цветового кодирования диаграмм, их пропорционального соотношения относительно области листа или кадра, а также эффективности восприятия диаграмм при сравнении нескольких наборо-

вов данных или оценки динамики их структуры. Это требует отдельного исследования. Также можно отметить, что испытуемым предлагались относительно несложные диаграммы с наличием доминирующей доли, т. к. основной задачей данного исследования являлось обнаружение общих тенденций во время восприятия разных типов визуализации структуры данных.

Кластерный анализ проводился на основе одного из возможных признаков: координатного расположения фиксаций глазодвигательной активности. В настоящей статье не рассматривалась кластеризация, основанная на принципах оценки близости или порядка следования саккад, продолжительности фиксаций и пр. Использование различных принципов кластерного анализа для обработки данных глазодвигательной активности требует дальнейшего отдельного исследования.

СПИСОК ЛИТЕРАТУРЫ

1. Few S.C. Show me the numbers: Designing tables and graphs to enlighten. 2 ed. Burlingame: Analytics press, 2012. XVIII. 351 p.
2. Елисеева И.И., Юзбашев М.М. Общая теория статистики. М.: Финансы и статистика, 2004. 655 с.
3. Лаптев В.В., Орлов П.А., Драгунова О.В. Визуализация динамических структур данных с помощью потоковых диаграмм в веб-аналитике // Научно-технические ведомости Санкт-Петербургского государственного политехнического университета. Информатика. Телекоммуникации. Управление. 2017. Т. 10. № 4. С. 7–16. DOI: 10.18721/JCSTCS.10401
4. Лаптев В.В. Проектные основы инфографики. М.: АВАТАР, 2016. 287 с.
5. Karsten K.G. Charts and graphs: An introduction to graphic methods in the control and analysis of statistics. New York: Prentice-Hall, 1923. XI. 734 p.
6. Eells W.C. The relative merits of circles and bars for representing component parts // J. of the American Statistical Association. 1926. Vol. 21. Pp. 119–132.
7. Croxton F.E., Stein H. Graphic comparisons by bar, squares, circles, and cubes // J. of the American Statistical Association. 1932. Vol. 27. No. 177. Pp. 54–60.
8. Cleveland W.S., McGill R. Graphical perception: Theory, experimentation, and application to the development of graphical methods // J. of the American Statistical Association. 1984. Vol. 79. No. 387. Pp. 531–554.
9. Cleveland W.S. The elements of graphing data. NJ: Hobart Press, 1994. 297 p.
10. Spence I., Lewandowsky S. Displaying proportions and percentages // Applied Cognitive Psychology. 1991. Vol. 5. Pp. 61–77.
11. Kosara R., Skau D. Judgment error in pie chart variations // Proc. of Eurographics Conf. on Visualization (EuroVis). 2016 // URL <http://dx.doi.org/10.2312/eurovisshort.20161167> DOI: 10.2312/eurovisshort.20161167
12. Cai X., Efstathiou K., Xie. X., Wu Y. A study of the effect of doughnut chart parameters on proportion estimation accuracy // Computer Graphics Forum. 2018. Vol. 00. Pp. 1–13. DOI: 10.1111/cgf.13325
13. Bertin J. Semiology of graphics. Diagrams. Networks. Maps. Redlans: Esri Press, 2011. 438 p.
14. Лаптев В.В., Орлов П.А. Кластерный анализ визуального восприятия структуры данных // Бизнес-информатика. 2015. № 3 (33). С. 34–43.
15. Sharif B., Maletic J.I. An empirical study on the comprehension of stereotyped UML class diagram

layouts // Proc. of the IEEE 17th Internat. Conf. on Program Comprehension. 2009. Pp. 268–272.

16. **Huang W.** Using eye tracking to investigate graph layout effects // Proc. of the 6th Asia-Pacific Symp. on Visualisation. 2007. Pp. 97–100.

17. **Goldberg J.H., Helfman J.I.** Comparing information graphics: A critical look at eye tracking // Proc. of the 3rd BELIV'10 Workshop: Beyond Time and Errors: Novel Evaluation Methods for Information Visualization. 2010. Pp. 71–78.

18. **Prats M., Garner S., Jowers I., McKay A., Pedreira N.** Interpretation of geometric shapes – an eye movement study // Proc. of the Symp. on Eye-Tracking Research & Applications. 2010. Pp. 243–250.

19. **Duchowski A.** Eye tracking methodology. 3rd ed. Springer, 2017. 408 p.

20. **Ярбус А.Л.** Роль движений глаз в процессе зрения. М.: Наука, 1965. 165 с.

21. **Митькин А.А.** Электроокулография в инженерно-психологических исследованиях. М.: Наука, 1974. 141 с.

Статья поступила в редакцию 10.03.2019.

REFERENCES

1. **Few S.C.** *Show me the numbers: Designing tables and graphs to enlighten*. 2 ed. Burlingame: Analytics press, 2012, XVIII, 351 p.

2. **Yeliseyeva I.I., Yuzbashev M.M.** *Obshchaya teoriya statistiki [General theory of statistics]*. Moscow: Finansy i statistika Publ., 2004, 655 p. (rus)

3. **Laptev V.V., Orlov P.A., Dragunova O.V.** Visualization of dynamic data structures with flow charts in web analytics. *St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems*, 2017, Vol. 10, No. 4, Pp. 7–16. DOI: 10.18721/JCSTCS.10401 (rus)

4. **Laptev V.V.** *Proyektnyye osnovy infografiki [Infographic Design Basics]*. Moscow: AVATAR Publ., 2016, 287 p. (rus)

5. **Karsten K.G.** *Charts and graphs: An introduction to graphic methods in the control and analysis of statistics*. New York: Prentice-Hall, 1923, XI, 734 p.

6. **Eells W.C.** The relative merits of circles and bars for representing component parts. *Journal of the American Statistical Association*, 1926, Vol. 21, Pp. 119–132.

7. **Croxtan F.E., Stein H.** Graphic comparisons by bar, squares, circles, and cubes. *Journal of the American Statistical Association*, 1932, Vol. 27, No. 177, Pp. 54–60.

8. **Cleveland W.S., McGill R.** Graphical perception: Theory, experimentation, and application to the development of graphical methods. *Journal of the American Statistical Association*, 1984, Vol. 79, No. 387, Pp. 531–554.

9. **Cleveland W.S.** *The elements of graphing data*. JNJ: Hobart Press, 1994, 297 p.

10. **Spence I., Lewandowsky S.** Displaying proportions and percentages. *Applied Cognitive Psychology*, 1991, Vol. 5, Pp. 61–77.

11. **Kosara R., Skau D.** Judgment error in pie chart variations. Proceedings of Eurographics Conference on Visualization (EuroVis). 2016. Available: <http://dx.doi.org/10.2312/eurovisshort.20161167> DOI: 10.2312/eurovisshort.20161167

12. **Cai X., Efstathiou K., Xie X., Wu Y.** A study of the effect of doughnut chart parameters on proportion estimation accuracy. *Computer Graphics Forum*, 2018, Vol. 00, Pp. 1–13. DOI: 10.1111/cgf.13325

13. **Bertin J.** *Semiology of graphics. Diagrams. Networks. Maps*. Redlans: Esri Press, 2011, 438 p.

14. **Laptev V.V., Orlov P.A.** Klasternyi analiz vizual'nogo vosprijatija struktury dannyh [Cluster analysis of visual perception of data structure]. *Business Informatics*, 2015, Vol. 3 (33), Pp. 34–43. (rus)

15. **Sharif B., Maletic J.I.** An empirical study on the comprehension of stereotyped UML class diagram layouts. *Proceedings of the IEEE 17th International Conference on Program Comprehension*, 2009, Pp. 268–272.

16. **Huang W.** Using eye tracking to investigate graph layout effects. *Proceedings of the 6th Asia-Pacific Symposium on Visualisation*, 2007, Pp. 97–100.

17. **Goldberg J.H., Helfman J.I.** Comparing information graphics: a critical look at eye tracking. *Proceedings of the 3rd BELIV'10 Workshop: Beyond Time and Errors: Novel Evaluation Methods for Information Visualization*, 2010, Pp. 71–78.

18. **Prats M., Garner S., Jowers I., McKay A., Pedreira N.** Interpretation of geometric shapes – an eye movement study. *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications*, 2010, Pp. 243–250.

19. **Duchowski A.** *Eye tracking methodology*. 3rd ed. Springer, 2017, 408 p.

20. **Yarbus A.L.** *Rol dvizheniy glaz v protsesse zreniya* [The role of eye movements in the process of vision]. Moscow: Nauka Publ., 1965, 165 p. (rus)

21. **Mitkin A.A.** *Elektrookulografiya v inzhenerno-psikhologicheskikh issledovaniyakh* [Electroculography in engineering and psychological research]. Moscow: Nauka Publ., 1974, 141 p. (rus)

Received 10.03.2019.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

ЕРМОЛОВА Татьяна Константиновна

ERMOLOVA Tatiana K.

E-mail: togatsbp@gmail.com

ИВАЩЕНКО Полина Денисовна

IVASCHENKO Polina D.

E-mail: deadpolonium@yandex.ru

ЛАПТЕВ Владимир Владимирович

LAPTEV Vladimir V.

E-mail: laptevsee@yandex.ru

DOI: 10.18721/JCSTCS.12203
УДК 004.05

МЕТОДИКА ВЫБОРА ШАБЛОНА ПРОГРАММНОЙ АРХИТЕКТУРЫ НА ОСНОВЕ АНАЛИЗА ПАРАМЕТРОВ КАЧЕСТВА СИСТЕМЫ

Ю.В. Ядгарова¹, В.В. Таратухин²

¹ Московский государственный технический университет имени Н.Э. Баумана,
Москва, Российская Федерация;

² Национальный исследовательский университет «Высшая школа экономики»,
Европейский исследовательский центр в области информационных систем (ERCIS),
Москва, Российская Федерация;

Предложен подход к выбору базового шаблона программной архитектуры и сопутствующих тактик проектирования, исходя из требуемых параметров качества систем. Проанализированы основные концепты и понятия предметной области, предложена предметная онтология. Рассмотрены связи между основными понятиями, приведены основные шаблоны программных архитектур. Изучено понятие тактик проектирования, а также их связь с шаблонами архитектур. Сформулирована задача построения программной архитектуры с заданными параметрами качества. В качестве решения предложена аналитическая модель выбора на основе минимизации трудоемкости проекта. Приведен подход к выбору базового шаблона архитектуры и тактик проектирования на примере систем класса превентивного обслуживания.

Ключевые слова: архитектура программного обеспечения, качество программного обеспечения, проектирование архитектуры ПО, параметры качества программных систем, тактики проектирования, шаблоны архитектуры, онтология проектирования ПО.

Ссылка при цитировании: Ядгарова Ю.В., Таратухин В.В. Методика выбора шаблона программной архитектуры на основе анализа параметров качества системы // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2019. Т. 12. № 2. С. 28–38. DOI: 10.18721/JCSTCS.12203

GENERAL APPROACH FOR SELECTION SOFTWARE ARCHITECTURE PATTERN BASED ON SYSTEM QUALITY ATTRIBUTES

Yu.V. Yadgarova¹, V.V. Taratukhin²

¹ Bauman Moscow State Technical University,
Moscow, Russian Federation;

² National Research University Higher School of Economics,
European Research Center for Information Systems (ERCIS),
Moscow, Russian Federation

The paper presents the approach of selection general architecture pattern and design tactics for the software project. Proposed method is based on the satisfying required software quality attributes. The domain ontology is defined along with the main concepts and structure. Base software architectural patterns and main design tactics were analyzed and as a result the problem of building software architecture with required quality attributes has stated. As a solution, analytical model of selecting necessary tactics has presented. The general approach to select the best software architecture pattern and tactics is defined and described with the example of preventive maintenance systems.

Keywords: software architecture, software quality, software architecture design, software systems quality attributes, design tactics, software architectural patterns, ontology of the software architecture design.

Citation: Yadgarova Yu.V., Taratukhin V.V. General approach for selection software architecture pattern based on system quality attributes. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2019, Vol. 12, No. 2, Pp. 28–38. DOI: 10.18721/JCSTCS.12203

Введение

Термин «Программная архитектура» относится к процессу принятия проектных решений при разработке информационных систем для удовлетворения специфических требований. Принимаемые на этапе проектирования архитектуры решения являются компромиссом между параметрами качества разрабатываемой системы и основаны на функциональных требованиях к системе. Решения влияют на качество системы в целом, которое она будет демонстрировать.

Организация Institute of Electrical and Electronics Engineers (IEEE) определяет программную архитектуру как «фундаментальную организацию систем, воплощенную в их компонентах, их связях друг с другом и средой и принципах, руководящих проектированием и эволюцией» [1].

При обсуждении архитектуры специфических систем проектировщики абстрагируются от деталей реализации и фокусируются на решениях, которые необходимо сделать для достижения определенных показателей системы. Эти решения в конечном итоге влияют на то, как система будет декомпозирована на компоненты, и как они взаимодействуют друг с другом. Если подобные решения начинают применяться ко всем системам определенного типа для достижения похожих целей, данные архитектурные шаблоны становятся видимыми. В некоторых источниках их называют *архитектурными типами*, или *стилями*. Детальным исследованиям шаблонов архитектуры ПО посвящены работы [2, 3]. Один из методов оценки применимости архитектурных стилей и шаблонов рассмотрен в [4], а сравнение различных методов оценки стилей приведено в [5]

Соответственно, *тип архитектуры* можно определить как абстракцию программной архитектуры. Другими словами, программ-

ная архитектура является конкретизацией одного или множества архитектурных типов.

Тип архитектуры определяется четырьмя основными свойствами.

1. Множество архитектурных элементов, из которых состоит решение. В качестве примера приведем клиент-серверную программную архитектуру, где элементами соответственно являются машины и ПО клиента и сервера.

2. Структура системы – множество правил и топографических констант, которые определяют допустимые композиции элементов в системе.

3. Семантическая интерпретация композиции элементов, что означает наличие четко прослеживаемого значения у каждой определенной связки элементов.

4. Понимание ограничений и допущений, применимых к данному типу. Каждый тип имеет преимущества и недостатки, которые могут быть критичными или некритичными для определенных видов систем. На основе данных компромиссов чаще всего определяются необходимые параметры качества для реализации в системе.

Элементы процесса проектирования конкретной программной архитектуры с учетом указанных основ в общем случае показаны на рис. 1.

Говоря о структурных элементах типов программных архитектур, выделяют понятие *шаблонов архитектуры*, представляющих собой компоненты и связи между ними (элементы в терминах типа архитектуры). Другими словами, шаблон архитектуры представляет собой абстракцию структуры программной системы, состоящую из компонентов и связей между ними. В литературе нередко используется смешение этих определений, и более широкое понятие типа архитектуры применяют к шаблонам.

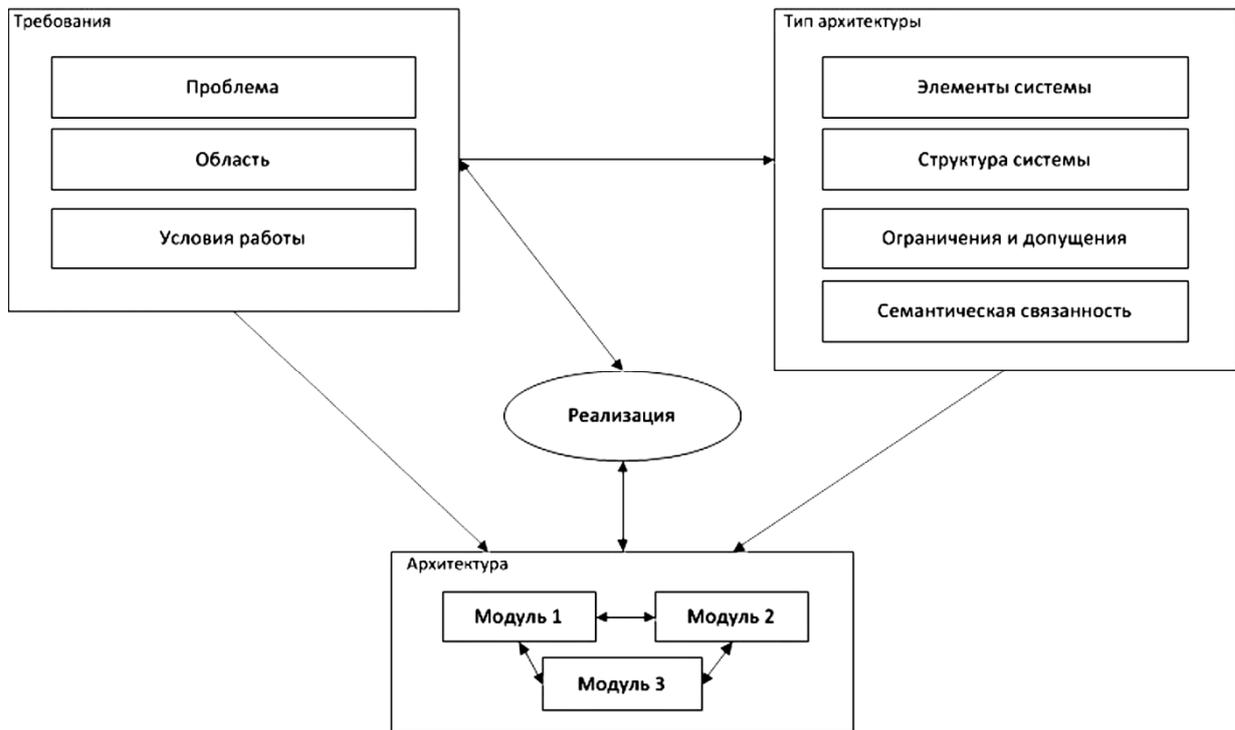


Рис. 1. Элементы процесса проектирования программной архитектуры
Fig. 1. Elements of the software architecture design process

Приведем некоторые основные архитектурные шаблоны:

- клиент-серверная архитектура;
- архитектура «Точка-точка»;
- архитектура «Каналы и фильтры»;
- событийно-ориентированная архитектура;
- архитектура «Издатель-подписчик»;
- сервисно-ориентированная архитектура (SOA);
- REST-архитектура;
- архитектура слоев;
- облачная архитектура.

Обращаясь к реализации существующих программных архитектур, можно утверждать, что каждый шаблон архитектуры может по построению удовлетворять некоторым параметрам качества (примером может служить хорошая масштабируемость клиент-серверной архитектуры). Для остальных параметров введем понятие *тактики проектирования*. Тактика проектирования (типовой сценарий) – проекционное решение, воздействующее на до-

стижение определенного параметра качества [6]. Реализация тактики всегда меняет ответ системы на конкретное воздействие.

Таким образом, параметры качества программных систем напрямую зависят от исходного шаблона программной архитектуры, а также от реализованных тактик проектирования. Для достижения определенного множества параметров качества программной системы необходима реализация базового шаблона архитектуры и известного количества тактик. В то же время вопрос трудоемкости и достижимости тактик для каждого типа архитектуры остается открытым.

Предметная онтология проектирования программной архитектуры

Для программных систем будем придерживаться определения онтологии, представленного в работе [7]. Согласно этому определению, онтология является эксплицитной (явной) спецификацией концептуализации знаний. Формальную модель

предметной онтологии записывают в виде $O_d = \langle C, F, R \rangle$, где C – конечное множество концептов (понятий) предметной области, R – конечное множество отношений между концептами, F – конечное множество функций интерпретации (аксиоматизация), заданных на концептах и (или) отношениях предметной онтологии O_d .

В зависимости от множеств C, F, R , онтологии могут вырождаться в частные виды, классификация которых приведена в работе [8]. Например, при $R = \emptyset$ и $F = \emptyset$ предметная онтология трансформируется в словарь.

Для описания предметной онтологии проектирования архитектуры будем использовать язык моделирования UML 2 с различными типами отношений. Множество аксиом применительно к данной предметной онтологии будет формироваться далее в модели.

В качестве базовых понятий онтологии рассмотрим ее составные части. Множество концептов онтологии проектирования архитектуры включают в себя:

- архитектуру системы;
- особенность проектирования;
- тактику проектирования;
- параметр качества;
- шаблоны архитектуры;
- элементы шаблона архитектуры;
- реализацию шаблона архитектуры;
- особенность архитектуры системы;
- тактику проектирования;
- параметр качества;

- тактики проектирования;
- особенности архитектуры системы;
- параметры качества;
- шаблоны архитектуры;
- элементы шаблона архитектуры;
- компоненты;
- связи.

Отношения между указанными выше концептами можно определить через нотацию UML 2. Представление онтологии в терминах диаграммы классов приведено на рис. 2.

Как видно из диаграммы, при моделировании предметной области используются отношения реализации, зависимости, композиции и агрегации.

Архитектура системы связана отношением агрегации с реализацией шаблона архитектуры (т. к. реализация может существовать вне контекста конкретной программной архитектуры).

Реализация шаблона архитектуры в свою очередь связана с интерфейсом шаблона архитектуры, который зависит от особенностей архитектуры системы и содержит (отношение композиции) элементы шаблона архитектуры.

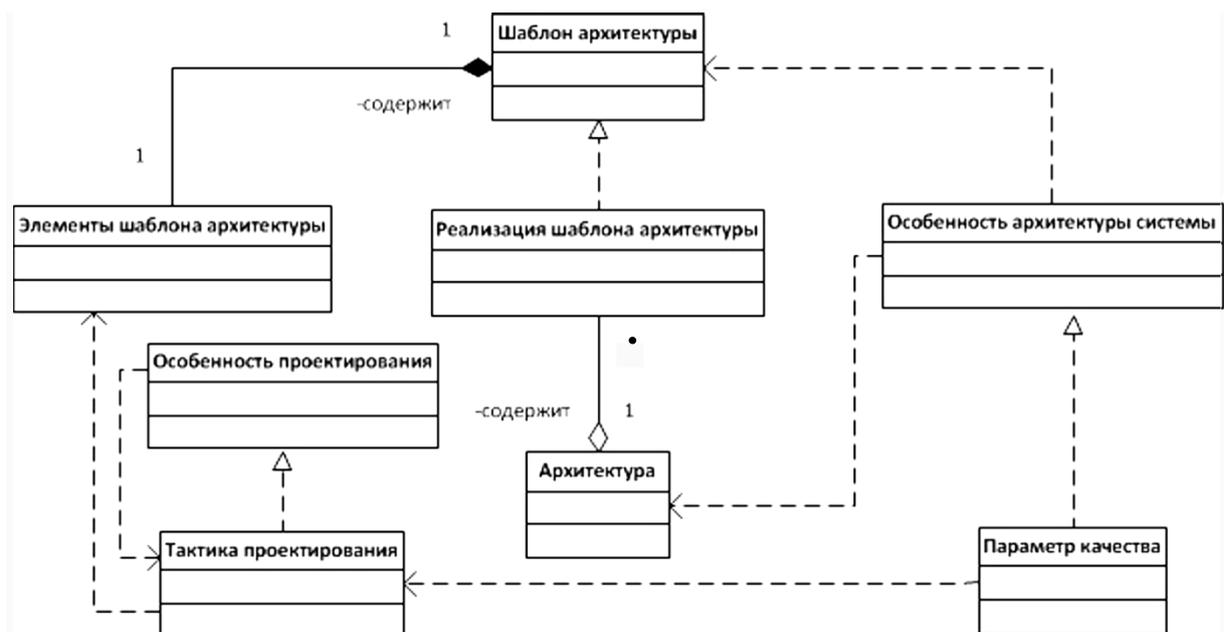


Рис. 2. Структура понятий (диаграмма классов)

Fig. 2. Concept structure (class diagram)

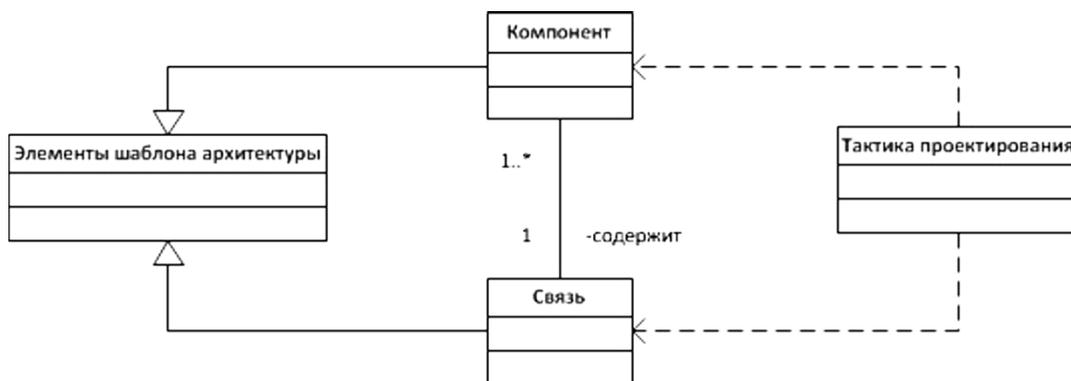


Рис. 3. Элементы шаблона архитектуры (диаграмма классов)

Fig. 3. Elements of the architecture template (class diagram)

На элементы шаблона архитектуры влияют выбранные тактики проектирования, реализующие особенности проектирования и зависящие от них.

Тактики проектирования зависят от необходимых параметров качества системы, которые реализуют особенность архитектуры системы.

Говоря о структуре понятий в области проектирования архитектуры применительно к данной работе, необходимо обратить особое внимание на связь элементов шаблона архитектуры и тактик проектирования. Структурная декомпозиция указанных понятий приведена на рис. 3.

Компоненты архитектуры и связи между ними являются элементами шаблона

архитектуры (отношение обобщения). Тактика проектирования влияет на архитектуру путем влияния на компоненты шаблона и связи между ними.

Здесь следует отдельно указать, что часть данной работы – конкретизация отношения зависимости компонентов и связей между ними для различных тактик проектирования. Для этого было введено понятие *типа изменения шаблона тактикой*, отражающее сущность модификации шаблона архитектуры и, следовательно, влияющее на трудоемкость. Каждая тактика может вносить от одного до нескольких типов изменений. Различные типы изменений приведены в табл. 1.

Таблица 1

Типы изменения шаблона тактикой

Table 1

Types of pattern change tactics

Тип изменения	Трудоемкость	Описание
Добавление компонента в шаблон Доб _{кш}	Низкая	Добавление нового совместимого компонента в шаблон, связь с другими. Структура остается прежней
Модификация компонента Мод _к	Средняя - Высокая	Меняется структура компонентов шаблона. Влияет на остальные компоненты
Реализация внутри компонента Вн _к	Низкая	Меняется один из компонентов шаблона
Дублирование компонента Дуб _к	Низкая	Дублируется один из компонентов, добавляются связи
Добавление компонента вне шаблона Доб _{квш}	Высокая	Добавляется компонент, по структуре не подходящий шаблону

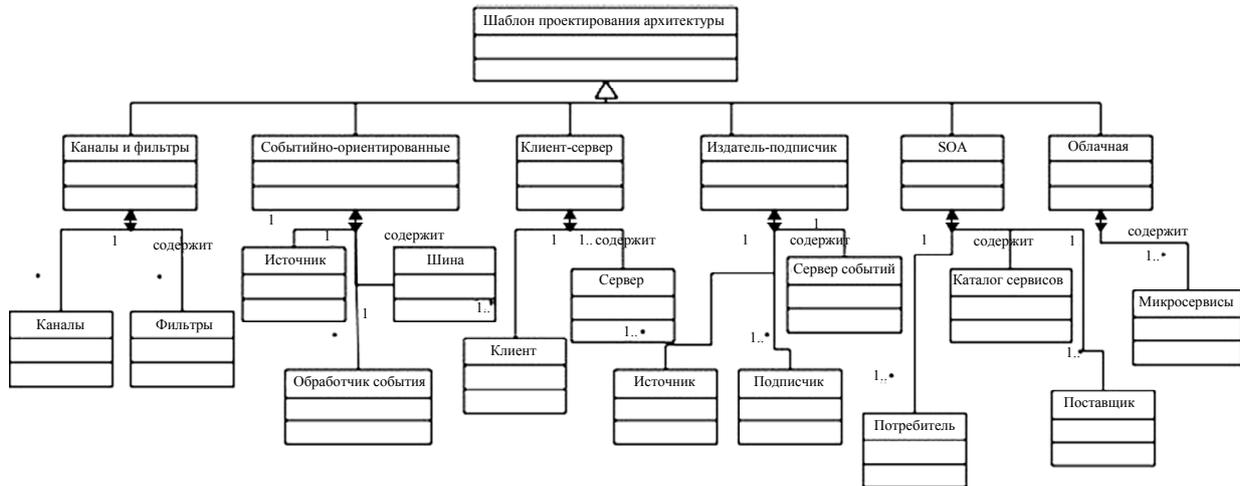


Рис. 4. Метасистема понятий шаблонов архитектуры

Fig. 4. Metasystem of concepts of architecture patterns

В зависимости от типа изменения шаблона может существенно изменяться трудоемкость реализации тактики проектирования, что будет показано далее.

Тип изменения шаблона архитектуры является понятием, зависящим от конкретного шаблона архитектуры. Определение типа полностью зависит от выбранного базового шаблона. Таким образом, реализация тактик проектирования для разных шаблонов будет иметь различные типы изменения и, следовательно, различную трудоемкость.

Говоря о таксономии объектов, необходимо выделить отдельно видовые отношения шаблонов архитектур. На рис. 4 показана метасистема понятий некоторых шаблонов архитектуры, включающая различные виды и элементы указанных шаблонов.

Каждый из специфических шаблонов архитектуры содержит элементы, зависящие от вида шаблона. Так, элементами в шаблоне SOA являются Потребитель сервиса, Каталог сервисов и Поставщик.

Модель выбора базового шаблона архитектуры ПО и тактик проектирования

Задача выбора базового шаблона архитектуры и тактик для программных систем состоит в нахождении такой совокупности компонентов и связей между ними для ба-

зового шаблона и множества тактик, реализация которых приводит к достижению требуемых функциональных параметров и параметров качества системы при минимизации трудоемкости.

Полагаем, что критерием достижения параметра качества программной системы является реализация в выбранной архитектуре множества тактик проектирования $T = t_i, i = 1, 2, \dots, n$, отвечающих за упомянутый параметр качества. При этом в литературе подчеркивается, что некоторые шаблоны программных архитектур по своему исходному построению удовлетворяют определенным параметрам качества. Данное условие выразим с помощью лингвистической переменной K_{ij} , определяющей соответствие базового шаблона архитектуры i требуемому параметру качества j :

$$K_{ij} = \begin{cases} BC & \left| \begin{array}{l} \text{тип архитектуры удовлетворяет} \\ \text{по параметру качества,} \end{array} \right. \\ CP & \left| \begin{array}{l} \text{тип архитектуры нейтрален} \\ \text{к параметру качества,} \end{array} \right. \\ HЗ & \left| \begin{array}{l} \text{тип архитектуры не удовлетворяет} \\ \text{по параметру качества.} \end{array} \right. \end{cases}$$

В случае если базовый шаблон архитектуры содержит высокую оценку по требуемому параметру качества, реализация тактик проектирования для данного парамет-

ра не требуется, что в свою очередь представим булевой переменной x_{ij} , значение которой равно:

$$x_{ij} = \begin{cases} 1, & \text{реализация тактик для параметра качества не требуется,} \\ 0, & \text{реализация тактик для параметра качества необходима.} \end{cases}$$

Информация о соответствии базовых типов архитектур некоторым параметрам качества приведена в табл. 2.

В качестве оценки трудоемкости среди множества аналитических методик оценки выбрана наиболее применимая к начальной стадии проекта: СОСОМО II (СОnstructive СОst МОdel – Конструктивная модель стоимости) [9, 10].

Для решения задачи выбора шаблона архитектуры и тактик проектирования, введем понятие *сети параметров качества* системы. Им будем называть совокупность

параметров качества, зависящую от типа системы и требований $S_t = S\{i, t\}$, где $i = 1..k$ – параметр качества, $t = 1..n$ – тип системы. Каждый параметр качества характеризуется либо реализацией его в базовом шаблоне архитектуры, либо суммой тактик проектирования, необходимых для его достижения на базовом шаблоне:

$$S_t \left\{ \begin{array}{l} x_{ij}, \\ \sum_{k=1}^m T_k, \end{array} \right.$$

где T_k – тактика проектирования; m – количество тактик, необходимое для реализации параметра; $x_{ij} = 1$, если реализация тактики проектирования для достижения не требуется; $x_{ij} = 0$ в обратном случае.

Тактика проектирования может быть представлена как модификация элементов архитектурного шаблона (компонентов и связей между ними).

Таблица 2

Соответствие типов архитектур некоторым параметрам качества

Table 2

Correspondence of types of architectures to some quality parameters

Типы архитектур	Интероперабельность	Модифицируемость	Производительность	Масштабируемость	Надежность	Информационная безопасность
Клиент-сервер	1	1	0	0	0	0
Точка-точка	0	0	0	1	0	0
Каналы и фильтры	0	0	1	0	1	0
Событийно-ориентированная	0	1	0	0	0	0
Издатель-подписчик	0	1	0	0	0	0
Сервисно-ориентированная	1	1	0	0	0	0
Брокер	1	1	0	0	0	0
Архитектура слоев	0	0	0	0	0	0
Облачная	1	1	1	1	0	0

Однако реализация одних и тех же тактик проектирования для различных базовых шаблонов архитектуры может существенно отличаться, а, следовательно, отличаться будет и трудоемкость реализации. Для того чтобы учесть данное ограничение, введем понятие типа изменения шаблона архитектуры тактикой. Подсчитаем с помощью методики СОСОМО II относительную трудоемкость реализации добавления компонента в базовый шаблон архитектуры, состоящий из n компонентов.

Трудоемкость проекта задается формально как функция зависимости от количества тысяч строк кода:

$$Q = \alpha \cdot (KLOC)^b \cdot EAF, \quad (1)$$

где Q – трудозатраты, выраженные в человеко-месяцах; EAF – фактор корректировки трудозатрат в зависимости от среды; α – фактор, зависящий от детальности оценки (постоянная величина, для предварительной оценки равная 2,94); $b = B + 0,01 \cdot \sum_{j=1}^5 SF_j$,

где SF_j – факторы масштаба. Тогда исходную трудоемкость реализации можно представить как $Q = \alpha \cdot (KLOC_0)^b \cdot EAF$. Так как детальность оценки и среда остаются неизменными, реализация дополнительного компонента зависит от количества тысяч строк кода $KLOC$ реализации и фактора масштаба b . Трудоемкость готового проекта, выраженная через сумму трудоемкостей компонентов и связей между ними:

$$\begin{aligned} Q &= \sum_{i=1}^n Q_i + \sum_{j=1}^m Q_j = \\ &= \sum_{i=1}^n \alpha \cdot KLOC_i^b \cdot EAF + \sum_{j=1}^m \alpha \cdot KLOC_j^b \cdot EAF = \quad (2) \\ &= \alpha \cdot EAHF \cdot \left(\sum_{i=1}^n \alpha \cdot KLOC_i^b + \sum_{j=1}^m \alpha \cdot KLOC_j^b \right). \end{aligned}$$

Следует отметить, что существенным допущением при данном преобразовании является то, что при реализации каждого последующего компонента фактор масштаба остается неизменным.

Выведем трудоемкость разработки одного модуля через трудоемкость всего проекта (считаем количество модулей):

$Q_i = \alpha \cdot (KLOC_M / n)^b \cdot EAF$, где n – количество модулей в проекте, $KLOC_M$ – количество тысяч строк кода на реализацию всех модулей. Тогда, принимаем $t = KLOC_M / n$, где t – усредненное количество строк кода на один модуль. При этом аналогично, $Q_j = \alpha \cdot (KLOC_C / m)^b \cdot EAF$, $t_{cb} = KLOC_C / m$, где $KLOC_C$ – количество тысяч строк кода на связь между модулями, m – количество таких связей. Тогда, переписывая выражение (1), получаем трудоемкость проекта как

$$Q = \alpha \cdot EAF \cdot (n \cdot t^b + m \cdot t_{cb}^b). \quad (3)$$

Возвращаясь к типам изменения шаблона архитектуры, в ходе исследования было выделено пять основных способов модификации. Данные способы приведены в табл. 2 и отражают характер изменений. Приведенные значения трудоемкости выражены как функция от величин t и t_{cb} . Например, расчет величины $\Delta_{доб}$ (Добавление нового совместимого компонента в шаблон) заключается в следующем. При наличии базового шаблона архитектуры с известными величинами n и t и t_{cb} добавление еще одного компонента меняет известную трудоемкость (3) как минимум (с наличием хотя бы одной связи) на величину равную:

$$\Delta_{доб} = \alpha \cdot EAF \cdot (t^b + t_{cb}^b).$$

При типе изменений Мод_k (Модификация компонента) известная трудоемкость (2) изменяется как

$$\Delta_{\text{мод}} = \alpha \cdot EAF \cdot n \cdot (t_{\text{мод}})^b,$$

где $t_{\text{мод}}$ – изменение количества кода на модуль. Учитывая, что количество модулей и изменения в них могут быть существенными, трудоемкость модификации компонентов значительна. На основе расчета для каждого типа изменения построен пример графика зависимости величины трудоемкости изменения от t , принимая $t_{cb} = 0,2t$ (рис. 5). При количестве модулей равном 10, а $t_{\text{мод}} = 0,5t$ график принимает вид, показанный на рисунке.

Трудоёмкость добавления модуля

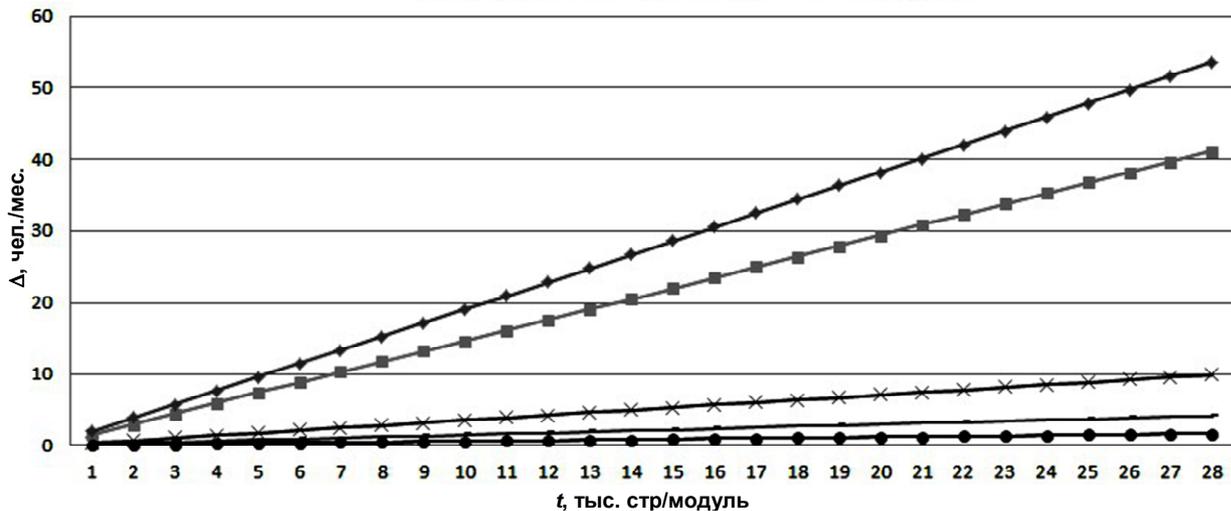


Рис. 5. Трудоёмкость изменения шаблона архитектуры
 (—×—) — Δдоб_ш; (—■—) — Δмод; (—) — Δвн; (—●—) — Δдуб; (—▲—) — Δдоб_вн_ш

Fig. 5. The complexity of changing the architecture template

Таким образом, общая трудоёмкость проекта для выбранного базового типа состоит из базовой относительной трудоёмкости реализации шаблона (выраженной через величины t и t_{cb}) и суммы относительных трудоёмкостей n тактик проектирования:

$$Q = Q_0(t, t_{cb}) + \sum_{i=1}^n Q_i(t, t_{cb}, m) = \alpha \cdot EAF \cdot \left(n \cdot t + m \cdot t_{cb} + \sum_{i=1}^n Q_i(t, t_{cb}, m) \right), \quad (4)$$

где r — тип изменения по табл. 2. Величины t и t_{cb} являются зависящими как от шаблона архитектуры, так и от реализации компонента, и рассчитываются с помощью метода функциональных точек для каждого конкретного проекта. Следовательно, задача сводится к нахождению такой комбинации базового шаблона архитектуры и тактик, при которой относительная величина трудоёмкости является минимальной: $Q \rightarrow \min$.

В табл. 3 представлен пример относительной оценки шаблона архитектуры «Брокер», выполненный для систем превентивного обслуживания.

В число необходимых достигаемых параметров качества для систем Интернета

Вещей указанного типа входят надёжность, производительность, модифицируемость и информационная безопасность как первичные параметры. Особенностью выбранного шаблона архитектуры является высокий уровень модифицируемости и интероперабельности системы (см. табл. 2). При этом для достижения параметров качества необходимо реализовать дополнительно тактики, адресующие надёжность, производительность и безопасность. Часть списка данных тактик с относительной стоимостью реализации ($n = 5$, $t_{mod} = 0,2t$, $t_{cb} = 0,1t$) указан в табл. 3.

В столбце «Тип изменения шаблона» указаны соответствующие типы изменений, необходимые для реализации тактики проектирования. Относительная стоимость соответственно подсчитана как стоимость изменения трудоёмкости в зависимости от типа.

Подставляя данные значения в выражение (3), получаем выражение для относительной оценки трудоёмкости $Q = \alpha \cdot EAF(n \cdot t + m \cdot 0,2 \cdot t + 33,863 \cdot t)$, которое можно использовать в оценке и выборе результирующей архитектуры.

Таблица 3

Оценка шаблона архитектуры «Брокер»

Table 3

Evaluation of the Broker architecture pattern

Тактика	Тип изменения шаблона	Относительная стоимость
Увеличение вычислительной эффективности (П)	Мод _к	1,19t
Уменьшение накладных расходов (П)	Мод _к	1,19t
Управление темпом событий (П)	Доб _{кш}	3,857t
Распараллеливание (П)	Мод _к , Дуб _к	1,428t
Копии данных (П)	Дуб _к	0,238t
Улучшение доступных ресурсов (П)	Мод _к	1,19t
Политика планирования (П)	Доб _{кш} , Мод _к	5,047t
Пинг-эхо (Н)	Мод _к	1,19t
Обработка исключений (Н)	Мод _к	1,19t
Голосование (Н)	Дуб _к , Доб _{кш}	4,095t
Активная избыточность (Н)	Дуб _к	0,238t
Пассивная избыточность (Н)	Дуб _к , Мод _к	1,458t
Повторное включение компонента: затемнение (Н)	Дуб _к , Мод _к	1,458t
Аутентификация пользователей (ИБ)	Доб _{кш}	3,857t
Авторизация пользователей (ИБ)	Доб _{кш}	3,857t
Сохранение конфиденциальности данных (ИБ)	Мод _к	1,19t
Сохранение целостности (ИБ)	Мод _к	1,19t

Заключение

В настоящей статье продемонстрирован подход к проектированию программной архитектуры, основанный на достижении требуемых параметров качества систем. После произведенного анализа проблемы построена онтология предметной области, а также приведена модель достижения требуемых параметров с использованием комбинаций архи-

тектурного шаблона и необходимых тактик проектирования. В качестве критерия выбора шаблона и тактик в модели используется трудоемкость проектирования системы, рассчитанная по методике СОСОМО II. На основе предложенной аналитической модели демонстрируется выбор базового шаблона архитектуры и тактик проектирования для систем класса превентивного обслуживания.

СПИСОК ЛИТЕРАТУРЫ

1. Coplien J.O., Bjørnvig G. Lean architecture: For agile software development. John Wiley & Sons, 2011 // URL: http://www.spooch.dk/Ebooks/IT%20Various/lean_architecture.pdf
2. Midwinter A., et al. Software architectural styles // URL: <https://slideplayer.com/slide/8014475/>
3. Pahl C., Giesecke S., Hasselbring W. An ontology-based approach for modelling architectural styles // ECSA. 2007. Vol. 4758. Pp. 60–75.
4. Kazman R., et al. The architecture tradeoff analysis method // Proc. of the 4th IEEE Internat. Conf. on Engineering of Complex Computer Systems. IEEE, 1998. Pp. 68–78.
5. Maurya L.S. Comparison of software architecture evaluation methods for software quality attributes // J. of Global Research in Computer Science. 2010. Vol. 1. No. 4.
6. Clements P., Kazman R., Bass L. Software architecture in practice. 3rd Ed.
7. Gruber T.R. The role of common ontology in achieving sharable, reusable knowledge bases // Proc. of the 2nd Internat. Conf. on Principles of Knowledge Representation and Reasoning. Morgan Kaufmann, 1991. Pp. 601–602.
8. Евгеньев Г.Б. Интеллектуальные системы проектирования. 2012. С. 416.

9. **Boehm B.W., et al.** Software engineering economics. Englewood Cliffs (NJ): Prentice-hall, 1981. Vol. 197.

10. **Boehm B., et al.** Cost models for future software life cycle processes: COCOMO 2.0 // Annals of software engineering. 1995. Vol. 1. No. 1. Pp. 57–94.

Статья поступила в редакцию 20.03.2019.

REFERENCES

1. **Coplien J.O., Bjørnvig G.** *Lean architecture: For agile software development*. John Wiley & Sons, 2011. Available: http://www.spooch.dk/Ebooks/IT%20Various/lean_architecture.pdf

2. **Midwinter A., et al.** *Software architectural styles*. Available: <https://slideplayer.com/slide/8014475/>

3. **Pahl C., Giesecke S., Hasselbring W.** An ontology-based approach for modelling architectural styles. *ECSA*, 2007, Vol. 4758, Pp. 60–75.

4. **Kazman R., et al.** The architecture tradeoff analysis method. *Proceedings of the 4th IEEE International Conference on Engineering of Complex Computer Systems*, IEEE, 1998, Pp. 68–78.

5. **Maurya L.S.** Comparison of software architecture evaluation methods for software quality attributes. *Journal of Global Research in Computer Science*, 2010, Vol. 1, No. 4.

6. **Clements P., Kazman R., Bass L.** *Software architecture in practice*. 3rd Ed.

7. **Gruber T.R.** The role of common ontology in achieving sharable, reusable knowledge bases. *Proceedings of the 2nd International Conference on Principles of Knowledge Representation and Reasoning*. Morgan Kaufmann, 1991, Pp. 601–602.

8. **Evgenev G.B.** *Intellektualnyye sistemy proyektirovaniya [Intelligent Design Systems]*. 2012, P. 416. (rus)

9. **Boehm B.W., et al.** *Software engineering economics*. Englewood Cliffs (NJ): Prentice-hall, 1981, Vol. 197.

10. **Boehm B.W., et al.** Cost models for future software life cycle processes: COCOMO 2.0. *Annals of software engineering*, 1995, Vol. 1, No. 1, Pp. 57–94.

Received 20.03.2019.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

ЯДГАРОВА Юлия Владимировна

YADGAROVA Yulia V.

E-mail: y.v.yadgarova@gmail.com

ТАРАТУХИН Виктор Владимирович

TARATUKHIN Victor V.

E-mail: vtaratoukhine@hse.ru

DOI: 10.18721/JCSTCS.12204
УДК 004.942

МНОГОКРИТЕРИАЛЬНАЯ ОПТИМИЗАЦИЯ РЕКТИФИКАЦИОННОГО ПРОЦЕССА ПО АЛГОРИТМУ SPEA2

Д.А. Костенко, В.А. Онуфриев, В.П. Шкодырев

Санкт-Петербургский политехнический университет Петра Великого,
Санкт-Петербург, Российская Федерация

Описан процесс многокритериальной оптимизации по методу Парето-оптимальности. В качестве объекта управления выступает крупное промышленное предприятие. Объект декомпозируется и представляется в виде иерархии вложенных орграфов. Вершины орграфов отмечают состояние производимого продукта, рёбра – технологические операции. На основании технической документации по объекту составляются списки целевых показателей и факторов управления, помогающие систематизировать источники влияния на качество и объем производимого продукта. Нейронная сеть, обученная на архиве статистических данных, идентифицирует связи между параметрами выходного продукта и источниками влияния. Полученные связи дискретизируются по времени и подаются на вход алгоритма SPEA2. Алгоритм производит сравнение зависимостей и выстраивает Парето-оптимальный фронт, состоящий из комбинаций значений управляемых параметров.

Ключевые слова: Парето-оптимальность, многокритериальная оптимизация, идентификация, SPEA2, нейронная сеть.

Ссылка при цитировании: Костенко Д.А., Онуфриев В.А., Шкодырев В.П. Многокритериальная оптимизация ректификационного процесса по алгоритму SPEA2 // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2019. Т. 12. № 2. С. 39–49. DOI: 10.18721/JCSTCS.12204

MULTICRITERIAL OPTIMISATION OF RECTIFICATIONAL PROCESS BASED ON SPEA2 ALGORITHM

D.A. Kostenko, V.A. Onufriev, V.P. Shkodyrev

Peter the Great St. Petersburg Polytechnic University,
St. Petersburg, Russian Federation

This article describes the process of multicriterial optimization using the Pareto efficiency method. A large-scale industrial plant was taken as a controllable object. The object was decomposed and represented as a hierarchy of embedded orgraphs. The orgraph's vertices mark the current state of the product while the edges stand for technological operations. Based on the object's technical documentation, a list of

influencing factors is created. The list contains every technological parameter affecting the quality of the final product. A neural network trained on a set of statistical data is utilized to identify dependencies between discrete influencing factors and the product quality. These dependencies are then processed with the SPEA2 algorithm, outputting a set of combinations of optimized parameters values known as the Pareto front.

Keywords: Pareto efficiency, multicriterial optimisation, identification, SPEA2, neural network.

Citation: Kostenko D.A., Onufriev V.A., Shkodyrev V.P. Multicriterial optimisation of rectificational process based on SPEA2 algorithm. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2019, Vol. 12, No. 2, Pp. 39–49. DOI: 10.18721/JCSTCS.12204

Введение

Говоря о многокритериальных задачах оптимизации, мы имеем в виду решение задач в условиях неопределенности. При этом выбор вариантов осуществляется по группе оценок, находящихся в противоречии друг с другом [1].

Необходимость оптимизировать несколько ключевых показателей качества (Key Performance Indicators – KPI) одновременно возникает в различных сферах человеческой деятельности, в том числе в нефтеперерабатывающей промышленности [2]. Ввиду сложности промышленных процессов и неопределенности условий принятия решений [3] оптимизации предшествует идентификация [4], в целях которой процессы могут быть декомпозированы [5].

Для исключения из рассмотрения заведомо неконкурентоспособных вариантов решения широко применяется принцип Парето. Однако выбор окончательного результата осуществляется на основе дополнительной информации о предпочтениях лица, принимающего решения [6, 7].

Частным примером процесса, характеризующего группой параметров, является перегонка нефти. Она представляет собой многоступенчатое разложение сырья на фракции под воздействием различных комбинаций давления и температуры. Процесс частично протекает в ректификационной установке (РУ), одна из которых была взята за прототип. Для модели РУ были назначены следующие KPI:

- качество (степень соответствия выходного продукта установленным нормам);

- производительность (объем выхода продукта);

- рентабельность (экономическая эффективность производства продукта);

- надежность (количество отказов оборудования на единицу времени);

- безопасность (количество аварий на единицу времени).

Ректификация описывается значительным количеством параметров: до нескольких сотен характеристик на одну ректификационную установку. Сюда входят температура в отгонной секции, шлеме и на тарелках колонны; давление в зоне питания, давление в отгонной секции и т. д. Внутри и снаружи колонны протекают как последовательные (многоступенчатая очистка сырья, разогрев, прокачка), так и параллельные (конденсация фракций на разных тарелках) процессы. Включение промежуточных продуктов в технологию перегонки, как в случае с орошением колонны конденсированным ректификатом, создают дополнительный – горизонтальный уровень иерархии между операциями. Необходимо также учесть временную задержку, вносимую инерционностью системы, и непрерывный режим работы установки.

Таким образом, процессы, находящиеся на верхнем уровне иерархии, имеют неочевидную связь с процессами более нижних уровней, а зависимости между ними не могут быть представлены функциями вида $y = f(g, u)$. Однако для управления установкой требуется влиять на процессы верхнего уровня, изменяя параметры на нижнем и, наоборот, влиять на процессы

нижнего уровня, изменяя параметры на верхнем.

В данной статье описанная выше проблема решается путем декомпозиции сложной системы (ректификационной установки) вплоть до отдельных узлов и процессов. Полученная структура представляется в виде графа. Вершиной созданной иерархии становится набор КРІ, ключевых показателей. Каждый из них разбивается на несколько слагаемых, находящихся уровнем ниже. Процесс деления продолжается до тех пор, пока достигнутый уровень иерархии не будет содержать однозначно интерпретируемой зависимости вида $y = f(x)$. Идентификация зависимости выполняется при помощи нейронной сети, обученной на архиве статистических данных по работе установки. При подъеме на следующий уровень зависимость примет вид $y = g(f(x))$. Поднимаясь вверх по полученному дереву, можно определить зависимость между изменением КРІ и входного параметра из основания иерархии.

При этом ключевые параметры эффективности, расположенные на вершине, могут противоречить друг другу. В частности, повышение производительности за счет введения более агрессивных рабочих параметров процесса неизбежно повлечет рост

количества отказов и общее снижение надежности, что в итоге может снизить показатель рентабельности не только конкретной установки, но и всего завода. Антагонизм вида «дешево – быстро – качественно» приводит к необходимости применения алгоритма многокритериальной оптимизации, обеспечивающего баланс между конфликтующими КРІ установки.

Цель нашего исследования – построение математической модели RECTИФИКАЦИОННОГО процесса и многокритериальная оптимизация с применением принципа Парето.

Для анализа использованы статистические данные работы нефтеперегонной установки действующего нефтеперерабатывающего завода (НПЗ). На основании статистики базовых параметров перегонного процесса составлены графические представления зависимости производительности разных фракций от температуры, характеризующих самый нижний уровень иерархии. Для оптимизации верхнего уровня иерархии (КРІ) путем изменения управляющих воздействий нижнего уровня между ними требуется выявить четкую связь. Для этого выполнена идентификация модели зависимости выхода фракций 240-320 и 300-350 от температуры в отгонной секции РУ [8].

Таблица 1

Управляемые параметры и целевые показатели объекта

Table 1

Controlled object parameters and targets

Входные параметры	Управляемые параметры, U		Целевые показатели, G	
Сырая нефть	U ₁	Объем нефти на вводе в колонну	G ₁	Объем выхода фракции 140-240
	U ₂	Температура нефти на вводе в колонну	G ₂	Объем выхода фракции 240-300
	U ₃	Давление в колонне на участке ввода нефти	G ₃	Объем выхода фракции 300-350
	U ₄	Температура в отгонной секции	G ₄	Объем выхода мазута
	U ₅	Давление в отгонной секции	G ₅	Объем выхода верхнего продукта
	U ₆	Температура на тарелке 23		
	U ₇	Температура на тарелке 36		
	U ₈	Температура на тарелке 49		

Идентификация данных

Установка состоит из подогревателя сырья и теплообменника, ректификационной колонны, конденсатора-холодильника и кипятильника. Нагретое сырье вводится в зону питания ректификационной колонны, где разделяется на твердую и паровую фазы. В результате ректификации сверху колонны отбирается изопентан как верхний продукт, на тарелках между шлемом колонны и зоной питания конденсируются более тяжелые фракции.

Снизу, из отгонной секции, отбирается мазут как остаток (рис. 1) [9].

Для оптимизации был выбран показатель температуры (U4 в табл. 1) в отгонной секции колонны, изменение которого было призвано максимизировать выход фракций 240-300 и 300-350 (G2 и G3 в табл. 1).

Для идентификации представленных на рис. 2 зависимостей применена нейронная сеть (НС). НС обучалась на выборке из статистики колебаний параметров, полученной за сутки работы колонны-образца.

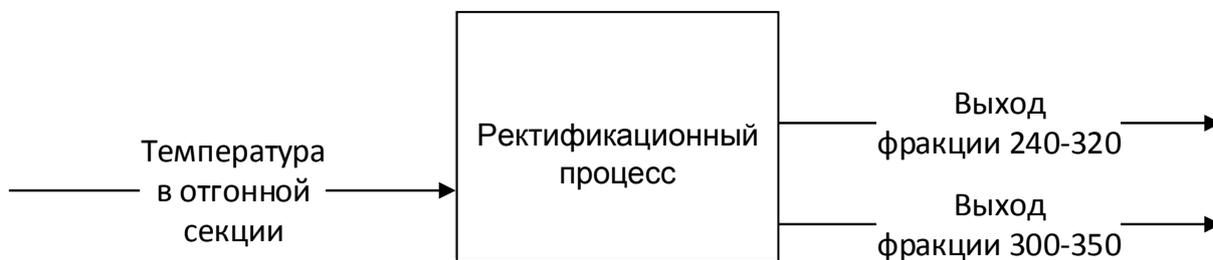


Рис. 1. Упрощенная схема ректификационного процесса

Fig. 1. Simplified scheme of the rectification process

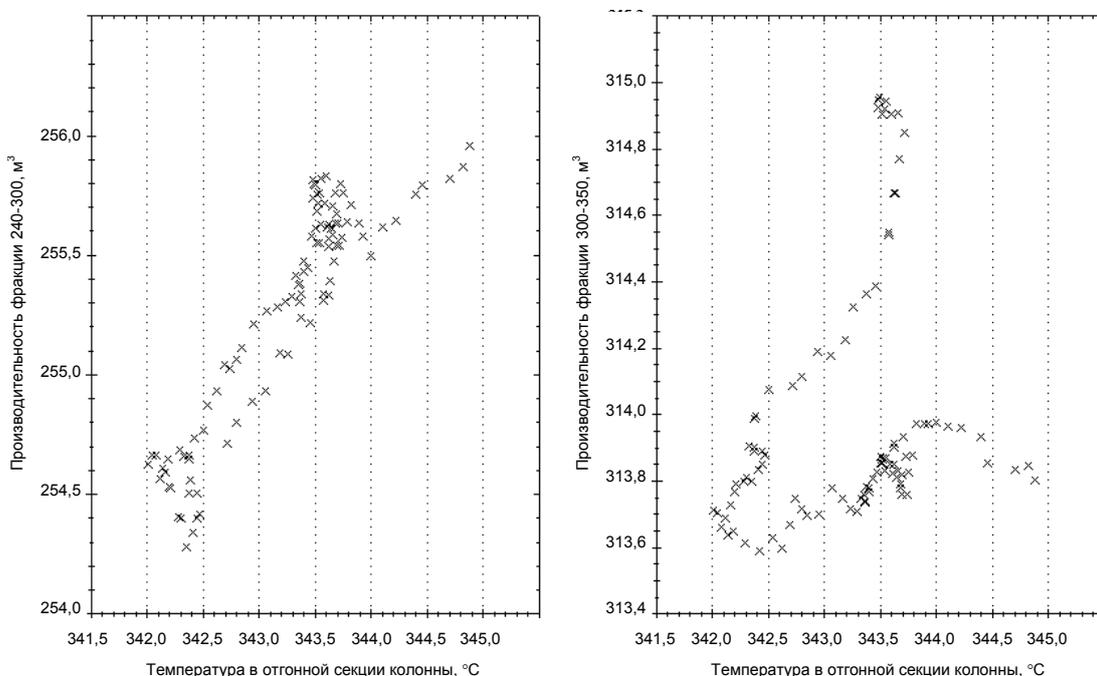


Рис. 2. Зависимость объема выхода фракции 240-300 (слева) и 300-350 (справа) от температуры

Fig. 2. The dependence of the output volume of the fraction 240-300 (left) and 300-350 (right) on temperature

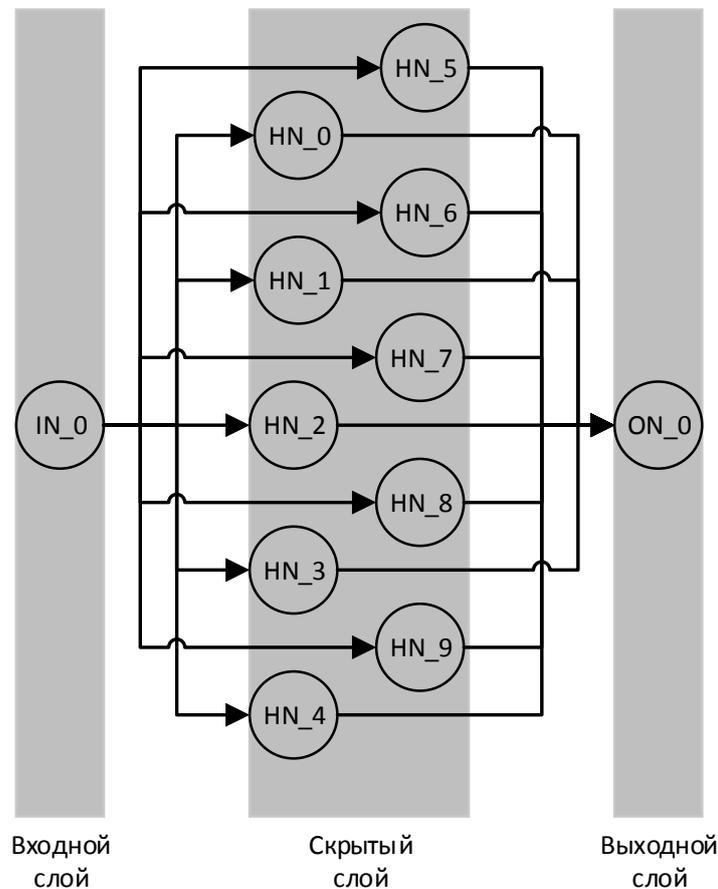


Рис. 3. Схема нейронной сети
 Fig. 3. Scheme of the neural network

Применена нейронная сеть, включающая один входной, один скрытый и один выходной слой (рис. 3). Входной и выходной слой содержат по одному нейрону, скрытый слой – 10. Обучение по принципу обратного распространения ошибки. Скрытый и выходной слой используют сигмоидную активационную функцию вида:

$$F(x) = \frac{e^x}{e^x + 1} \quad (1)$$

Таким образом, формула, описывающая работу нейронной сети:

$$ON = f\left(\sum_{k=0}^{10} f\left(\sum_{k=0}^{10} IN * IHN\right) * OHN\right), \quad (2)$$

где f – сигмоидная активационная формула (1), а IN, ON, IHN и OHN приведены в табл. 2 и 3, демонстрирующих конфигурации скрытого слоя для давления и температуры соответственно.

Применение описанной выше нейронной сети к исходным данным позволило получить модели связи температуры в отгонной секции с производительностью фракций 240-300 и 300-350, графическое представление которых зафиксировано на рис. 4.

Для проверки корреляции между полученными моделями (линии на рис. 4) и исходными наборами данных (точки на рисунке) были вычислены коэффициенты корреляции $\rho_{x,y}$, составившие 0,76312 для левого графика и 0,90781 для правого. Применена следующая формула:

$$\rho_{x,y} = \frac{\frac{1}{n} \sum_{t=1}^n X_t Y_t - \left(\frac{1}{n} \sum_{t=1}^n X_t\right) \left(\frac{1}{n} \sum_{t=1}^n Y_t\right)}{\sigma_x * \sigma_y}, \quad (3)$$

где σ_x и σ_y являются средними значениями соответствующих выборок.

Таблица 2

Скрытый нейронный слой, фракция 240-300

Table 2

The hidden neural layer, fraction 240-300

Нейрон	Вход (IN)	Выход (ON)	Вес до источника (IHN)	Вес до цели (OHN)
0	0,5329	0,4675	0,9646	0,5356
1	-0,4489	0,4218	-0,8125	-1,0897
2	0,6221	0,6083	1,1261	0,6041
3	-0,1149	0,5999	-0,2079	-0,7524
4	0,7458	0,5611	1,3499	0,8198
5	2,5314	0,7924	4,5817	3,1874
6	-1,0151	0,3031	-1,8377	-1,8294
7	1,5302	0,7177	2,7695	1,8181
8	-0,0096	0,4126	-0,0174	-0,3680
9	-0,3281	0,3505	-0,5939	-0,8191

Таблица 3

Скрытый нейронный слой, фракция 300-350

Table 3

The hidden neural layer, fraction 300-350

Нейрон	Вход (IN)	Выход (ON)	Вес до источника (IHN)	Вес до цели (OHN)
0	-0,4983	0,2103	-0,9457	0,7026
1	-3,5414	0,0843	-6,7410	-3,9722
2	-0,7764	0,3018	-1,4716	1,3219
3	0,7931	0,4216	1,5021	-1,416
4	-0,7696	0,3039	-1,4587	1,3393
5	-0,1638	0,1935	-0,3109	0,2529
6	-0,1228	0,2843	-0,2331	0,1390
7	-0,2815	0,2045	-0,5342	0,4098
8	-0,5891	0,2605	-1,1171	0,9753
9	0,0778	0,2562	0,1475	-0,1713

Показатели корреляции могли быть улучшены путем увеличения: обучающей выборки для нейронной сети; количества скрытых нейронных слоев и/или нейронов в них; числа итераций. Однако, чтобы не отклоняться от основной темы работы, данные показатели были признаны удовлетворительными.

Имея математические модели взаимосвязей базовых параметров перегонного процесса, стало возможным сравнить их

между собой и определить оптимальные точки согласно методу многокритериальной оптимизации.

Оптимизация по Парето

При решении задачи:

$$f_i(x) \rightarrow \min_{x \in D}, \quad i \in [1:k], \quad D \subset R^n, \quad (4)$$

где R_n – n -мерное евклидово пространство, нужно рассмотреть две точки x' и $x'' \in D$.

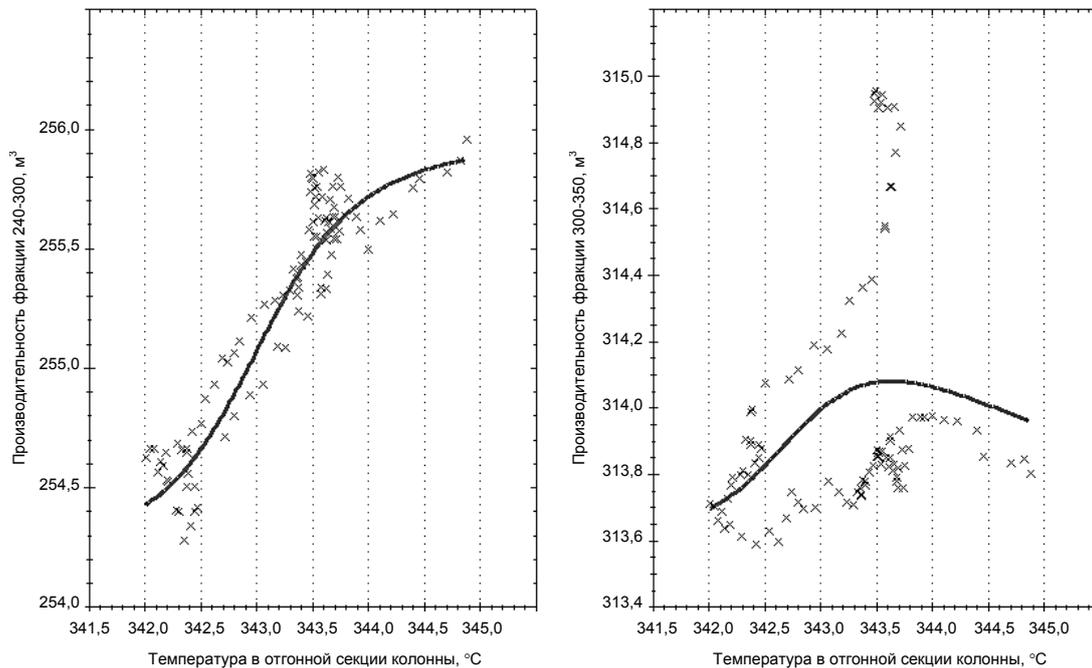


Рис. 4. Графическое представление идентифицированных моделей

Fig. 4. Graphic representation of the identified models

Если следующие неравенства выполняются для всех $i \in [1:k]$, причем по крайней мере одно из неравенств строгое, то будем говорить, что точка x' предпочтительнее чем x'' :

$$f_i(x') \leq f_i(x''). \quad (5)$$

Если для некоторой точки $x^0 \in D$ не существует более предпочтительных точек, то x^0 можно назвать эффективным или Парето-оптимальным решением многокритериальной задачи. Множество, включающее все эффективные решения, обозначается $P(D)$ и называется множеством Парето для векторного отображения $f: D \rightarrow R^n$, $f = (f_1, \dots, f_k)$, при этом $P(D) \subset D \subset R^n$. Образ множества $P(D)$ в пространстве критериев R^k обозначается как $P(f)$. Множество $P(f) = f(P(D))$ называется множеством эффективных оценок [10].

Смысл введенного понятия эффективного решения состоит в том, что оптимальное решение многокритериальной задачи следует искать только среди множества $P(D)$ (принцип Парето). В противном случае всегда найдется точка x , оказывающаяся

наиболее предпочтительной, независимо от расстановки приоритетов и относительной важности отдельных частных критериев [1].

Для оптимизации по Парето был выбран Strength Pareto Evolutionary Algorithm 2 (SPEA2) [11], обеспечивающий наиболее репрезентативное распределение точек по сравнению с другими алгоритмами (VEGA, FFGA [12], NPGA [13], SPEA).

Как и перечисленные выше алгоритмы, SPEA2 оперирует понятием «силы», рассчитываемым как количество точек, которые доминируются по Парето данной точкой. Однако высокий показатель силы не гарантирует, что точка располагается близко к границе Парето, поэтому здесь вводится основанная на силе величина – слабость. Слабость точки есть суммарная сила всех точек, доминирующих над данной. Для точки i в группе G формула слабости выражается как

$$\text{Слабость}(i) = \sum_{g \in G} \text{Сила}(g). \quad (6)$$

Для выбора оптимальной точки желательно, чтобы хилость была как можно

меньше: недоминируемая особь имеет слабость, равную нулю. Для выполнения расчетов по алгоритму SPEA2 применяется величина, обратная слабости, которую можно обозначить как приспособленность:

$$\text{Приспособленность}(i) = \frac{1}{1 + \text{Слабость}(i)}. \quad (7)$$

Кроме того, SPEA2 реализует возможность доукомплектования внешнего множества лучшими доминирующими индивидами из текущей популяции, если мощность внешнего множества меньше допустимого значения, а кластеризация обеспечивает более репрезентативное распределение недоминируемых решений [14].

В сокращенном виде алгоритм SPEA2 можно представить следующим образом:

Шаг 1. Инициализация: создать первоначальную популяцию P_0 и пустой архив (отдельное множество) $\bar{P}_0 = \emptyset$. Установить $t = 0$.

Шаг 2. Назначение параметра пригодности: рассчитать значения пригодности для элементов множеств P_t и \bar{P}_t .

Шаг 3. Селекция на основе окружения: скопировать всех недоминируемых индивидов из P_t и \bar{P}_t в P_{t+1} . Если размер P_{t+1} превышает размер \bar{N} , P_{t+1} должен быть уменьшен; в обратном случае размер P_{t+1} наращивается за счет доминируемых индивидов из P_t и \bar{P}_t .

Шаг 4. Остановка: если $t \geq T$ (достигнут лимит поколений) или выполнен другой критерий остановки алгоритма, выполнение прекращается, а набором искомым индивидов считается P_{t+1} .

Шаг 5. Селекция на основе спаривания: провести среди элементов P_{t+1} бинарный турнир с заменой, чтобы заполнить пул спаривания.

Шаг 6. Мутация: применить операторы мутации и рекомбинации к пулу спаривания, сохранив полученную популяцию в P_{t+1} . Увеличить счетчик поколений на 1 ($t = t + 1$) и перейти к шагу 2.

Порядок выполнения шагов алгоритма представлен на рис. 5.

Алгоритм SPEA2 применен для поиска множества значений температуры в колонне, обеспечивающих максимизацию выхода фракций 240-300 и 300-350 (рис. 6). Для этого использовались полученные ранее идентифицированные модели. Сохраняя прежний размер выборки, мы соотнесли графики множества Парето и моделей, чтобы получить числовые значения производительности. Полный перечень точек и соответствующих им значений приведен в табл. 4.

Результатом применения алгоритма к идентифицированным моделям стало построение множества Парето из восьми точек, отсортированных по убыванию оптимальности.

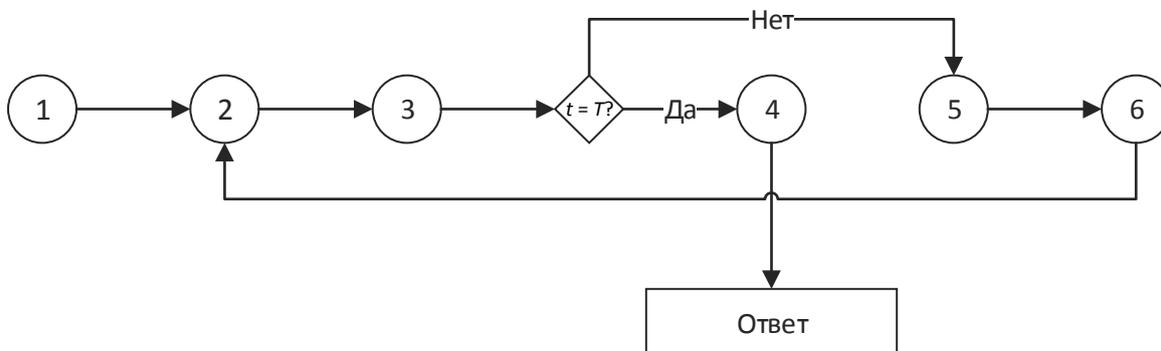


Рис. 5. Порядок выполнения шагов алгоритма SPEA2
 Fig. 5. The order of execution of the steps of the algorithm SPEA2

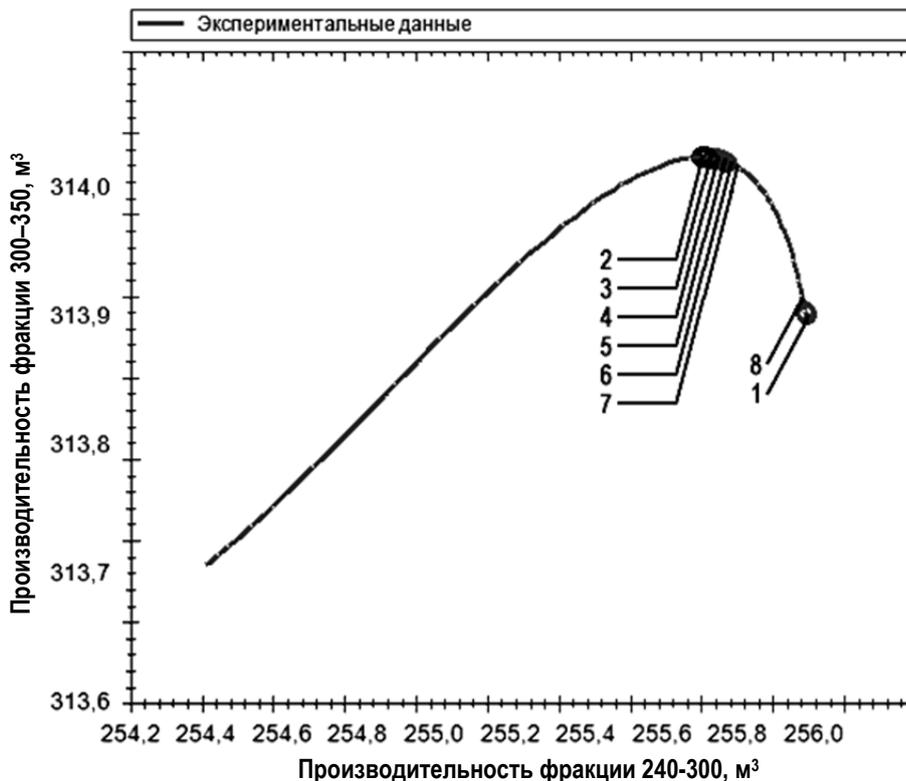


Рис. 6. Графическое представление множества Парето
 Fig. 6. Graphical representation of the Pareto set

Таблица 4

Числовые значения множества Парето

Table 4

The numerical values of the Pareto set

Точка	Температура, °С	Выход фракции 240-300, м³	Выход фракции 300-350, м³
1	344,823852539063	255,869043673089	313,914703058703
2	343,325988769531	255,623350989602	314,04678075533
3	343,371765136719	255,636549610692	314,046650398148
4	344,712951660156	255,866950335304	313,918779855307
5	343,374877929688	255,661217222795	314,045467958112
6	343,35595703125	255,672724869121	314,044445521759
7	343,364624023438	255,649165327318	314,046207859264
8	343,407836914063	255,683708140522	314,043155299305

Степень оптимальности точек определяется максимизацией выхода фракций и повышением рабочей температуры внутри РУ.

Выводы

В ходе работы выполнена декомпозиция ректификационной установки, определены

входные, управляемые и целевые показатели. На основе собранной статистики при помощи нейронной сети идентифицированы модели зависимостей между параметрами. Модели использованы для определения множества Парето из восьми точек, выбранных лучшими по соотношению значений.

В дальнейшем планируется расширить сферу практического применения рассмотренного выше подхода. Для этого потребуется составить полные цепочки связей от базовых управляющих параметров до КРІ верхнего уровня, что поможет реализовать отслеживание ключевых индикаторов работы установки или всего предприятия в

реальном времени. Доработав управляющее программное обеспечение, можно дать обратный ход процессу, подбирая управляющие процессы, обеспечивающие заранее заданные значения КРІ. Это позволит не только оптимизировать сложные процессы, но и обеспечить более эффективный контроль над ними.

СПИСОК ЛИТЕРАТУРЫ

1. Черноруцкий И.Г. Методы оптимизации. Компьютерные технологии. СПб.: БХВ-Петербург, 2011. 384 с.
2. Буренина И.В., Варакина В.А. Система единых показателей оценки эффективности деятельности вертикально-интегрированных нефтяных компаний // *Науковедение*. 2014. № 1(20). С. 6.
3. Машунин К.Ю., Машунин Ю.К. Моделирование технических систем в условиях неопределенности и принятие оптимального решения // *Известия РАН. Теория и системы управления*. 2013. № 4. С. 19–35.
4. Kostenko D., Kudryashov N., Maystrishin M., Onufriev V., Potekhin V., Vasiliev A. Digital twin applications: Diagnostics, optimisation and prediction // *Proc. of the 29th DAAAM Internat. Symp. DAAAM Internat.*, 2018. Pp. 574–581.
5. Yaochu J. Pareto-optimality is everywhere: From engineering design, machine learning, to biological systems // 2008 3rd Internat. Workshop on Genetic and Evolving Systems. IEEE. 2008. DOI: 10.1109/GEFS.2008.4484555
6. Schleich B., Anwer N., Mathieu L., Wartzack S. Shaping the digital twin for design and production engineering // *CIRP Annals*. 2017. Vol. 1. № 66. Pp. 141–144.
7. Бродецкий Г.Л. Методы оптимизации многокритериальных решений в логистике. М.: 2009. 224 с.
8. Кини Р.Л., Райфа Х. Принятие решений при многих критериях: предпочтения и замещения. М.: Радио и связь, 1981. 560 с.
9. Гуревич И.Л. Общие свойства и первичные методы переработки нефти и газа. М.: Химия, 1972. 361 с.
10. Подиновский В.В., Ногин В.Д. Парето-оптимальные решения многокритериальных задач. М.: Наука, 1982. 256 с.
11. Zitzler E., Laumanns M., Thiele L. SPEA2: Improving the strength Pareto evolutionary algorithm // *Techn. Rep. TIK Report*. Zurich: Department of Electrical Engineering Swiss Federal Institute of Technology (ETH), 2001. 21 p.
12. Fonseca C.M., Fleming P.J. Genetic algorithm for multiobjective optimization, formulation, discussion and generalization // *Genetic Algorithms: Proc. of the 5th Internat. Conf. San Francisco CA: Morgan Kaufmann Publishers Inc.*, 1993. Vol. 1. Pp. 416–423.
13. Horn J.N., Nafpliotis A.L., Goldberg D.E. A niched Pareto genetic algorithm for multiobjective optimization // *Proc. of the 1st IEEE Conf. on Evolutionary Computation*. IEEE World Congress on Computational Intelligence. Piscataway NJ: IEEE Service Center, 1994. Vol. 1. Pp. 82–87.
14. Тарутин А.В., Набатов А.В. Применение методов генетических алгоритмов для построения множества Парето в задачах многокритериальной оптимизации // *Инженерный вестник Дона*. 2015. № 4. С. 39.

Статья поступила в редакцию 30.12.2018.

REFERENCES

1. Chemoruckiy I.G. *Metody optimizacii. Kompyuternye tekhnologii* [Optimisation methods. Computer technology]. St. Petersburg: BHV-Petersburg, 2011. 384 p. (rus)
2. Burenina I.V., Varakina V.A. Sistema yedinykh pokazateley otsenki effektivnosti deyatel'nosti vertikalno-integrirovannykh neftyanykh kompaniy [The single indicator system of performance assessment of vertically integrated oil enterprises]. *Naukovedeniye* [Science studies], 2014, No. 1(20). Pp. 6. (rus)
3. Mashunin K.Yu., Mashunin Yu.K. Modelirovaniye tekhnicheskikh sistem v usloviyakh neopredelennosti i prinyatiye optimalnogo resheniya [Modelling of technical systems under uncertain circumstances and making an optimal decision]. *Izvestiya RAN. Teoriya i sistemy upravleniya* [RAN News. Theory of control systems], 2013, No. 4, Pp. 19–35. (rus)
4. Kostenko D., Kudryashov N., Maystrishin M., Onufriev V., Potekhin V., Vasiliev A. Digital twin

applications: diagnostics, optimisation and prediction. *Proceedings of the 29th DAAAM International Symposium*. DAAAM International, 2018, Pp. 574–581.

5. **Yaochu J.** Pareto-optimality is everywhere: From engineering design, machine learning, to biological systems. *2008 3rd International Workshop on Genetic and Evolving Systems*. IEEE, 2008. DOI: 10.1109/GEFS.2008.4484555

6. **Sleich B., Anwer N., Mathieu L., Wartzack S.** Shaping the digital twin for design and production engineering. *CIRP Annals*, 2017, Vol. 1, No. 66, Pp. 141–144.

7. **Brodetskiy G.L.** *Metody optimizatsii mnogokriterialnykh resheniy v logistike [Methods of multicriteria decisions optimization in logistics]*. Moscow: 2009, 224 p. (rus)

8. **Keeney R.L., Raifa H.** *Prinyatiye resheniy pri mnogikh kriteriyakh: predpochteniya i zameshcheniya [Decisions with Multiple Objectives: Preferences and Value Tradeoffs]*. Moscow: Radio i svyaz Publ., 1981, 560 p. (rus)

9. **Gurevich I.L.** *Obshchiye svoystva i pervichnyye metody pererabotki nefii i gaza [Common properties and primary methods of oil and gas processing]*. Moscow: Khimiya, 1972, 361 p. (rus)

10. **Podinovskiy V.V., Nogin V.D.** *Pareto-optimalnye resheniya mnogokriterialnykh zadach*

[*Pareto-optimal solutions for multicriterial tasks*]. Moscow: Nauka Publ., 1982, 256 p. (rus)

11. **Zitzler E., Laumanns M., Thiele L.** SPEA2: Improving the strength Pareto evolutionary algorithm. *Techn. Rep. TIK Report*. Zurich: Department of Electrical Engineering Swiss Federal Institute of Technology (ETH), 2001, 21 p.

12. **Fonseca C.M., Fleming P.J.** Genetic algorithm for multiobjective optimization, formulation, discussion and generalization. *Genetic Algorithms: Proceeding of the 5th International Conference*. San Francisco CA: Morgan Kaufmann Publishers Inc., 1993, Vol. 1, Pp. 416–423.

13. **Horn J.N., Nafpliotis A.L., Goldberg D.E.** A niched Pareto genetic algorithm for multiobjective optimization. *Proceedings of the 1st IEEE Conference on Evolutionary Computation. IEEE World Congress on Computational Intelligence*. Piscataway NJ: IEEE Service Center, 1994, Vol. 1, Pp. 82–87.

14. **Tarutin A.V., Nabatov A.V.** *Primeneniye metodov geneticheskikh algoritmov dlya postroyeniya mnozhestva Pareto v zadachakh mnogokriterialnoy optimizatsii [Application of genetical algorithms to build a Pareto set in a multi criteria optimization task]*. *Inzhenernyy vestnik Dona [Engineering news of Don]*, 2015, No. 4, P. 39. (rus)

Received 30.12.2018.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

КОСТЕНКО Дмитрий Андреевич

KOSTENKO Dmitri A.

E-mail: zaba-1@bk.ru

ОНУФРИЕВ Вадим Александрович

ONUFRIEV Vadim A.

E-mail: ovavadim@gmail.com

ШКОДЫРЕВ Вячеслав Петрович

SHKODYREV Viacheslav P.

E-mail: shkodyrev@mail.ru

DOI: 10.18721/JCSTCS.12205

УДК 004.652.42, 004.657, 004.655.3:004.652.4

ИССЛЕДОВАНИЕ АЛГОРИТМОВ ДЕКОМПОЗИЦИИ И ВЫПОЛНЕНИЯ ЗАПРОСОВ НА ВЫБОРКУ В ГЕТЕРОГЕННЫХ СИСТЕМАХ УПРАВЛЕНИЯ РЕЛЯЦИОННЫМИ БАЗАМИ ДАННЫХ

С.Г. Попов, А.А. Пурий

Санкт-Петербургский политехнический университет Петра Великого,
Санкт-Петербург, Российская Федерация

При работе с независимыми базами данных появляется проблема кусочной автоматизации, состоящая в невозможности получать данные из независимых баз с заранее неизвестной схемой. У проблемы два аспекта: отсутствие семантической связи между независимыми базами и динамика семантики схем. Разработаны и исследованы алгоритмы построения интегрированных запросов на выборку данных к гетерогенным распределенным системам управления базами данных. Вычислены алгоритмы декомпозиции и выполнения запроса на выборку. Для проверки работоспособности алгоритмов разработано программное обеспечение и база данных, обеспечивающие обращение к независимым базам, как если бы они находились в единой СУБД, но без полной репликации объединяемых баз данных.

Ключевые слова: гетерогенные СУБД, метаданные, распределенные запросы, декомпозиция запросов, распределенные СУБД, интеграция баз данных, обработка запросов, СУБД, базы данных.

Ссылка при цитировании: Попов С.Г., Пурий А.А. Исследование алгоритмов декомпозиции и выполнения запросов на выборку в гетерогенных системах управления реляционными базами данных // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2019. Т. 12. № 2. С. 50–67. DOI: 10.18721/JCSTCS.12205

STUDY DECOMPOSING AND EXECUTING ALGORITHMS OF DATA EXTRACTION QUERIES IN HETEROGENEOUS RELATIONSHIP DATABASE MANAGEMENT SYSTEMS

S.G. Popov, A.A. Purii

Peter the Great St. Petersburg Polytechnic University,
St. Petersburg, Russian Federation

When working with independent databases, the problem of partial automation arises, which consists in the impossibility of obtaining data from independent databases with a previously unknown scheme. The problem consists of two aspects: the lack of semantic connection between independent databases, and the dynamics of the semantics of schemes. The goal is the development and study of algorithms

for constructing integrated queries for data sampling to heterogeneous distributed database management systems. To test the functionality of the algorithms, software and database of databases, which provides the ability to access independent databases, as if they were a single DBMS, but without full replication of the merging databases.

Keywords: heterogeneous DBMS, metadata, distributed queries, queries decomposition, distributed DBMS, databases integration, queries processing, DBMS, databases.

Citation: Popov S.G., Purii A.A. Study decomposing and executing algorithms of data extraction queries in heterogeneous relationship database management systems. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2019, Vol. 12, No. 2, Pp. 50–67. DOI: 10.18721/JCSTCS.12205

Проблема интеграции гетерогенных систем управления базами данных

Независимые базы данных содержат данные об одних и тех же объектах реального мира, но для получения полного набора данных об одном и том же объекте необходимо последовательно извлекать данные из этих баз, а затем осуществлять их дополнительную обработку в специализированном программном обеспечении. Эта проблема вызвана кусочной автоматизацией, состоящей во фрагментации частей единой информационной среды, и, как следствие, невозможности получать данные из независимых баз с заранее известной схемой.

Проблема состоит из двух аспектов. Первый: независимые базы не связаны семантически, соединение ранее независимых баз требует дополнительных усилий администратора баз данных, получив при этом семантически корректный результат [11, 12]. Второй: динамика семантики схем, состоящая в том, что набор и схемы баз данных для выполнения к ним запросов постоянно меняется. Проблема отсутствия семантических связей разрешается с помощью автоматизированных предметно-ориентированных алгоритмов [13, 15, 16], в том числе с использованием метаданных [14, 17, 20].

Второй аспект проблемы может быть решен с помощью нескольких подходов, состоящих в учете динамики семантической схемы. На выбор решения влияют

два фактора: динамика семантической схемы [18] и динамика семантики запросов [19]. Современные СУБД рассчитаны на решение задач с большим числом запросов и низкой динамикой схемы данных. В этом случае все базы можно объединить в одной СУБД. Такой выбор не подходит для решения задач с высокой динамикой схем и любой интенсивностью запросов, поскольку при высокой динамике придется включать схемы, которые используются, например, единожды. Кроме того, такое решение требует согласия владельцев на репликацию их баз в единую СУБД. Построение запросов к таким системам может быть решено при помощи класса систем интеграции гетерогенных СУБД.

В этом случае под системой интеграции распределенных гетерогенных СУБД подразумевается системное программное обеспечение и база данных баз данных, которая предоставляет возможность обращаться к независимым базам, как если бы они находились в единой СУБД, но без полной репликации объединяемых баз данных.

Описание методов и технологий реализации распределенных запросов

Построение оптимизированных планов выполнения запросов являлось одной из важнейших задач для систем управления базами данных. Исследования в этой области, проводимые в последние годы, не

только решают задачи построения запросов с использованием новых методов, таких как глубокое обучение с подкреплением, но и задачи построения запросов в более сложных, распределенных системах.

Для оптимизации запросов с помощью точных оценок может использоваться метод оптимизации запросов с помощью выполнения уточняющего подзапроса [1]. Результат такого подзапроса позволяет оценить количество данных, участвующих в запросе, и на этом основании построить оптимальный запрос. Подобный подход позволяет избегать накопления ошибок, возникающих из-за устаревших статистических данных о количестве записей в базе, при построении плана, и основывается только на точных оценках. Данный подход разработан для баз данных, расположенных в ОЗУ, а обработка запросов производилась массивно параллельно на GPU. Негативный аспект подобного подхода в неэффективной работе с базами данных, расположенными в постоянной памяти.

Альтернативным подходом для построения запросов с помощью точных оценок может быть реоптимизация динамических запросов с использованием временного планирования [7]. Временное планирование заключается в выполнении набора заранее определенных запросов для сбора метрик о состоянии системы, например, загрузка CPU, количество свободной памяти, скорость передачи данных по сети. Если состояние системы изменилось таким образом, что текущий план выполнения не является оптимальным, то происходит реоптимизация, и план запроса перестраивается. Подобный подход позволяет создавать более устойчивые к изменениям окружения динамические планы выполнения запросов.

Другим способом оптимизации запросов является использование методов глубокого обучения с подкреплением [2, 3]. Оптимизация заключается не в полной оптимизации запроса, а в составлении оптимального порядка соединений таблиц, передающихся

в оптимизатор запроса. Вначале строится дерево соединений, и на нём осуществляется итеративно выбор порядка соединений. На каждой итерации две таблицы или поддерева объединяются в поддерево или результирующее дерево соединений. Нейронная сеть при обучении пытается достичь результата стоимости и времени выполнения запроса меньше, чем результат стандартного оптимизатора соединений в базе данных. Таким образом, она должна обучаться отдельно для каждой реализации системы управления базы данных.

Для обработки запросов в распределенной базе данных в среде беспроводных сенсоров [4] могут использоваться распределенный и централизованный методы оптимизации и обработки [9]. Централизованный алгоритм для проведения операции соединения собирает данные всех таблиц, входящих в запрос, на центральный сервер, на котором и происходит обработка, что повышает нагрузку на сеть. Алгоритм распределенной циклически-вложенной обработки соединений строит дерево, узлами которого являются регионы, вычисляющие запросы соединения на подчиненных им таблицах или регионах. В результате вычислений в корне дерева будет содержаться ответ на запрос. Исследование показало, что время выполнения запроса распределенным алгоритмом было ниже, чем время выполнения централизованным алгоритмом.

Для оптимизации запросов к распределенным базам данных можно использовать улучшенный генетический алгоритм. Оптимизация выполняется только для JOIN. Соединения в виде хромосом задаются закодированными деревьями – иерархией соединений таблиц. Фитнесс-функция – время, которое необходимо затратить на соединение, с учетом затрат для передачи данных. При осуществлении кроссовера используются различные вероятности мутации для разных рангов: чем выше ранг, тем ниже вероятность. Популяция кластеризуется на ранги алгоритмом нечёткой кластеризации. При мутации происходит взаим-

ное перемещение двух таблиц в иерархии, вероятность мутации также зависит от ранга. Критериями завершения алгоритма являются достижение заданного числа эпох или достижение заданной точности.

Обработка интегрированных запросов к гетерогенным базам данных реализована в промышленных программных продуктах, представителями которых являются Apache Calcite и Presto. Пакет Calcite – фреймворк с открытым исходным кодом для оптимизированной обработки запросов для гетерогенных источников данных, – позволяет использовать кроссплатформенную оптимизацию между несколькими системами управления данными, путем использования стандартного интерфейса [8]. Также Calcite требует ручного построения плагинов и проводников для извлечения метаданных и правил для оптимизации запросов. Calcite имеет возможность обрабатывать только не интегрированные SQL-запросы.

Аналогом Calcite является движок с открытым исходным кодом для обработки распределенных SQL-запросов Presto [10]. Presto имеет встроенный оптимизатор, использующий статистическую информацию о таблицах в удаленных базах данных. Основное направление оптимизации – упорядочивание JOIN и распределение их выполнения между узлами. Для упорядочивания JOIN Presto имеет три режима: автоматический, удаление избыточных декартовых произведений (CROSS JOIN) и без оптимизации. В автоматическом режиме оптимизатор упорядочивает JOIN на основе статистических данных, если статистические данные получить невозможно, то оптимизация производится в режиме удаления избыточных декартовых соединений. Исполнение JOIN может происходить в двух режимах: широковещательном и раздельном. В широковещательном режиме каждый узел, участвующий в запросе, использует все данные, они реплицируются на каждый узел. В раздельном режиме каждый узел использует только часть данных. Оптимизатор может выбирать подходящие режимы исполнения автоматически или использо-

вать какой-то определенный в зависимости от настройки. Важной функцией является механизм анализа плана запроса. Он показывает, какие действия будут проведены при выполнении запроса, оценочные показатели количества обрабатываемых строк, количества использования CPU, памяти и сетевого трафика. Такой анализ плана помогает при отладке запросов.

Постановка задачи

Наша задача – разработка и исследование алгоритмов декомпозиции и выполнения интегрированных запросов на выборку данных к гетерогенным СУБД. Для проверки работоспособности и эффективности разрабатываемых алгоритмов предложим архитектуру системы выполнения запросов к гетерогенным СУБД. На основе архитектуры реализуем макет программного обеспечения промежуточного слоя, обеспечивающий хранение метаданных объединяемых баз данных и реализующий интерпретатор запросов с возможностью декомпозиции и выполнения запросов. Исследуем предложенные алгоритмы, определив границы их применимости.

Алгоритмы выполнения распределенных запросов

Технология централизованного выполнения запросов. При выполнении запросов на выборку данных к гетерогенным СУБД наиболее важной частью является гомогенизация источников, поскольку запросы, быстро выполняющиеся на одном типе гетерогенной СУБД, могут выполняться медленно на другом. Поэтому возникает потребность в переносе данных из гетерогенных источников данных в некоторый источник определенного типа, чтобы избавиться от неоднозначности при построении запросов, а также при получении оценок во время построения этих запросов. Один из самых действенных способов получения данных из гетерогенных СУБД – выполнение к каждой СУБД запросов на выборку данных, при этом минимизировать объем передаваемых данных, выполняя

выборку по определенному условию и с проекцией только на используемые в основном запросе атрибуты. Пример декомпозиции исходного запроса на подзапросы к гетерогенным СУБД приведен на рис. 1. Результаты таких запросов необходимо поместить в некоторую локальную СУБД, к которой имеется полный доступ, и особенности ее функционирования известны. И только после этих операций можно производить соединение локализованных таблиц.

Алгоритмы декомпозиции запросов обеспечивают взаимодействие между различными СУБД, расположенными на разных удаленных серверах. Для осуществления взаимодействия применяется централизованный подход к обработке запросов. При обработке исходного запроса для построения подзапросов, а также динамической проверки корректности оригинального запроса, используются метаданные удаленных СУБД. Для их получения применя-

ется локальная база данных метаданных, в которой хранятся метаданные всех серверов, входящих в систему.

Выполнение запроса осуществляется в три этапа и реализуется четырьмя алгоритмами: алгоритмом проверки корректности запроса, двумя алгоритмами декомпозиции и одним алгоритмом выполнения плана. Алгоритм проверки корректности запроса обеспечивает динамическую проверку синтаксиса запроса к нескольким СУБД. Алгоритмы декомпозиции используют левый вывод для разделения исходного запроса на подзапросы к удаленным СУБД. Алгоритм выполнения плана объединяет промежуточные результаты в окончательный результат запроса. Для получения данных из удаленных СУБД алгоритм выполнения плана использует подход создания локальных копий во временной базе данных на локальном сервере. Схема взаимодействия алгоритмов в ходе централизованного выполнения запроса приведена на рис. 2.

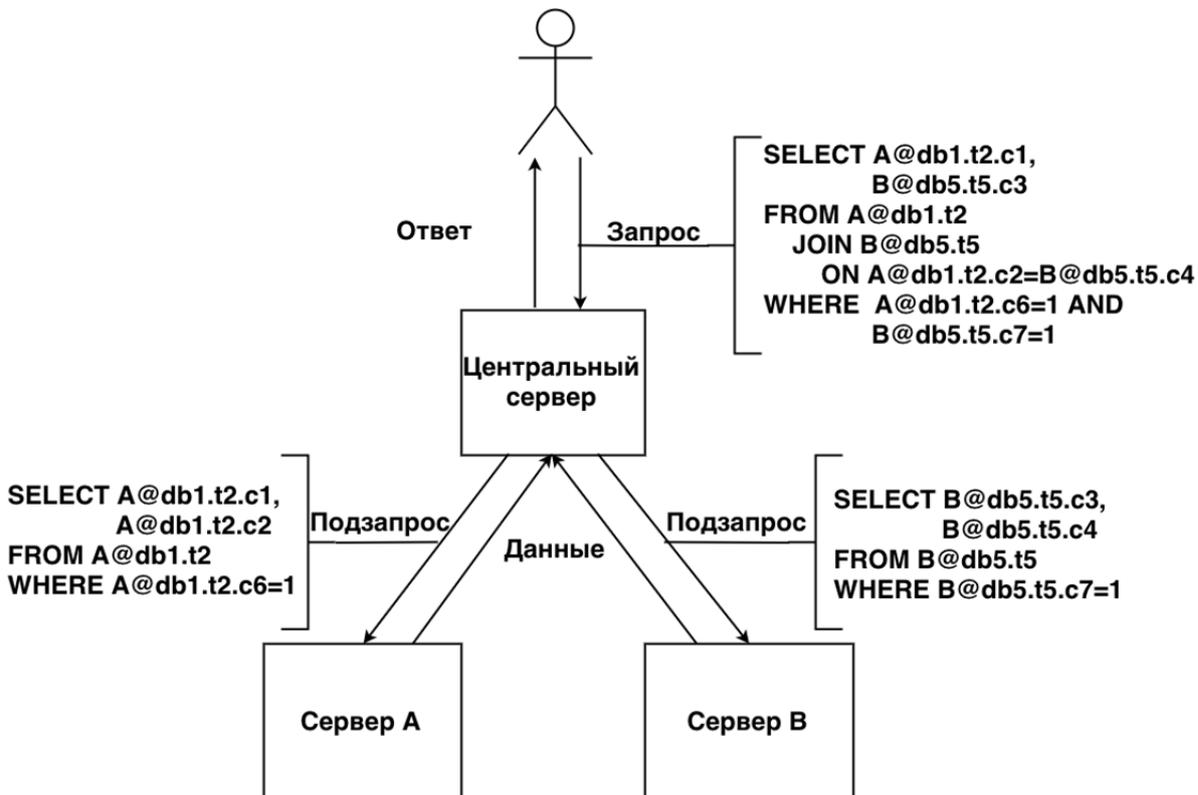


Рис. 1. Пример декомпозиции запроса

Fig. 1. Query decomposition example

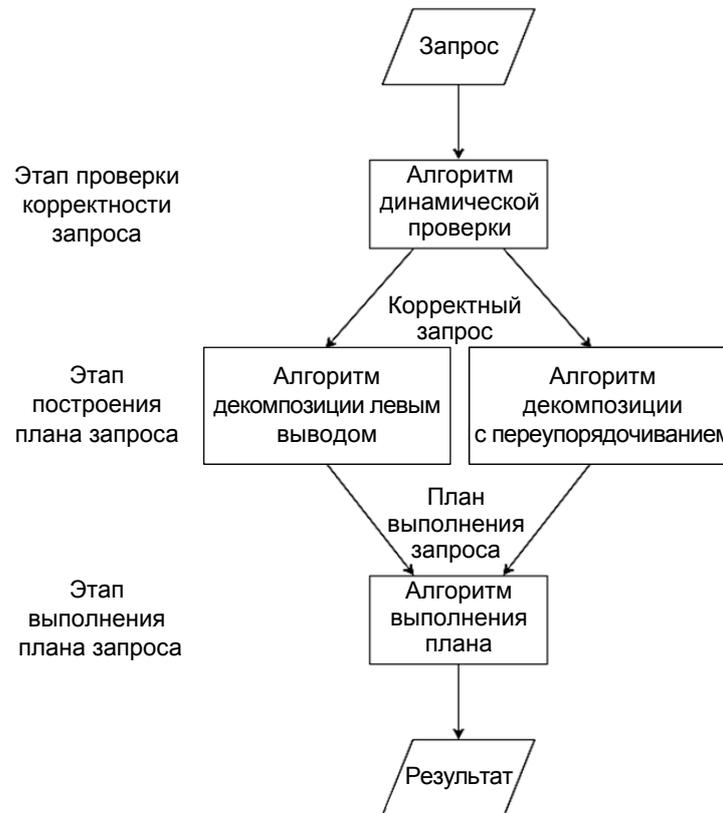


Рис. 2. Взаимодействие алгоритмов централизованного метода выполнения запросов

Fig. 2. Centralized query execution method algorithms interaction

Метод выполнения запросов на выборку заключается в использовании одного сервера или центрального сервера в качестве коммутатора для всех серверов СУБД, входящих в систему. Основной задачей данного коммутатора является управление потоками данных из удаленных СУБД и их локальное соединение. План действий с промежуточными результатами, исполняемый алгоритмом выполнения плана, строится при обработке исходного запроса к системе алгоритмами декомпозиции, а результатом выполнения такого плана должен быть результат выполнения исходного запроса.

Этап проверки корректности запроса. На данном этапе выполняется алгоритм динамической проверки корректности. В его задачи входит проверить: все ли имена серверов, баз данных, таблиц и атрибутов, входящие в запрос, существуют; верно ли расставлены все идентификаторы

в запросе, например, все ли идентификаторы в «from» и «join» частях запроса являются таблицами, а все идентификаторы в «on» части «join» являются атрибутами. Для таких проверок необходимо знать только имена соответствующих частей идентификаторов.

Алгоритм динамической проверки относится к классу линейной сложности, его оценка — $O(n)$, где n — число идентификаторов в запросе.

Этап построения плана запроса. На этом этапе выполняется выбор последовательности локализации и соединений таблиц. Соединения выполняются двумя алгоритмами: на основе стратегий левого вывода и переупорядочиванием.

Алгоритм декомпозиции левым выводом составляет план выборки и соединения таблиц согласно их порядку во «from» части запроса. Согласно технологии централизованной обработки, данные внеш-

них серверов должны быть локализованы, только после этого можно производить их соединение. В случае с левым выводом алгоритм обрабатывает таблицы попарно. Сначала берутся первые две таблицы, локализуются, затем соединяются. Поскольку результат их соединения находится на центральном сервере и является левой таблицей в следующем соединении, то при последующих соединениях необходимо локализовать только правую таблицу. Для снижения объемов передаваемых данных локализуется не вся удаленная таблица, а только ее проекция с используемыми в запросе атрибутами.

Алгоритмическая сложность декомпозиции запроса левым выводом линейная — $O(n)$, где n — число таблиц в части запроса «from». Учитывая, что число таблиц в части «from» запроса редко превышает 10, использование такого алгоритма для построения планов выполнения запросов является целесообразным.

Алгоритм с переупорядочиванием дополняет алгоритм декомпозиции левым выводом переупорядочиванием соединений в «from» перед началом построения дерева соединений. Для переупорядочивания производится перебор допустимых вариантов порядка соединений. Каждое из них статически оценивается числом строк в результирующей таблице после соединения всех таблиц в указанном порядке.

Для расчета оценки используются кардинальные числа таблиц и коэффициент уникальности атрибута. Оценки объёмов данных в процессе выполнения соедине-

ний приведены в табл. 1. В таблице $u(t)$ обозначен коэффициент уникальности для атрибута в таблице t , по которому происходит соединение. Для соединения типа «декартово произведение» верхняя оценка является точной и равна $T_1 * T_2$.

Наилучшим считается то соединение, число строк в результирующей таблице которого минимально. При равной оценке у нескольких порядков соединений приоритет отдается порядку, наиболее эффективно выполняемому используемыми программными средствами реализации.

Сложность алгоритма декомпозиции равна сумме сложностей для рекуррентной процедуры перебора и оценки возможных соединений, и сложности алгоритма декомпозиции левым выводом. Сложность алгоритма декомпозиции левым выводом линейна и зависит от числа таблиц во «from» части запроса. Поскольку перебор осуществляется путем обхода дерева возможных соединений в глубину, то его сложность будет равна $O(E)$, где E — число ребер в дереве, оно зависит от типа соединений таблиц и числа таблиц в запросе. Таким образом, сложность алгоритма с перестановками равна $O(E + n)$.

Этап выполнения запроса. На данном этапе реализуется алгоритм выполнения плана. Алгоритм выполняет пошаговое выполнение плана, составленного алгоритмами декомпозиции. Алгоритм обеспечивает ленивое выполнение плана путем прекращения выполнения запроса, если какая-либо из локализуемых таблиц или результат соединения пустой.

Таблица 1

Статические оценки для соединения двух таблиц

Table 1

Statistic estimates for two tables join

$T_1 \backslash T_2$	$u(T_2) = 1$	$u(T_2) < 1$
$u(T_1) = 1$	$\min(T_1, T_2)$	$\max(T_1, T_2)$
$u(T_1) < 1$	$\max(T_1, T_2)$	$(1 - u(T_1)) * T_1 * (1 - u(T_2))T_2$

Алгоритм выполнения плана реализует две процедуры: локализации таблиц и алгоритм соединения таблиц. Процедуры могут работать независимо, но для обеспечения ленивого выполнения порядок их вызова строго описан в плане. Сложность алгоритма выполнения плана линейна и зависит только от числа пунктов в плане — $O(n)$, где n — число пунктов в плане.

Реализация прототипа системы управления выполнением запроса к гетерогенным СУБД

Проверка эффективности алгоритмов выполняется в прототипе системы для обработки запросов к гетерогенным СУБД. Система включает в себя несколько модулей, отвечающих за работу определенной подсистемы. Одним самых важных модулей является модуль оптимизации. Этот модуль содержит реализацию алгоритмов декомпозиции. Помимо внутреннего взаимодействия система обменивается данными с различными СУБД. Для возможности функционирования системы необходимо наличие двух технологических баз: базы данных метаданных и временной базы данных.

База данных метаданных содержит метаданные всех СУБД, включенных в систему. Метаданные используются на всех этапах обработки и выполнения запроса. Для сохранения промежуточных результатов при выполнении запроса используется временная база данных. Временные базы создаются по одной на выполняемый запрос и удаляются при завершении выполнения запроса.

Система управления запросами реализована на основе модульного принципа модулей. Структурная схема модулей изображена на рис. 3.

На рис. 4 изображена функциональная схема, демонстрирующая, в каких конфигурациях среды возможно использование системы. Поскольку система взаимодействует со всеми СУБД через глобальную или локальную сеть, то возможно два типа размещения базы данных метаданных и временной базы данных. Первый — использование баз, размещенных в глобальной сети Ин-

тернет. Преимущество такого подхода в том, что он позволяет сразу множеству копий системы использовать базу метаданных как общий ресурс. Очевидный минус — в том, что при обращении к базе метаданных объем передаваемых данных невелик, но возникающие задержки могут уменьшить производительность.

Второй тип размещения — развертывание локальной СУБД. Преимуществом при таком размещении будет ускорение работы программы за счет отсутствия задержек и высокой скорости при доступе локальной СУБД. При подобном подходе можно использовать СУБД, располагающиеся в оперативной памяти. К недостаткам можно отнести повышение требований к используемому аппаратному обеспечению и необходимость установки и настройки СУБД.

Система реализована на языке программирования Python 3, технологические СУБД — MySQL, внешние базы данных — MySQL и PostgreSQL.

Исследование времени выполнения алгоритмов декомпозиции запросов

Для системы управления запросами к гетерогенным СУБД общее время выполнения запроса складывается из времени, затраченного на построение плана выполнения запроса и на выполнение плана. При выполнении плана основные затраты времени приходятся на локализацию, состоящую в получении данных из удаленных СУБД и в помещении их во временную базу, а также операции соединения, проводимые во временной базе.

Постановка эксперимента. При составлении запросов с соединениями типа «декартово произведение» применялись таблицы-словари небольшого размера. Поскольку размер итоговой таблицы в строках при «декартовом произведении» равен произведению числа строк всех входящих в него таблиц, при использовании больших таблиц время вычисления результата возрастало бы многократно, и поэтому на практике такие соединения не используются.

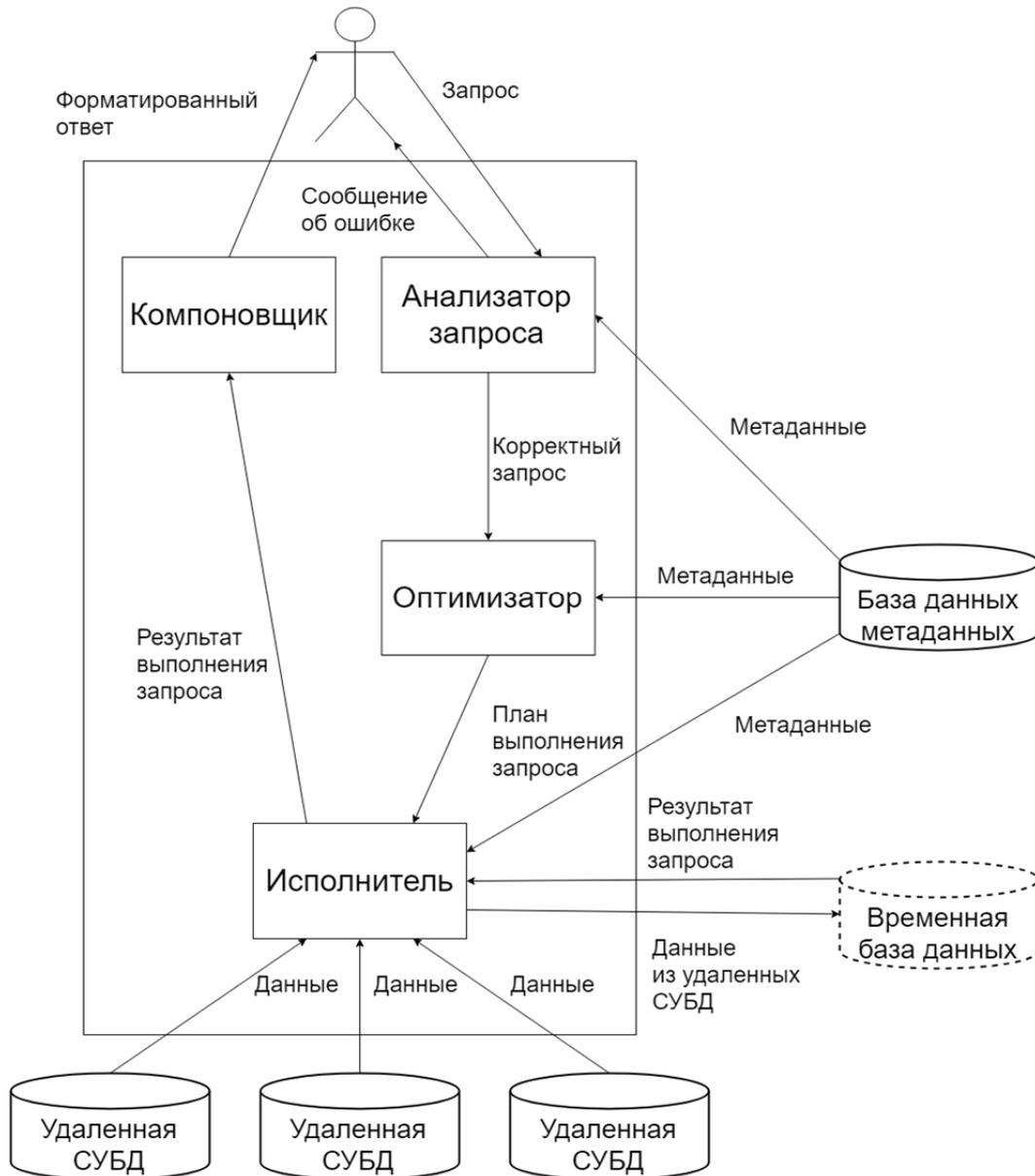


Рис. 3. Структурная схема системы управления выполнением запросов

Fig. 3. Structural diagram of query execution control system

При составлении запросов с соединениями по правилу использовались запросы с правилами соединения типа «первичный ключ – внешний ключ». Такое соединение уже позволяло включать в запрос таблицы больших размеров порядка 10^5 строк.

Составление запросов для смешанного типа соединений производилось путем удаления правил соединения из запросов типа соединение с условием. Для дополнительной проверки времени выполнения системы с

большими и малыми объемами записей запросы для последних двух типов соединений были повторно выполнены, но при сниженном объеме входящих в них таблиц.

Для проверки влияния скорости передачи данных на время выполнения запроса все проверки были выполнены как с доступом к СУБД через глобальную сеть Интернет, так и с доступом через локальную сеть. Параметры эксперимента приведены в табл. 2.

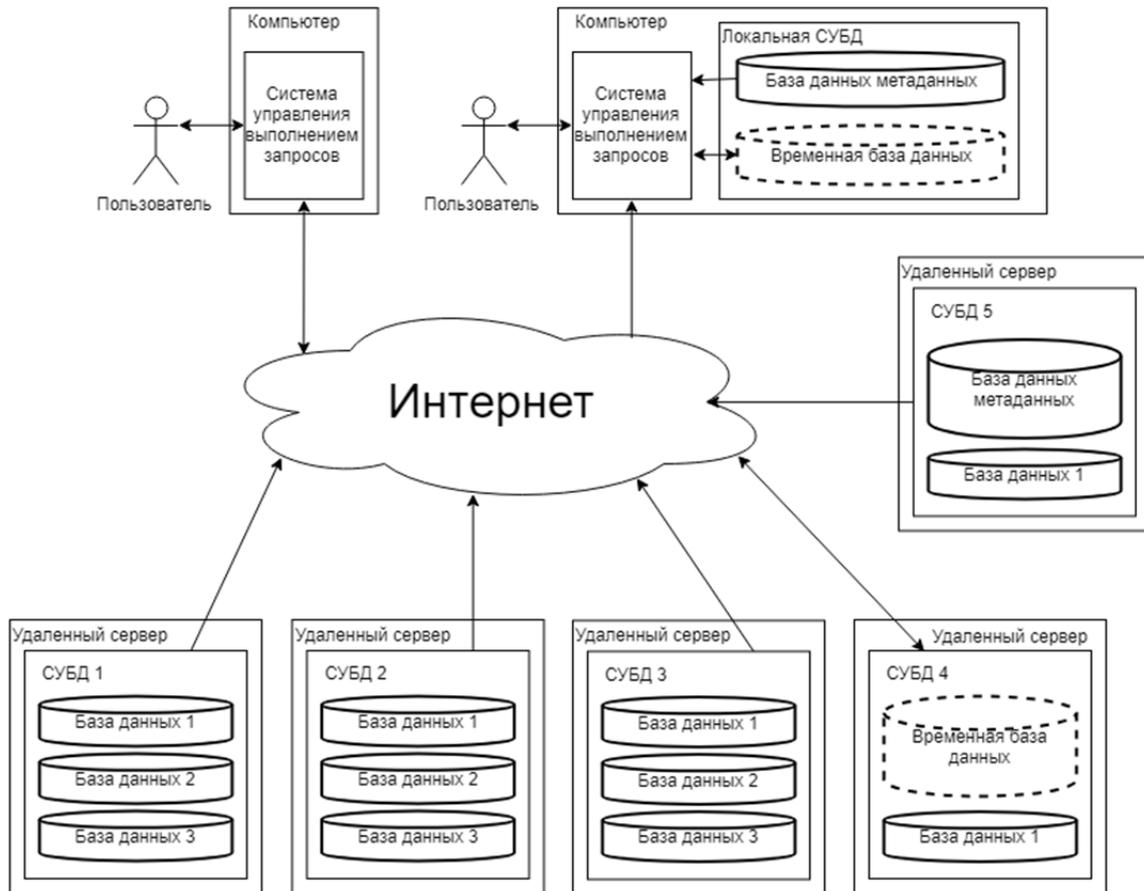


Рис. 4. Функциональная схема системы управления выполнением запросов

Fig. 4. Functional diagram of query execution control system

Таблица 2

Параметры и константы эксперимента

Table 2

Experiment parameters and constants

Параметр или константа	Значение
Параметры данных	
Типы СУБД	MySQL, PostgreSQL
Число внешних серверов БД, шт.	4
Типы сетей	Локальная сеть, Интернет
Объем данных, строки	70000, 5000
Типы соединений	«Декартово произведение», «соединение по правилу», «смешанные соединения»

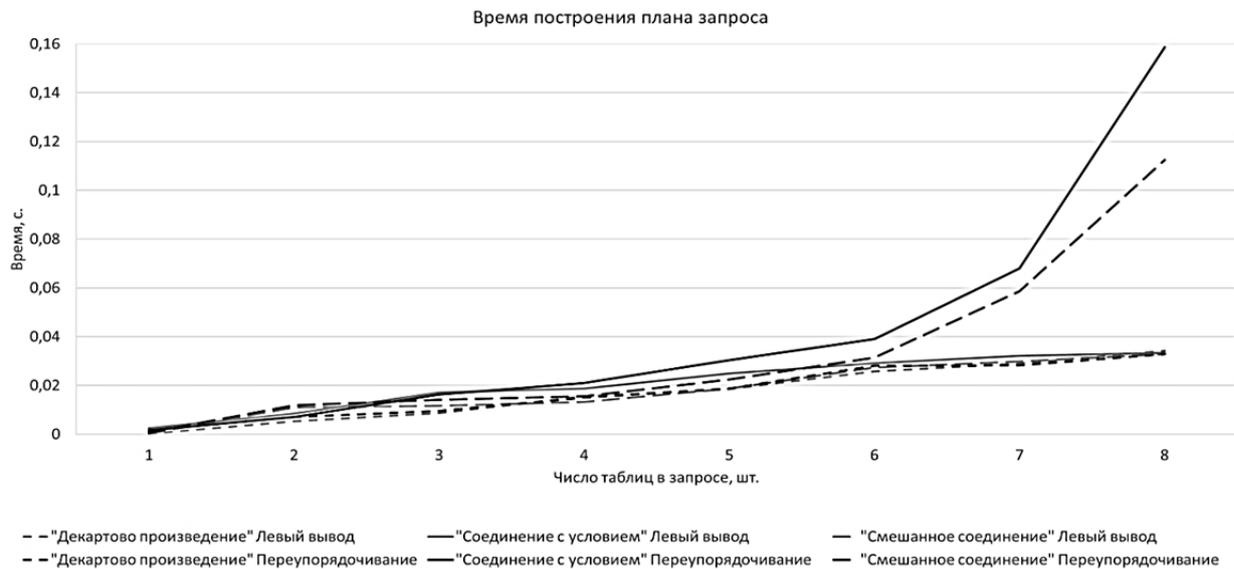


Рис. 5. Зависимость времени построения плана от числа таблиц в запросе

Fig. 5. Dependence of the plan building time on the tables count in the query

СУБД MySQL использовалась как для создания удаленных баз, так и для развертывания локального репозитория для размещения в нем базы данных метаданных и временной базы данных. СУБД PostgreSQL использовалась только для создания удаленных баз.

Для проведения экспериментов написано 40 запросов, по 8 запросов для каждого типа соединения с соответствующим возрастанием числа соединений, и по 8 запросов к СУБД MySQL и к СУБД PostgreSQL. Каждый эксперимент повторялся 20 раз.

Эксперимент 1. Выявление различий во времени построения плана для двух алгоритмов декомпозиции «левого вывода» и «перестановками» для построения плана для трех типов запросов на выборку.

Постановка эксперимента 1. Для каждого из двух алгоритмов и каждого из трех типов запросов «декартово произведение», «соединение по правилу», «смешанные соединения» последовательно выполнить запросы на выборку, содержащие от одной до пяти соединяемых таблиц.

На рис. 5 приведены результаты сравнения времени выполнения для запросов только с соединениями по условию, для запросов только с соединениями типа «декартово произведение» и для смешанного типа запросов. Данные, полученные в результате эксперимента, показывают, что время построения плана алгоритма деком-

позиции левым выводом и алгоритма с переупорядочиванием одного порядка.

Для запросов типа «декартово произведение» время выполнения совпадает за счет эвристики соединения таблиц в порядке возрастания числа строк в таблицах.

Время выполнения переупорядочивания зависит только от типа и количества соединений, поэтому для сравнения достаточно привести результаты только одной конфигурации расположения серверов и объемов, обрабатываемых данных. Наихудшим вариантом должно быть наличие в запросе только «декартовых произведений», поскольку в этом случае происходит полный обход дерева всех возможных комбинаций в глубину. Однако в алгоритме переупорядочивания учтены свойства оценок запросов только с соединениями типа «декартово произведение»: для всех возможных вариантов соединения верхняя оценка будет одинаковой и точной. Соответственно, нет необходимости в переборе всех возможных соединений, достаточно выбрать одно – соединение по возрастанию размеров таблиц в строках, поскольку оно обеспечивает наименьшее число операций соединения кортежей. Наилучший вариант – это наличие в запросе только соединений с условием. В данном случае также требуется полный обход дерева в глубину, но дерево состоит не из всех возможных соединений, а только из соединений, удо-

влетворяющих условиям. Соответственно, при смешении двух типов соединений результат должен находиться между наилучшим и наилучшим вариантами.

Наибольшее расхождение между временем построения наблюдается для запросов типа «соединение с условием», причем расхождение увеличивается при увеличении числа таблиц в соединении. Причина такого расхождения в увеличении размеров дерева возможных соединений, которое влечет за собой дополнительный перебор вариантов соединений, а значит, и наиболее затратную с точки зрения вычислительных ресурсов операцию – проверку допустимости соединения. В точке максимального расхождения разница во времени составляет порядка 0,1 с.

Для уменьшения числа рассматриваемых поддеревьев и, соответственно, уменьшения времени построения плана, используется отсечение заведомо неподходящих ветвей. Отсечение ветвей происходит при условии, что оценка в текущем узле дерева больше или равна текущей минимальной оценке. Поскольку поддеревья, которые начинаются с соединений типа «декартово произведение», быстро отсекаются, общее время построения уменьшается. Поэтому разница между двумя планами увеличивается тем больше, чем больше в запросе соединений с условием, и поэтому время построения плана для смешанных соединений меньше, чем для соединений с условием.

Эксперимент 2. Выявление различий времени выполнения планов запросов, построенных двумя алгоритмами.

Постановка эксперимента 2. Для баз данных больших и малых объемов данных, расположенных в локальной и глобальной сетях, и трех типов запросов («декартово произведение», «соединение по правилу», «смешанные соединения») выполнить план соединения от одной до пяти таблиц.

Время выполнения запроса складывается из времени передачи данных по сети и времени соединения этих данных во временной базе данных. При такой конфигурации результаты для локальной сети в большей степени будут демонстрировать непосредственно время соединения локализованных таблиц во временной базе, поскольку в локальной сети влияние задержек минимально, а

скорость передачи стабильная и высокая. Для глобальной сети на результаты будет оказываться влияние больших задержек и нестабильной скорости передачи данных.

Цель данного эксперимента – выявить, существует ли преимущество во времени выполнения плана запроса, построенного алгоритмом с переупорядочиванием, и каков вклад времени передачи данных между таблицами в общее время выполнения запроса.

На рис. 6, 7 приведены результаты выполнения планов, построенных алгоритмами декомпозиции для запросов типа «декартово произведение», «соединение с условием» и «смешанное соединение», для больших и малых объемов данных при соединении через локальную сеть. Время выполнения плана, построенного алгоритмом с переупорядочиванием, не во всех случаях показало уменьшение времени выполнения. Так, для малых объемов данных не выявлено снижение времени выполнения. Для больших объемов данных выполнение плана, построенного алгоритмом декомпозиции с переупорядочиванием, занимало меньше времени, чем выполнение плана, построенного алгоритмом декомпозиции левым выводом. При условии, что максимальное увеличение времени построения плана алгоритмом с перестановками по сравнению с левым выводом порядка 0,1 с, а максимальное уменьшение времени выполнения порядка 53 с. При таком значительном выигрыше, с учетом малых затрат для получения этого выигрыша, алгоритм декомпозиции с перестановками является более предпочтительным для построения планов выполнения запросов к гетерогенным СУБД больших объемов, чем алгоритм декомпозиции левым выводом. Для малых объемов может наблюдаться незначительное увеличение времени выполнения.

На рис. 8, 9 приведены результаты выполнения планов, построенных алгоритмами декомпозиции для запросов типа «декартово произведение», «соединение с условием» и «смешанное соединение», для больших и малых объемов данных при соединении через глобальную сеть. Нестабильная скорость соединения и задержки не оказывают значимого воздействия на разницу во времени выполнения.



Рис. 6. Зависимость времени выполнения плана от числа таблиц в запросе для большого объема данных для запросов типа «декартово произведение», «соединение с условием» и «смешанное соединение» в локальной сети

Fig. 6. Dependence of the plan execution time on the tables count in the query for a large amount of data for queries such as «Cartesian product», «conditional join» and «mixed join» in the local area network

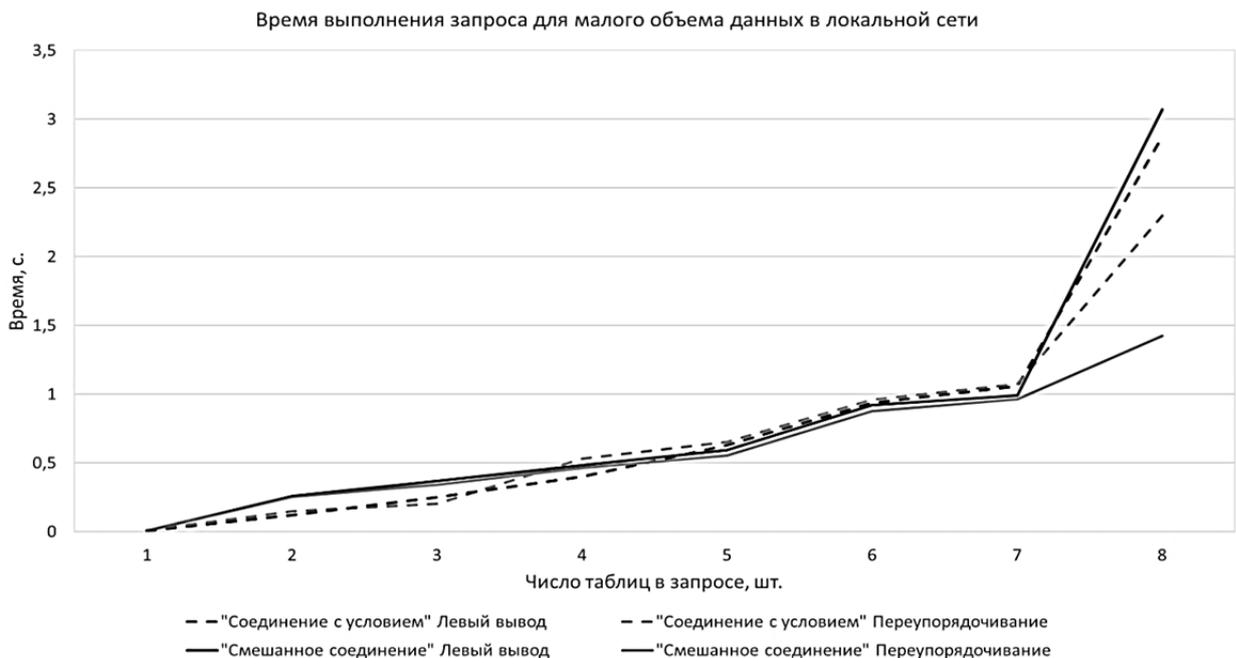


Рис. 7. Зависимость времени выполнения плана от числа таблиц в запросе для малого объема данных для запросов типа «декартово произведение», «соединение с условием» и «смешанное соединение» в локальной сети

Fig. 7. Dependence of the plan execution time on the tables count in the query for a small amount of data for queries such as «Cartesian product», «conditional join» and «mixed join» in the local area network



Рис. 8. Зависимость времени выполнения плана от числа таблиц в запросе для большого объема данных для запросов типа «декартово произведение», «соединение с условием» и «смешанное соединение» в глобальной сети

Fig. 8. Dependence of the plan execution time on the tables count in the query for a large amount of data for queries such as «Cartesian product», «conditional join» and «mixed join» in the wide area network

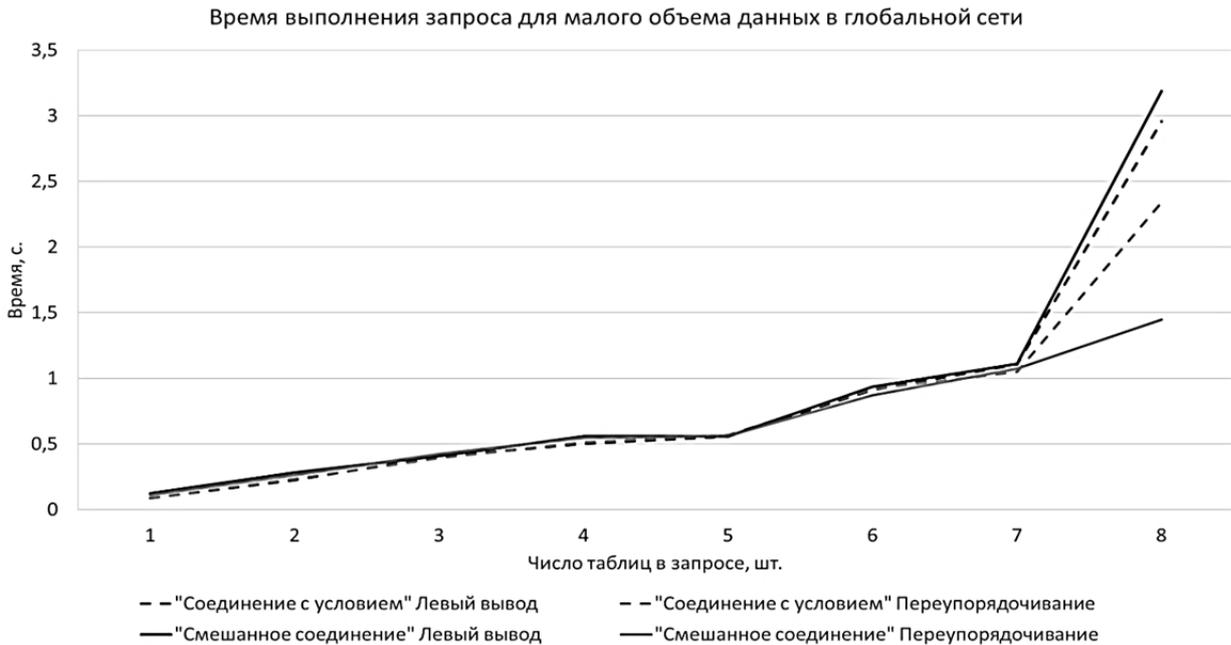


Рис. 9. Зависимость времени выполнения плана от числа таблиц в запросе для малого объема данных для запросов типа «декартово произведение», «соединение с условием» и «смешанное соединение» в глобальной сети

Fig. 9. Dependence of the plan execution time on the tables count in the query for a small amount of data for queries such as «Cartesian product», «conditional join» and «mixed join» in the wide area network

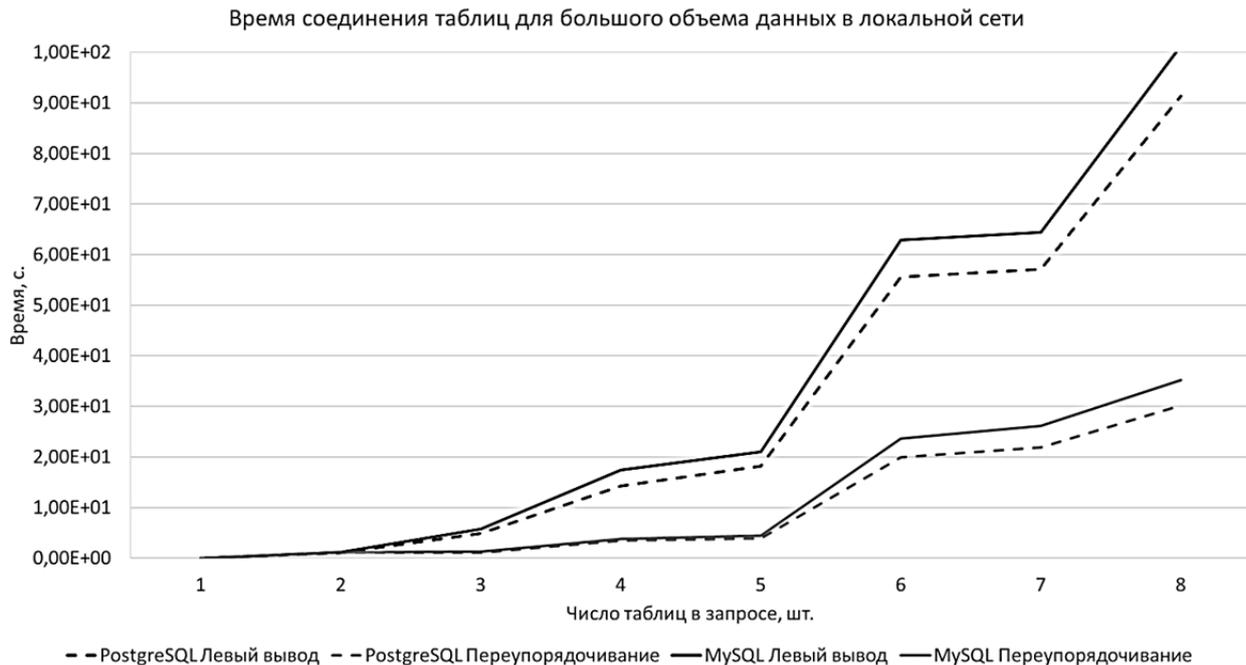


Рис. 10. Зависимость времени соединения таблиц от числа таблиц для большого объема данных для запросов типа «смешанное соединение» для СУБД MySQL и СУБД PostgreSQL

Fig. 10. Dependence of the tables join time on the tables count in the query for a large amount of data for a «mixed join» queries using PostgreSQL and MySQL DBMS

Эксперимент 3. Исследовать время соединения таблиц больших объемов для запросов из 1–5 таблиц на выборку, из систем управления базами данных MySQL, PostgreSQL, расположенных в локальной сети, для трех типов запросов: «декартово произведение», «соединение по правилу», «смешанные соединения».

Постановка эксперимента 3. Для баз данных больших объемов, расположенных в локальной сети, выполнить план соединения запросов смешанного типа от одной до пяти таблиц только к СУБД MySQL и только к СУБД PostgreSQL.

На рис. 10 приведены результаты измерения времени непосредственно соединения таблиц. Время соединения таблиц только из СУБД PostgreSQL значительно ниже, чем время соединения таблиц из MySQL. Учитывая, что запросы содержат таблицы, присутствующие на обоих серверах и заполненные идентичными данными, время соединений должно быть равным. Единственная причина такого расхождения кроется в преобразовании типов. Поскольку временная база данных, используемая для соединения таблиц, находится на MySQL сервере, то для

локализации таблиц необходимо приводить типы данных PostgreSQL к типам MySQL. В данном проекте такое преобразование выполняется при заполнении базы данных метаданными. Соответственно, при соединении во временной базе данных аналогичных таблиц только из СУБД MySQL или только из СУБД PostgreSQL, происходит соединение таблицы с идентичной структурой, но различными типами атрибутов. Такая особенность работы приводит к возникновению разницы во времени соединения.

Заключение

Рассмотрено решение проблемы кусочной автоматизации, состоящей в фрагментации частей единой информационной среды и, как следствие, невозможности получать данные из независимых баз с заранее неизвестной схемой. Решение состоит в учете динамики семантической схемы данных.

Целью являлась разработка и исследование алгоритмов построения интегрированных запросов на выборку данных к гетерогенным распределенным системам управления базами данных. Для проверки работоспособности и эффективности предложен-

ного алгоритма разработан макет программного обеспечения промежуточного слоя, обеспечивающий хранение метаинформации объединяемых баз данных и реализующий интерпретатор запросов с возможностью их декомпозиции и выполнения.

В качестве метода выполнения запросов на выборку данных к гетерогенным СУБД реализована технология централизованной обработки. Она заключается в гомогенизации данных гетерогенных источников, объединении их в единое хранилище и выполнении запроса в этом хранилище. Метод выполнения запроса содержит четыре алгоритма: алгоритм динамической проверки, алгоритм декомпозиции левым выводом, алгоритм декомпозиции с переупорядочиванием и алгоритм выполнения плана. Алгоритм динамической проверки осуществляет динамическую синтаксическую проверку корректности запроса. Алгоритм декомпозиции левым выводом производит построение плана на основе левого обхода дерева соединений запроса. Алгоритм декомпозиции с переупорядочиванием изменяет порядок соединений в запросе на порядок с наименьшей оценкой. Алгоритм выполнения плана производит ленивое выполнение плана запроса, построенного алгоритмами декомпозиции.

Для реализации технологии централизованной обработки разработана архитектура системы управления выполнением запроса, состоящая из модулей, реализующих алгоритмы. Структура системы обеспечивает варианты размещения внешних компонентов как в локальной сети, так и в глобальной сети Интернет. Для хранения метаданных используется технологическая база данных метаданных.

Для исследования эффективности поставлены три эксперимента. Первый – для выявления различий во времени построения плана для двух алгоритмов декомпозиции левым выводом и перестановками для построения плана для трех типов запросов на выборку. Второй – для выявления различий времени выполнения планов запросов, построенных двумя алгоритмами, для локальной и глобальной сетей. Третий – для выявления различий во времени соединения таблиц для систем управления базами данных MySQL и PostgreSQL.

В результате проведения экспериментов установлено, что при сравнимом времени построения плана алгоритмом декомпозиции левым выводом и алгоритмом декомпозиции с переупорядочиванием время выполнения запроса и для локальной, и для глобальной сети время выполнения плана, построенного алгоритмом декомпозиции с переупорядочиванием, всегда меньше для больших объемов и совпадает для малых объемов для исследуемых типов запросов «декартово произведение», «соединение с условием» и «смешанное соединение» при различном числе таблиц в запросах.

При анализе результатов эксперимента по сравнению СУБД MySQL и СУБД PostgreSQL показано, что время соединения во временной базе данных зависит от трансляции типов данных из удаленных гетерогенных СУБД в типы данных СУБД, где располагается временная база данных.

Работа подготовлена в ходе реализации комплексного проекта в рамках Постановления Правительства РФ от 09.04.2010 № 218 при финансовой поддержке Министерства образования и науки РФ. Договор № 03.G25.31.0259 от 28.04.2017.

СПИСОК ЛИТЕРАТУРЫ

1. **Shin J.H., Rusu F., Suhan A.** Exact selectivity computation for modern in-memory database query optimization // arXiv preprint arXiv:1901.01488. 2019.
2. **Marcus R., Papaemmanouil O.** Towards a hands-free query optimizer through deep learning // arXiv preprint arXiv:1809.10212. 2018.
3. **Krishnan S., et al.** Learning to optimize join queries with deep reinforcement learning // arXiv preprint arXiv:1808.03196. 2018.
4. **Vaidehi V., Devi D.S.** Distributed database management and join of multiple data streams in wireless sensor network using querying techniques // 2011 Internat. Conf. on Recent Trends in Information Technology. IEEE, 2011. Pp. 594–599.
5. **Chen F.X., Xie X.S.** Application on query of distributed database based on improved genetic algorithm // Applied Mechanics and Materials. Trans Tech Publications, 2014. Vol. 556. Pp. 4617–4621.
6. **Shao Hua Liu, Xing Xu** Distributed database query based on improved genetic algorithm // Proc. of the 2016 3rd Internat. Conf. on Information Science and Control Engineering. IEEE, 2016. Pp. 348–351.

7. **Gharibi W., Mousa A.** Query optimization based on time scheduling approach // *East-West Design & Test Symposium 2013*. IEEE, 2013. Pp. 1–7.
8. **Begoli E., et al.** Apache calcite: A foundational framework for optimized query processing over heterogeneous data sources // *Proc. of the 2018 Internat. Conf. on Management of Data*. ACM, 2018. Pp. 221–230.
9. **Hameurlain A., Morvan F.** Evolution of query optimization methods // *Transactions on Large-Scale Data-and Knowledge-Centered Systems I*. Springer, Berlin, Heidelberg, 2009. Pp. 211–242.
10. Документация высокопроизводительной системы выполнения распределенных запросов Presto // URL: <https://prestosql.io/docs/current/optimizer/statistics.html> (Дата обращения: 20.04.2019).
11. **Кропотин А.А., Ивашко А.Г.** Реализация метода идентификации семантических конфликтов метаданных и несоответствия интегрируемых данных исходя из их семантического описания // *Вестник ТюмГУ: Физико-математическое моделирование. Нефть, газ, энергетика*. 2017. Т. 3. № 2. DOI: 10.21684/2411-7978-2017-3-2-115-127
12. **Павлов С.В., Ефремова О.А.** Онтологическая модель интеграции разнородных по структуре и тематике пространственных баз данных в единую региональную базу данных // *Онтология проектирования*. 2017. Т. 7. № 3(25). DOI: 10.18287/2223-9537-2017-7-3-323-333
13. **Герасимов А.Н. и др.** Методы автоматизированного извлечения метаданных научных публикаций для библиографических и реферативных баз цитирования // *Интернет и современное общество: Труды объединенной научной конференции*. 2016. С. 41–48.
14. **Петрова Н.В. и др.** Разработка базы метаданных «Информационные ресурсы единой геофизической службы РАН» // *Современные методы обработки и интерпретации сейсмологических данных*. 2018. С. 199–203.
15. **Зыкин В.С., Цымблер М.Л.** Обновление многотабличных представлений на основе коммутативных преобразований базы данных // *Вестник ЮУрГУ. Вычислительная математика и информатика*. 2019. Т. 8. № 2. С. 92–106.
16. **Чубаров Д.Л., Добрецов Н.Н., Кихтенко В.А.** Отображение модели данных NETCDF в реляционную модель для работы с коллекциями данных дистанционного зондирования // *Обработка пространственных данных в задачах мониторинга природных и антропогенных процессов*. 2017. С. 324–328.
17. **Пестунов И.А., Федотов А.М., Жижимов О.Л.** Структура сервисов управления метаданными для разнородных информационных систем. 2015.
18. **Рублев В.С.** Отношение истории и динамика схем баз данных СУБД DIM // *Моделирование и анализ информационных систем*. 2015. Т. 19. № 2. С. 97–108.
19. **Иванова Е.В., Соколинский Л.Б.** Декомпозиция операций пересечения и соединения на основе доменно-интервальной фрагментации колоночных индексов // *Вестник Южно-Уральского государственного университета. Вычислительная математика и информатика*. 2015. Т. 4. № 1. DOI: 10.14529/cmse150104
20. **Sangkla K., Seresangtakul P.** Information integration of heterogeneous medical database systems using metadata // *21st Internat. Computer Science and Engineering Conf. Bangkok*, 2017. Pp. 1–5.

Статья поступила в редакцию 04.05.2019.

REFERENCES

1. **Shin J.H., Rusu F., Suhan A.** Exact selectivity computation for modern in-memory database query optimization. *arXiv preprint arXiv:1901.01488*. 2019.
2. **Marcus R., Papaemmanouil O.** Towards a hands-free query optimizer through deep learning. *arXiv preprint arXiv:1809.10212*. 2018.
3. **Krishnan S., et al.** Learning to optimize join queries with deep reinforcement learning. *arXiv preprint arXiv:1808.03196*. 2018.
4. **Vaidehi V., Devi D.S.** Distributed database management and join of multiple data streams in wireless sensor network using querying techniques. *2011 International Conference on Recent Trends in Information Technology*. IEEE, 2011, Pp. 594–599.
5. **Chen F.X., Xie X.S.** Application on query of distributed database based on improved genetic algorithm. *Applied Mechanics and Materials. Trans Tech Publications*, 2014, Vol. 556, Pp. 4617–4621.
6. **Shao Hua Liu, Xing Xu** Distributed database query based on improved genetic algorithm. *Proceedings of the 2016 3rd International Conference on Information Science and Control Engineering*. IEEE, 2016. Pp. 348–351.
7. **Gharibi W., Mousa A.** Query optimization based on time scheduling approach. *East-West Design & Test Symposium 2013*. IEEE, 2013, Pp. 1–7.
8. **Begoli E., et al.** Apache calcite: A foundational framework for optimized query processing over heterogeneous data sources. *Proceedings of the 2018 International Conference on Management of Data*. ACM, 2018, Pp. 221–230.
9. **Hameurlain A., Morvan F.** Evolution of query optimization methods. *Transactions on Large-*

Scale Data-and Knowledge-Centered Systems I. Springer, Berlin, Heidelberg, 2009, Pp. 211–242.

10. Dokumentatsiya vysokoproizvoditelnoy sistemy vypolneniya raspredelennykh zaprosov Presto [Presto 318 Documentation]. Available: <https://prestosql.io/docs/current/optimizer/statistics.html> (Accessed: 20.04.2019). (rus)

11. **Kropotin A.A., Ivashko A.G.** Realizatsiya metoda identifikatsii semanticheskikh konfliktov metadannykh i nesootvetstviya integriruyemykh dannykh iskhodya iz ikh semanticheskogo opisaniya [Implementation of a method for identifying semantic conflicts of metadata and inconsistency of merging data based on their semantic description]. *Vestnik TyumGU: Fiziko-matematicheskoye modelirovaniye. Neft, gaz, energetika* [Tyumen State University Herald. Physical and Mathematical Modeling. Oil, Gas, Energy], 2017, Vol. 3, No. 2. (rus) DOI: 10.21684/2411-7978-2017-3-2-115-127

12. **Pavlov S.V., Yefremova O.A.** Ontologicheskaya model integratsii raznorodnykh po strukture i tematike prostranstvennykh baz dannykh v yedinuyu regionalnuyu bazu dannykh [Ontological model for integration of structurally heterogeneous spatial databases of various subject areas into a uniform regional database]. *Ontologiya proyektirovaniya* [Ontology of designing], 2017, Vol. 7, No. 3 (25). (rus) DOI: 10.18287/2223-9537-2017-7-3-323-333

13. **Gerashimov A.N., et al.** Metody avtomatizirovannogo izvlecheniya metadannykh nauchnykh publikatsiy dlya bibliograficheskikh i referativnykh baz tsitirovaniya [Automated methods of metadata extraction from scientific publications for bibliographic databases]. *Trudy obyedinennoy nauchnoy konferentsii «Internet i sovremennoye obshchestvo»* [Proceedings of the Joint Scientific Conference on *Internet and Modern Society*], 2016, Pp. 41–48. (rus)

14. **Petrova N.V., et al.** Razrabotka bazy metadannykh Informatsionnye Resursy Yedinoy Geofizicheskoy Sluzhby RAN [Development of the metadata base «Information resources of the unified geophysical service of the RAS»]. *Sovremennye Metody Obrabotki i Interpretatsii Seysmologicheskikh Dannykh* [Proceedings at the XIII International Seismological Workshop Modern

Methods and Interpretation of Seismological Data], 2018, Pp. 199–203. (rus)

15. **Zykin V.S., Tymbler M.L.** Obnovleniye mnogotablichnykh predstavleniy na osnove kommutativnykh preobrazovaniy bazy dannykh [Updating of multi-table views based on commutative database transformations]. *Vestnik YuUrGU. Vychislitel'naya matematika i informatika* [Bulletin of the South Ural State University. Computational Mathematics and Software Engineering], 2019, Vol. 8, No. 2, Pp. 92–106. (rus)

16. **Chubarov D.L., Dobretsov N.N., Kikhthenko V.A.** Otobrazheniye modeli dannykh NETCDF v relyatsionnyuyu model dlya raboty s kollektiyami dannykh distantsionnogo zondirovaniya [Mapping the NETCDF data model to a relational model for working with remote sensing data collections]. *Obrabotka prostranstvennykh dannykh v zadachakh monitoringa prirodnykh i antropogennykh protsessov* [Spatial Processing in Tasks of Monitoring Natural and Anthropogenic Processes], 2017, Pp. 324–328. (rus)

17. **Pestunov I.A., Fedotov A.M., Zhizhimov O.L.** Struktura servisov upravleniya metadannymi dlya raznorodnykh informatsionnykh sistem [The structure of metadata management services for heterogeneous information systems]. 2015. (rus)

18. **Rublev V.S.** Otnosheniye istorii i dinamika skhem baz dannykh SUBD DIM [Evolution of DBMS DIM database schemes]. *Modelirovaniye i analiz informatsionnykh sistem* [Model. Anal. Inform. Sist.], 2015, Vol. 19, No. 2, Pp. 97–108. (rus)

19. **Ivanova Ye.V., Sokolinskiy L.B.** Dekompozitsiya operatsiy peresecheniya i soyedineniya na osnove domenno-intervalnoy fragmentatsii kolonochnykh indeksov [Decomposition of intersection and join operations based on the domain-interval fragmented column indices]. *Vestnik Yuzhno-Uralskogo gosudarstvennogo universiteta. Vychislitel'naya matematika i informatika* [Bulletin of the South Ural State University. Computational Mathematics and Software Engineering], 2015, Vol. 4, No. 1. (rus) DOI: 10.14529/cmse150104

20. **Sangkla K., Seresangtakul P.** Information integration of heterogeneous medical database systems using metadata. *21st International Computer Science and Engineering Conference*, Bangkok, 2017, Pp. 1–5.

Received 04.05.2019.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

ПОПОВ Сергей Геннадьевич

POPOV Sergey G.

E-mail: popovserge@spbstu.ru

ПУРИЙ Александр Александрович

PURIИ Alexandr A.

E-mail: puri.alexander@gmail.com



Моделирование вычислительных, телекоммуникационных, управляющих и социально-экономических систем

DOI: 10.18721/JCSTCS.12206
УДК 519.6:527

МЕТОДИКА ОПРЕДЕЛЕНИЯ РАССТОЯНИЯ ОТ ТОЧКИ ДО ОТРЕЗКА В ЗАДАЧАХ НАВИГАЦИИ

В.А. Ботнев¹, С.М. Устинов²

¹ Акционерное общество «Вяртсила Цифровые Технологии»,
Санкт-Петербург, Российская Федерация;

² Санкт-Петербургский политехнический университет Петра Великого,
Санкт-Петербург, Российская Федерация

Предложена эффективная методика для определения расстояния между отрезком и точкой на поверхности сфероида для решения последующих задач управления. Указаны причины возникновения и метод выявления многоэкстремальности в предложенной задаче. Представлен алгоритм ее решения в этих условиях. В случае многоэкстремальности отрезок разбивается на гарантированно унимодальные части, которые анализируются отдельно. Для унимодального отрезка в общем случае применяется гномоническая проекция специального типа (метод проекции), что позволяет обеспечить квадратичную сходимость. Определяются границы применимости метода проекции, и обосновывается оптимальный выбор центра гномонической проекции. При неприменимости метода проекции расстояние определяется с помощью универсального метода минимизации. Методика демонстрирует высокую точность решения, может быть рекомендована для использования в коммерческих навигационных и геодезических программных продуктах и распространена на задачи управления в различных приложениях навигации.

Ключевые слова: методы моделирования, геодезические задачи, локсодромия, ортодромия, задачи многоэкстремальной оптимизации.

Ссылка при цитировании: Ботнев В.А., Устинов С.М. Методика определения расстояния от точки до отрезка в задачах навигации // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2019. Т. 12. № 2. С. 68–79. DOI: 10.18721/JCSTCS.12206

DISTANCE FINDING METHOD BETWEEN A POINT AND A SEGMENT IN NAVIGATION

V.A. Botnev¹, S.M. Ustinov²

¹ Wartsila Digital Technologies,
St. Petersburg, Russian Federation;

² Peter the Great St. Petersburg Polytechnic University,
St. Petersburg, Russian Federation

The authors proposed an efficient method for determining the distance between an interval and a point on the surface of a spheroid to address some ensuing control problems. The authors described detection of multi extremal situations in the proposed

problem and the causes of its occurrence. The authors suggested the algorithm of its solution under such conditions. In the case of multiextremality, the algorithm divides the interval into parts that are analyzed separately. In general case of unimodal intervals the algorithm uses special gnomonic projections with squared degree of convergence. The limits of applicability of the projection method are determined, and the optimal choice of the center of the gnomonic projection is justified. If the method is not applicable the universal minimization method provides the required solution. The method demonstrates high accuracy of the solution and the authors recommend it for use in commercial navigation and geodetic software products. The technique extends to control problems associated with different applications of navigation.

Keywords: modeling methods, geodesic problems, loxodrome, orthodrome, multi-extremal optimization problems.

Citation: Botnev V.A., Ustinov S.M. Distance finding method between a point and a segment in navigation. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2019, Vol. 12, No. 2, Pp. 68–79. DOI: 10.18721/JCSTCS.12206

Введение

Во многих приложениях навигации и геодезии необходимо определять расстояние между точкой и отрезком. При этом отрезок может быть частью одной из линий на поверхности Земли: ортодромы или локсодромы. В первом случае линия (ортодрома) является кратчайшим путем из одной точки в другую по поверхности Земли. Во втором случае в силу ряда практических потребностей и исторических причин (особенно в задачах навигации) линией между точками служит линия постоянного азимута, что отвечает траектории с постоянным курсом (локсодрома) [1].

В многочисленных задачах навигации требуется многократно определять расстояние между точкой и отрезком. Интерес к их анализу не ослабевает уже более полувека [2, 3]. Кроме того, решения некоторых из них предполагают современные стандарты в области мореплавания. В частности, контроль прохождения маршрута судном является одной из основополагающих функций для ЭКНИС (электронно-картографическая навигационно-информационная система). Он необходим согласно указаниям международного стандарта IEC-61174 Ed.4.0 — 2015. Поэтому при разработке соответствующего программного обеспечения нужно уметь решать задачи подобного рода, что предполагает разработку эффективных ал-

горитмов. Одной из лидирующих компаний в этой области является Транзас, продукты которого охватывают полный круг задач для целей навигации. Предлагаемый в настоящей статье алгоритм доведен до реализации в коммерческом продукте Navi-Sailor 4000 этой компании [4]. Рассмотрим некоторые реальные примеры, требующие решения задачи, указанной в заголовке.

Маршрутное рандеву. Судно движется в соответствии с некоторым маршрутом на некотором удалении от него. При движении вдоль такого маршрута, используя AIS (система в судоходстве, служащая для идентификации судов, их габаритов, курса и других данных с помощью радиоволн диапазона УКВ), можно заранее определить какие ещё суда движутся вдоль этого маршрута. Принимая во внимание их координаты, вектор скорости и геометрические размеры, а также учитывая координаты, скорость и размеры собственного судна, можно рассчитать точку встречи на маршруте. Это часто необходимо, т. к. возможная точка встречи может приходиться на узкое место (узкий канал, сужение русла реки, узкий фарватер и т. д.), достаточное по ширине для прохода одного судна, но недостаточное для прохода одновременно двух судов.

Для предотвращения столкновения или посадки на мель нужно заранее изменить скорость собственного судна, чтобы место

встречи судов было там, где ширина фарватера достаточна для расхождения с другим судном. В такой задаче нужно уметь проецировать положение судна на маршрут (судно всегда идет с некоторым отклонением от маршрута), чтобы сопоставить положению судна точку на маршруте. Маршрут состоит из ортодромических или локсодромических отрезков. Поэтому для нахождения проекции судна на маршрут нужно для произвольного положения судна уметь находить расстояние до каждого отрезка маршрута и рассчитывать ближайшую точку на ближайшем отрезке маршрута.

Маршрутный мониторинг. Для отслеживания прохождения судном маршрута с помощью ЭКНИС, как было указано выше, необходимо иметь функцию маршрутного мониторинга. С этой целью маршрут задают последовательностью путевых точек (waypoints), соединенных либо ортодромическими, либо локсодромическими отрезками (плечами маршрута). Каждому плечу маршрута в мониторинге сопоставляется определенная путевая точка – это та точка, на которую нужно идти вдоль этого плеча по ходу движения по маршруту. При старте мониторинга осуществляется привязка судна к определенной путевой точке. Эта привязка реализуется путем выбора ближайшего плеча маршрута. Чтобы выбрать ближайшее плечо, нужно вычислить расстояние от судна до всех отрезков маршрута. В ходе мониторинга для определения отклонения судна (cross track error – ХТЕ) от текущего плеча маршрута также нужно уметь вычислять расстояние от судна до этого плеча.

Выход из опасного региона. В ходе плавания судно может получить информацию об опасном для плавания регионе (например, через NAVTEX, международную автоматизированную систему оповещения). Если вдруг выяснилось, что судно находится в опасном регионе, то в ряде случаев нужно как можно быстрее его покинуть. Границы этого региона могут задаваться локсодромическими или ортодромическими отрезками. Чтобы определить кратчайший путь для выхода из опасного района,

нужно определить ближайший отрезок границы и ближайшую точку на нем (точку выхода из опасного района).

Решение всех этих задач не является тривиальным, т. к. даже при вычислении расстояния между двумя точками приходится сталкиваться с рядом проблем [5–7], в том числе с катастрофической потерей точности [8, 9]. Нахождение расстояния до отрезка очень близко к вопросу нахождения расстояния до линии [10], но обладает некоторыми особенностями. В частности, при нахождении расстояния до линии можно ограничиться только ближайшим к стартовой точке минимумом, в то время как при определении расстояния до отрезка необходимо рассмотреть все минимумы в пределах этого отрезка для получения абсолютного минимума.

Первоначально обратимся к определению отрезка. В Евклидовой геометрии на плоскости отрезок – это часть прямой, заключенная между двумя точками этой прямой. В рамках этой геометрии через две не совпадающие точки можно провести только одну прямую. На поверхности эллипсоида вращения соединение двух точек отрезком решается не так однозначно.

Если на поверхности сфероида аналогом прямых на плоскости считать геодезические (что справедливо не только для геодезии, но и для дифференциальной геометрии), то через любые две точки можно провести бесконечное множество различных геодезических в силу не замкнутости этих линий на такой поверхности. Если дополнительно потребовать, чтобы две точки, задающие отрезок, были соединены кратчайшей линией, то тем самым в подавляющем числе случаев геодезическая, соединяющая эти две точки, будет определяться однозначно. Исключением является ситуация расположения двух точек на противоположных полюсах. Тогда для определенности можно задать какое-либо правило соединения точек, например, по нулевому меридиану.

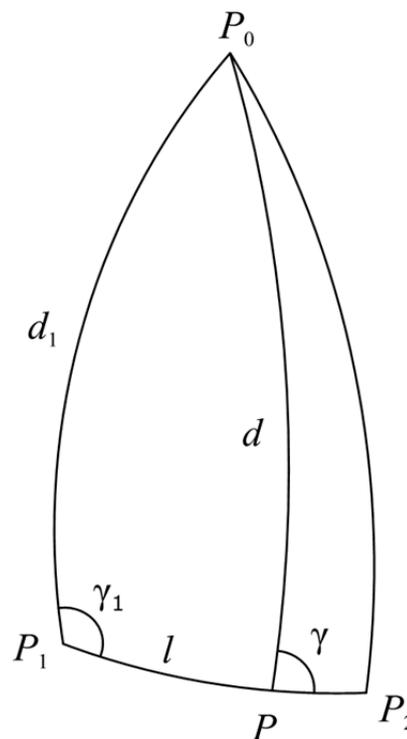
Еще один вариант неоднозначности возникает, когда точки размещены диаметрально противоположно на поверхности

сплюснутого сфероида (каковым является *WGS84*), симметрично относительно центра сфероида, и не совпадают с полюсами. При этом существуют два различных отрезка одинаковой длины, проходящие через северный и южный полюс. Здесь также можно задать правило соединения точек, например, из двух отрезков выбирать тот, который проходит через северный полюс.

Если линией, соединяющей две точки, является локсодрома (что часто осуществляется в навигации), то возникают схожие проблемы. Для их разрешения нужно потребовать (опять же исходя из соображений здравого смысла), чтобы отрезком в этом случае считался кратчайший кусок локсодромы, соединяющий эти две точки. Если точки лежат на противоположных полюсах, их так же, как и в предыдущем случае, можно соединить нулевым меридианом (который является частным случаем локсодромы). Если две точки лежат на противоположных меридианах и не совпадают с полюсами, для однозначного определения отрезка можно выбрать тот, который пересекает нулевой меридиан. В случае нахождения одной из точек на нулевом меридиане, можно выбирать тот отрезок, который находится в восточном полушарии.

Задача определения расстояния от точки P_0 до отрезка P_1P_2 (см. рисунок) может ставиться в четырех постановках. Отрезок P_1P_2 может быть частью локсодромы или ортодромы, а расстояние от точки P_0 до него, в свою очередь, может определяться как по ортодроме, так и по локсодроме.

Остановимся на варианте с ортодромическим отрезком и ортодромическим расстоянием, поскольку в данной статье речь будет идти только об этом случае. Определение минимального расстояния часто сводится к стандартной задаче минимизации без учета специфики предметной области, что оказывается весьма трудоемким. Для повышения эффективности в работах [5, 6] предложено использовать т. н. гномоническую проекцию. Однако этот прием не является универсальным. Предлагаемый здесь метод свободен от указанных недостатков.



Точка P_0 и отрезок P_1P_2 на сфероиде
Point P_0 and segment P_1P_2 on a spheroid

Об актуальности задачи свидетельствуют многочисленные дискуссии среди разработчиков соответствующего программного обеспечения. Одним из направлений решения этой задачи являлось использование сферического приближения. В целом ряде приложений получающаяся погрешность приемлема. Это относится не только к отдельным задачам морской навигации, но и к задачам воздушной навигации, где использование такой идеализации популярно [11–16]. Однако в тех случаях, когда плечи маршрута достаточно велики, и необходима высокая точность позиционирования, погрешность сферического приближения может оказаться неприемлемо большой [17].

Другим направлением определения расстояния от точки до отрезка является решение задачи с использованием классических методов минимизации на базе решения прямой и обратной геодезических задач. При этом на каждом шаге находится расстояние от фиксированной точки до

точки отрезка, а точка отрезка варьируется в соответствии с выбранным алгоритмом минимизации. В ряде приложений такой подход является приемлемым, если общее время решения не выходит за разумную величину. В частности, такой подход реализован в продукте Transas Navi-Sailor 4000 в версиях до 2.00.340. Однако требования новых стандартов для ЭКДИС по представлению маршрутов привели к увеличению времени исполнения необходимых расчетов. Это привело к тому, что использование классических методов минимизации стало неприемлемым для длинных маршрутов (состоящих из нескольких тысяч точек). Именно этот факт и продиктовал необходимость поиска более эффективного для данной задачи метода минимизации. Выяснилось, что с проблемой недостаточной эффективности классических методов минимизации столкнулись и при разработке программных продуктов других компаний [18].

Следует также отметить подход, связанный с разбиением ортодромических участков на локсодромические отрезки с последующим решением задачи уже для локсодромы. Такой способ также популярен, и соответствующие алгоритмы входят в состав Navi-Sailor. Популярность данного подхода обусловлена доминирующим положением карт в проекции Меркатора в морской навигации, где локсодромические отрезки являются прямыми линиями. Более того, двигаться по локсодроме, используя компас или гироскоп, значительно проще, чем по ортодроме, т. к. достаточно удерживать постоянный курс. Однако прохождение маршрута по ортодромическим плечам с использованием GPS и автоматических систем перестало быть проблемой, в то время как добавление промежуточных точек для локсодромического разбиения на маршрутах с большими плечами усложняет структуру маршрута и приводит к ненужной потере времени и топлива.

Если же интересует лишь расстояние до ортодромического отрезка, то отличие между локсодромой и ортодромой может быть

весьма значительным [19]. Это приводит к необходимости разбиения на множество частей с последующим нахождением расстояния до каждой из них, что оказывается неэффективно. Более того, расстояние от точки до локсодромического отрезка должно вычисляться совсем другим способом (одним из эффективных методов её решения видится решение полярного треугольника по двум углам и стороне не между ними). Это далеко не тривиально, и обсуждение ее эффективного решения требует рассмотрения в отдельной статье. Все сказанное делает задачу поиска расстояния по ортодроме весьма актуальной.

Постановка задачи

Рассматривается точка P_0 с широтой φ_0 и долготой λ_0 и ортодромичный отрезок, образованный точками P_1 с широтой φ_1 и долготой λ_1 и точкой P_2 с широтой φ_2 и долготой λ_2 . Необходимо найти расстояние от точки P_0 до отрезка P_1P_2 и точку Q на данном отрезке, где это расстояние реализуется. На сфере ортодрома является замкнутой линией, и минимум в пределах отрезка может быть только один (исключение составляют вырожденные варианты, когда P_0 и продолжение отрезка P_1P_2 на сфере до большого круга соотносятся как полюс и экватор). В случае эллипсоида вращения ортодрома уже не будет замкнутой и имеет достаточно сложное поведение. Поэтому иногда можно наблюдать несколько конкурирующих между собой локальных минимумов в пределах одного отрезка. При этом с практической точки зрения интересным для геодезических задач и для задач навигации чаще всего является именно глобальный минимум на отрезке.

Обозначим длину отрезка P_1P_2 за L . Пусть точка P лежит на отрезке P_1P_2 (см. рисунок), и пусть l — это длина отрезка P_1P , тогда $0 \leq l \leq L$. Предполагая, что позицию точки P на отрезке P_1P_2 можно варьировать, меняя параметр l , получим функцию $d(l)$ расстояния от P_0 до P . Глобальный минимум $d(l)$ на отрезке $[0, L]$ будет решением поставленной задачи.

Решение задачи

В работе [10] предложен алгоритм определения расстояния на сфероиде от точки P_0 до линии, являющейся ортодромой. В свою очередь, ортодрома задавалась стартовой точкой P_1 и азимутом. При определении расстояния от точки P_0 до ортодромы из всех локальных минимумов необходимо было выбрать ближайший к P_1 . Поставленная здесь задача будет решаться по той же схеме, что и в [10].

Существенное отличие заключается в том, что отрезок P_1P_2 — это только часть ортодромы, и среди всех локальных экстремумов требуется найти глобальный минимум.

С методической точки зрения целесообразно первоначально решить данную задачу на сфере, а не на сфероиде. Это позволит не только проиллюстрировать поведение ортодромы на более простой модели, но и даст подсказку, где искать области многоэкстремальности функции $d(l)$ для модели сфероида. На сфере ортодрома (геодезическая) замкнута и представляет собой дугу большого круга. В рамках сферической модели сформулированная задача нахождения расстояния между точкой и отрезком решается аналитически. Кроме того, в [20, 21] доказывалось, что функция $d(l)$ расстояния от точки P_0 до точки P , лежащей на произвольном отрезке P_1P_2 , является унимодальной, т. е. имеет не более одного экстремума в рамках отрезка. Таким образом, для сферы существует всегда только одно решение. Это верно за единственным исключением, когда точка P_0 и ортодрома соотносятся так, как полюс и экватор. В этом случае расстояние от любой точки ортодромы до P_0 одинаково и равно $d = \frac{\pi}{2}a$, где a — радиус сферы; угол между ортодромой и направлением на P_0 для любой точки ортодромы $\gamma = 90^\circ$.

Расширим область определения параметра l на всю ортодрому, на которой лежит отрезок P_1P_2 . В этом случае $-\pi a < l \leq \pi a$, причем отрицательные l отвечают точкам на ортодроме, расположенным по другую

сторону от P_1 относительно P_2 . Если устранить из рассмотрения указанное выше исключение (которое легко определить по значениям $d_1 = \frac{\pi}{2}a$ и $\gamma_1 = 90^\circ$), то значение l_m , при котором достигается минимальное расстояние до точки P_0 , определяется по формуле:

$$\tan(l_m) = (\sin(d_1) * \cos(\gamma_1)) / \cos(d_1). \quad (1)$$

Комбинация знаков числителя и знаменателя позволяет однозначно определить l_m (например, на языке C с помощью стандартной функции atan2).

Решаемая здесь задача — это определение ближайшей к P_0 точки Q лишь в пределах отрезка P_1P_2 . Если $0 \leq l_m \leq L$, то ближайшая точка на всей ортодроме и ближайшая точка на отрезке P_1P_2 суть одно и то же, и положение этой точки будет определяться путем решения прямой геодезической задачи. В противном случае минимальное значение $d(l)$ на отрезке P_1P_2 будет достигаться на одном из концов этого отрезка. Если $-\pi a + L/2 < l_m \leq 0$, то минимальное значение $d(l)$ на отрезке P_1P_2 достигается в точке P_1 , если $l_m < -\pi a + L/2$, или $L \leq l_m$, то оно достигается в точке P_2 . Если $l_m = -\pi a + L/2$, то минимальное значение $d(l)$ на отрезке P_1P_2 будет достигаться одновременно на обоих концах этого отрезка.

Как только известна точка Q , расстояние между точкой P_0 и отрезком P_1P_2 можно определить путем нахождения расстояния от Q до P_0 .

В случае сфероида картина значительно усложняется [22–24]. Во-первых, аналитическое решение в общем случае получить не удастся, а, во-вторых, указанная выше функция $d(l)$ даже на коротких относительно радиуса сфероида отрезках может иметь произвольное число экстремумов. Таким образом, решение в подавляющем большинстве случаев можно получить только численно, и даже если будет найден некий локальный минимум (что чаще всего обеспечивают традиционные методы минимизации), нет полной гарантии, что

он будет глобальным в рамках отрезка P_1P_2 . Это и представляет главную проблему статьи в отличие от [5].

Учитывая ситуацию на сфере, многоэкстремальность следует ожидать в первую очередь, когда точка P_0 и ортодрома соотносятся примерно так же, как соотносятся полюс и экватор в случае сферы (расстояние от точки до ортодромы d будет близким к $\frac{\pi}{2}a$, где a – большая полуось Земли, а угол γ между ортодромой и направлением на P_0 будет близок к 90°). Результаты многочисленных экспериментов, отраженные ниже, полностью подтверждают эту гипотезу. Их целью было установить условия существования нескольких экстремумов в пределах одного отрезка.

Эксперименты выполнялись по той же схеме, по которой решалась задача определения расстояния от точки до линии в работе [5]. Случайным образом 10^5 раз задавались точки P_0 и P_1 (расположенная на ортодроме). Для каждой из них 3600 раз фиксировался начальный азимут α , задающий ортодрому из P_1 (от 0 до 180° с шагом 0,05). В итоге проведено $3,6 \cdot 10^8$ экспериментов, результатом которых стал ряд немногочисленных наборов (P_0 и P_1 и начальный азимут), для которых наблюдалось сближение соседних экстремумов меньше, чем на заданное расстояние $L = 3$, где L измеряется в радиусах экватора. Далее для всей коллекции полученных данных эксперимент повторялся, но с уменьшением шага по углу α и с уменьшением расстояния L . Каждый из этих наборов

был изучен детально для выяснения того, как ведет себя $d(l)$ и угол $\gamma(l)$ в окрестности близких экстремумов.

Итак, пусть L – длина отрезка P_1P_2 , на котором оказывается более чем один экстремум; d_{\min} , d_{\max} – это минимальное и максимальное зафиксированные в эксперименте расстояния от отрезка P_1P_2 до точки P_0 , выраженные в радиусах экватора.

Выяснилось, что d_{\min} и d_{\max} располагаются по обе стороны от числа $\frac{\pi}{2}$, и уже начиная с $L \leq 1,5$ их разность стабилизировалась: $\Delta d = \frac{(d_{\max} - d_{\min})}{2} = 0,0046$. γ_{\min} , γ_{\max} – это минимальное и максимальное на отрезке P_1P_2 значения угла в градусах между ортодромой, на которой лежит этот отрезок, и направлением на точку P_0 . Так же, как и для d , значения γ_{\min} и γ_{\max} оказались по разные стороны от величины 90° . Однако, в отличие от Δd , значение $\Delta \gamma = \frac{(\gamma_{\max} - \gamma_{\min})}{2}$ заметно сокращалось при уменьшении L .

Введем параметр N , связанный с процедурой последовательного уменьшения L и задающий L по формуле $L = \frac{3}{2^N}$. При этом последовательному увеличению N соответствует сокращение L в два раза.

С учетом обозначений $L = \frac{3}{2^N}$, $\Delta \gamma = \frac{(\gamma_{\max} - \gamma_{\min})}{2}$, $\Delta d = \frac{(d_{\max} - d_{\min})}{2}$, $k = \frac{\Delta \gamma}{L^2}$ результаты экспериментов для различных N представлены в таблице.

Результаты экспериментов

The results of the experiments

N	0	1	2	3	4	5	6	7
$\Delta d \cdot 10^2$	5,09	0,46	0,46	0,46	0,46	0,46	0,46	0,46
$\Delta \gamma (^\circ)$	3,09	0,29	0,10	$2,5 \cdot 10^{-2}$	$6,5 \cdot 10^{-3}$	$1,6 \cdot 10^{-3}$	$4,1 \cdot 10^{-4}$	$1,0 \cdot 10^{-4}$
k	0,343	0,131	0,171	0,181	0,184	0,185	0,185	0,185

Эксперименты полностью подтвердили гипотезу « $\frac{\pi}{2}a$ и 90° ». Например, $N = 6$ соответствует отрезку длиной $L = 0,46875$, где унимодальность нарушалась хотя бы один раз, и дальность до точки P_0 изменялась только в узком диапазоне расстояний $\Delta d = 0,0046$, а по углу между касательной к ортодроме и направлением на точку P_0 в диапазоне $\Delta\gamma = 0,0041^\circ$. По мере увеличения N (уменьшения длины L) интервал по γ сокращался по квадратичной зависимости от L . Последний факт легко видеть из последней строки таблицы, где параметр k определялся по формуле $k = \frac{\Delta\gamma}{L^2}$. Таким образом, при достаточно малых L для величины $\Delta\gamma$ имеем:

$$\Delta\gamma = \frac{(\gamma_{\max} - \gamma_{\min})}{2} \approx 0,185L^2. \quad (2)$$

Реализация алгоритма

Выполненные исследования позволяют предложить процедуру нахождения глобального минимума. Для каждого значения L в соответствии с таблицей можно указать интервал по расстоянию d от d_{\min} до d_{\max} , и интервал по углу γ от γ_{\min} до γ_{\max} , для которых возможно существование близких экстремумов. Если выясняется, что для крайних точек отрезка значения d и γ попадают в соответствующие интервалы, то такой отрезок длиной L будем называть «проблемным».

Общая идея алгоритма заключается в следующем:

- если рассматриваемый отрезок не является «проблемным», то функция $d(l)$ унимодальная, и можно предложить алгоритм, описанный в работе [9] (вместо применения какого-либо универсального метода минимизации можно использовать расчет на сфере как начальное приближение для последующего использования метода минимизации в гномонической проекции, о чем речь пойдет ниже);

- если отрезок оказывается «проблемным», то он делится пополам и алгоритм рекурсивно повторяется для каждой его по-

ловины. Следует заметить, что интервал по углу при этом резко сокращается по квадратичной зависимости, как видно из таблицы.

В картографии отображения земной поверхности на плоскость называют проекцией [25]. Гномоническая проекция определяется следующим образом. Пусть точка A – центр проекции, а точка B – произвольная точка (обе находятся на поверхности земного эллипсоида). Тогда точка B проецируется в точку на плоскости с координатами [5, 6]:

$$x = \rho * \sin(\alpha_1), \quad y = \rho * \cos(\alpha_1), \quad \rho = \frac{m_{12}}{M_{12}},$$

где α_1 – азимут из точки A в точку B ; m_{12} – приведенная длина геодезической; M_{12} – коэффициент сближения геодезических (geodesic scale). заметим, что проекция не определена для неположительных значений M_{12} .

Тогда алгоритм реализации пункта (А) выглядит так:

1. Задаем на имеющейся ортодроме точку Q_0 – сферическое приближение к точке минимума Q , причем должна использоваться сфера усредненного радиуса $R = 2a/3 + b/3$ [26], где b – малая полуось, и на этой сфере точки берутся с такими же значениями географических координат и азимутов, как и на эллипсоиде. При этом Q_0 если совпало с P_1 , то поменяем местами P_1 и P_2 .

2. Переводим точки P_0 , P_1 и Q_j на плоскость в точки A_0 , A_1 и B_j , соответственно, выбрав в качестве центра проекции точку Q_j . При этом точки A_1 и B_j окажутся на прямой, отражающей ортодрому, а точка A_0 может оказаться как вне этой прямой, так и на ней.

3. Определяем точку прямой, ближайшую к A_0 . Переводим на эллипсоид найденное решение и получаем точку Q_{j+1} , которая является очередным приближением к ответу.

4. Вычисляем расстояние между Q_j и Q_{j+1} . Если расстояние оказывается меньше максимально допустимого отклонения от истинного минимума, то прерываем расчет, принимая за ответ точку Q_{j+1} . Если –

нет, то переходим к шагу 2, используя Q_{j+1} вместо Q_j .

Общий алгоритм имеет следующий вид.

Шаг 0. Для отрезка длиной L определяем значения Γ_{\min} , Γ_{\max} , D_{\min} , D_{\max} , руководствуясь таблицей или функциональной зависимостью (2).

Шаг 1. Для краевых точек отрезка определяем расстояния D_1 и D_2 , а также углы A_1 и A_2 . Если хотя бы одна из пар (расстояние – угол) не попала в интересующий диапазон, то функция считается унимодальной и используется указанный выше «стандартный» алгоритм.

Шаг 2. Если параметры обеих краевых точек попали в «опасный» диапазон, то определяем d и α для точки P_3 на середине отрезка. После этого повторяем шаги 0 и 1 для отрезка P_1P_3 вдвое меньшей длины ($L/2$). Деление пополам продолжается до тех пор, пока самый левый отрезок не будет признан «нормальным».

Шаг 3. Повторяем шаг 2 для ближайшего правого соседа отрезка, принятого на шаге 2.

Шаг 4. Сравниваем минимальные значения $d(l)$ для всех полученных отрезков и выбираем минимальный из них.

Многочисленные расчеты, выполненные на основе программного продукта Navi-Sailor 4000, в составе которого реализован предложенный алгоритм, показали, что точность нахождения минимального расстояния оказывается весьма высокой и составляет десятки нанометров. При этом точность определения положения точки, где достигается минимум, зависит от того, как сильно изменяется расстояние в окрестности минимума. В подавляющем большинстве вариантов точность составляет около метра. В редких случаях, когда изменение функции в окрестности минимума имеет относительно пологий характер, точность оказывается несколько ниже.

В рамках задачи маршрутного рандеву ближайший отрезок маршрута нужно находить при каждом обновлении позиции отдельно для собственного судна и отдельно

для цели. Одно из основных применений маршрутного рандеву – это речная навигация, где маршруты могут состоять из десятков тысяч точек, а аппаратные вычислительные ресурсы весьма скромные (результат экономии судовладельцев). Все эти факторы формируют высокие требования к эффективности расчета, что и обеспечивает представленный здесь алгоритм. В результате многократных замеров времени вычисления расстояния от судна до речного маршрута Санкт-Петербург – Ростов (около 3500 отрезков) выяснилось, что предложенный алгоритм минимизации дает в среднем выигрыш в 6,2 раза по сравнению с широко известным универсальным алгоритмом минимизации Брента [27].

Успех алгоритма продиктован тем, что вместо решения задачи «в лоб» предлагается использование гномонической проекции. Применение специфики геометрии земной поверхности (вместо использования универсальных численных методов) в прикладных задачах навигации оказалось весьма перспективным направлением [28–30]. Это позволяет существенно сократить время, необходимое для расчета, что очень актуально для облачных вычислений в морской навигации, когда сервер вынужден обрабатывать тысячи запросов от морских судов, подключенных к определенным онлайн-сервисам.

Заключение

Предложена методика решения задачи поиска кратчайшего расстояния между точкой и отрезком геодезической линии (ортодромы). Методику можно распространить на задачи определения расстояния до протяженных объектов, таких, как маршруты или области на поверхности Земли.

Методика позволяет выявлять случаи аномально близких экстремумов. Даны рекомендации по решению задачи в таких ситуациях. Используемый подход может быть рекомендован и в более общих случаях поиска глобального минимума в многоэкстремальных задачах.

На основе методики разработан алгоритм, который был реализован в составе

коммерческого программного продукта Navi-Sailor 4000 компании Transas, широко используемого для морской и речной навигации.

Предложенный алгоритм обеспечивает в среднем шестикратное увеличение производительности по сравнению с универсальным алгоритмом.

СПИСОК ЛИТЕРАТУРЫ

1. **Deakin R.E., Hunter M.N.** Geometric geodesy. Part A // Lecture Notes. School of Mathematical and Geospatial Sciences, RMIT University, Melbourne, Australia, 2008. 140 p.
2. **Faulkner F.D.** Numerical methods for determining optimum ship routes // J. of the Institute of navigation. Winter 1963. Vol. 10. No. 4. Pp. 351–367.
3. **Bijlsma S.J.** A computational method for the solution of optimal control problems in ship routing // J. of the Institute of Navigation. 2001. Vol. 48. No. 3. Pp. 145–154.
4. Transas sea traffic management // URL: <http://s3-eu-west-1.amazonaws.com/stm-stmvalidation/uploads/20180516214207/TRANSAS-STM.pdf> (Дата обращения: 27.05.2019).
5. **Karney C.F.F.** Algorithms for geodesics // J. Geodesy. 2013. Vol. 87(1). Pp. 43–55.
6. **Karney C.F.F.** Geodesics on an ellipsoid of revolution // Technical report. SRI International, Febr. 2011.
7. **Ботнев В.А., Устинов С.М.** Методы решения прямой и обратной геодезических задач с высокой точностью // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2014. № 3(198). С. 49–58.
8. **Press W.H., Teukolsky S.A., Vetterling W.T.** Numerical recipes in C // The Art of Scientific Computing. 3rd Ed. Cambridge University Press, 2007. 1235 p.
9. **Conte S.D., Carl de Boor** Elementary numerical analysis. An algorithmic approach // SIAM-Society for Industrial and Applied Mathematics. 2017. 480 p.
10. **Ботнев, В.А., Устинов С.М.** Методика определения расстояния между точкой и линией в геодезии // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2015. № 6(234). С. 33–44. DOI: 10.5862/JCSTCS.234.4
11. **Hickley P.** Great circle v rhumb line cross track distance at mid-longitude – The final answer // The J. of Navigation. 2008. Vol. 61. Pp. 367–368.
12. Aviation formulary // URL: <http://www.edwilliams.org/avform.htm#XTE> (Дата обращения: 27.05.2019).
13. Distance from point to line on Earth // URL: <https://stackoverflow.com/questions/7803004/distance-from-point-to-line-on-earth> (Дата обращения: 27.05.2019).
14. Distance between lon/lat point and line segment // URL: <https://www.sitepoint.com/community/t/distance-between-long-lat-point-and-line-segment/50583> (Дата обращения: 27.05.2019).
15. Projecting cross-track distance on great circle? // URL: <https://gis.stackexchange.com/questions/209540/projecting-cross-track-distance-on-great-circle> (Дата обращения: 27.05.2019).
16. How to calculate cross-track error (GPS/Core Location) // URL: <https://stackoverflow.com/questions/3997410/how-to-calculate-cross-track-error-gps-core-location> (Дата обращения: 27.05.2019).
17. Cross track distance to great circle line and rhumb line // URL: <https://gis.stackexchange.com/questions/121117/cross-track-distance-to-great-circle-line-and-rhumb-line> (Дата обращения: 27.05.2019).
18. Shortest distance between a point and a geodesic // URL: <https://sourceforge.net/p/geographiclib/discussion/1026621/thread/f122cc28/> (Дата обращения: 27.05.2019).
19. Cross-track difference rhumb vs great circle // URL: <http://www.atpforum.eu/forum/technicalsubjects/-071-operational-procedures-a/1315-cross-track-difference-rhumb-vs-great-circle> (Дата обращения: 27.05.2019).
20. **Kells L.M.** Plane and spherical trigonometry // Andesite Press, 2017. 526 p.
21. **Кранц П.** Сферическая тригонометрия. ЛКИ, 2019. 96 с.
22. **Helmert F.R.** Die mathematischen und physikalischen Theorem der höheren geodäsie // Die mathematischen Theorem. Leipzig, 1880. Vol. 1. 631 p.
23. **Rapp R.H.** Geometric geodesy. Part II // Technical report. Ohio State University, 1993. 177 p.
24. **Meyer T.H.** Introduction to geometrical and physical geodesy. ESRI Press, 2018. 260 p.
25. **Snyder J.P.** Map projection-a working manual // Professional Paper 1395. U.S. Geological Survey, 1987. 383 p.
26. **McCaw G.T.** Long lines on the Earth // Empire Survey Review. 1932. No. 1 (6). Pp. 259–263.
27. **Brent R.P.** Algorithms for minimization without derivatives. Englewood Cliffs, NJ: Prentice-Hall, 1973. 195 p.
28. **Мазуров Б.Т.** Линейные искажения при переходе от эллипсоида Красовского на секущий эллипсоид // Геодезия и картография. 2018. Т. 79. № 1. С. 8–13.

29. **Виноградов А.В., Мазуров Б.Т.** Перспективы использования специальных геодезических проекций и местных систем координат // Вестник СГУГиТ. 2017. Т. 22. № 1. С. 18–29.

30. **Соколов С.В.** Аналитические модели пространственных траекторий для решения задач навигации // Прикладная математика и механика. 2015. Т. 79. Вып. 1. С. 24–30.

Статья поступила в редакцию 03.04.2019.

REFERENCES

1. **Deakin R.E., Hunter M.N.** Geometric geodesy. Part A. *Lecture Notes*, School of Mathematical and Geospatial Sciences, RMIT University, Melbourne, Australia, 2008, 140 p.
2. **Faulkner F.D.** Numerical methods for determining optimum ship routes. *Journal of the Institute of Navigation*, Winter 1963, Vol. 10, No. 4, Pp. 351–367.
3. **Bijlsma S.J.** A computational method for the solution of optimal control problems in ship routing. *Journal of the Institute of Navigation*, 2001, Vol. 48, No. 3, Pp. 145–154.
4. Transas sea traffic management. Available: <http://s3-eu-west-1.amazonaws.com/stm-stmvalidation/uploads/20180516214207/TRANSAS-STM.pdf> (Accessed: 27.05.2019).
5. **Karney C.F.F.** Algorithms for geodesics. *Journal Geodesy*, 2013, Vol. 87(1), Pp. 43–55.
6. **Karney C.F.F.** *Geodesics on an ellipsoid of revolution*. Technical report, SRI International, Febr. 2011.
7. **Botnev V.A., Ustinov S.M.** Methods for solving direct and inverse geodetic problems with high accuracy. *St. Petersburg State Polytechnical University Journal. Computer science. Telecommunication and Control Systems*, 2014, No. 3(198), Pp. 49–58. (rus)
8. **Press W.H., Teukolsky S.A., Vetterling W.T.** Numerical recipes in C. *The Art of Scientific Computing*, 3rd Edition, Cambridge University Press, 2007, 1235 p.
9. **Conte S.D., Carl de Boor** Elementary numerical analysis. An algorithmic approach. *SIAM-Society for Industrial and Applied Mathematics*, 2017, 480 p.
10. **Botnev V.A., Ustinov S.M.** A Method for finding the distance between a point and a line in geodesy. *St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunication and Control Systems*, 2015, No. 6(234), Pp. 33–44. (rus) DOI: 10.5862/JCSTCS.234.4
11. **Hickley P.** Great circle v rhumb line cross track distance at mid-longitude – The final answer. *The Journal of Navigation*, 2008, Vol. 61, Pp. 367–368.
12. Aviation formulary. Available: <http://www.edwilliams.org/avform.htm#XTE> (Accessed: 27.05.2019).
13. Distance from point to line on Earth. Available: <https://stackoverflow.com/questions/7803004/distance-from-point-to-line-on-earth> (Accessed: 27.05.2019).
14. Distance between lon/lat point and line segment. Available: <https://www.sitepoint.com/community/t/distance-between-long-lat-point-and-line-segment/50583> (Accessed: 27.05.2019).
15. Projecting cross-track distance on great circle. Available: <https://gis.stackexchange.com/questions/209540/projecting-cross-track-distance-on-great-circle> (Accessed: 27.05.2019).
16. How to calculate cross-track error (GPS/Core Location). Available: <https://stackoverflow.com/questions/3997410/how-to-calculate-cross-track-error-gps-core-location> (Accessed: 27.05.2019).
17. Cross track distance to great circle line and rhumb line. Available: <https://gis.stackexchange.com/questions/121117/cross-track-distance-to-great-circle-line-and-rhumb-line> (Accessed: 27.05.2019).
18. Shortest distance between a point and a geodesic. Available: <https://sourceforge.net/p/geographiclib/discussion/1026621/thread/f122cc28/> (Accessed: 27.05.2019)
19. Cross-track difference rhumb vs great circle Available: <http://www.atpforum.eu/forum/technical-subjects/-071-operational-procedures-a/1315-cross-track-difference-rhumb-vs-great-circle> (Accessed: 27.05.2019).
20. **Kells L.M.** *Plane and spherical trigonometry*. Andesite Press, 2017, 526 p.
21. **Krants P.** *Spherical trigonometry*. LKI, 2019.
22. **Helmert F.R.** Die mathematischen und physikalischen theorem der höheren geodäsie. *Die mathematischen theorem*. Leipzig, 1880, Vol. 1, 631 p.
23. **Rapp R.H.** *Geometric geodesy. Part II*. Technical report. Ohio State University, 1993, 177 p.
24. **Meyer T.H.** *Introduction to geometrical and physical geodesy*. ESRI Press, 2018, 260 p.
25. **Snyder J.P.** *Map projection – a working manual*. Professional Paper 1395, U.S. Geological Survey, 1987, 383 p.
26. **McCaw G.T.** Long lines on the Earth. *Empire Survey Review*, 1932, No. 1 (6), Pp. 259–263.
27. **Brent R.P.** *Algorithms for minimization without derivatives*. Englewood Cliffs, NJ: Prentice-Hall, 1973, 195 p.

28. **Mazurov B.T.** Linear distortions in the transition from Krasovsky's ellipsoid to the secant ellipsoid. *Geodesy and Cartography*, 2018, Vol. 79, No 1, Pp. 8–13 (rus)

29. **Vinogradov A.V., Mazurov B.T.** Prospects for the use of special geodetic projections and local coordinate systems. *Journal of Siberian State*

University of Geosystems and Technologies, 2017, Vol. 22, No 1, Pp. 18–29. (rus)

30. **Sokolov S.V.** Analytical models of spatial trajectories for solving navigation problems. *Applied Mathematics and Mechanics*, 2015, Vol. 79, No. 1, Pp. 24–30. (rus)

Received 03.04.2019.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

БОТНЕВ Виктор Александрович

BOTNEV Victor A.

E-mail: botnev.victor@mail.ru

УСТИНОВ Сергей Михайлович

USTINOV Sergey M.

E-mail: usm50@yandex.ru

EExPolytech-2019

2019 IEEE INTERNATIONAL CONFERENCE ON ELECTRICAL ENGINEERING AND PHOTONICS
OCTOBER 17-18, 2019 SAINT PETERSBURG, RUSSIA



CALL FOR PAPERS

Topics

The main topics to be discussed at EExPolytech-2019 on Electrical Engineering and Photonics are the following:

- Circuits & Systems for Telecommunications
- Algorithms & Signal Processing
- Materials & Nanoscience
- Photonics

Important dates

Full Paper Submission date: 02 September 2019
Notification of acceptance date: 25 September 2019
Final Paper Submission date: 05 October 2019

Papers

The papers should be written in [IEEE/CAS A4 format](#) (double column including paper title, authors' names and affiliation, short abstract). They should not exceed 4 pages (including figures and references). Papers written in poor English and not in accordance with the requirements will be rejected. At least one author of each accepted paper must register and present the paper in order to have the paper included into the program and Proceedings.

All papers should be sent to eexpolytech@spbstu.ru. The accepted and presented papers will be indexed in **IEEE Xplore**.

Conference Fee

The conference fee is 200 €.

Russian citizens pay in rubles. After you receive confirmation that your work is accepted you will be informed about the exact sum and payment procedure. Participants from partner universities have reduced registration fee determined by the Organizing Committee.

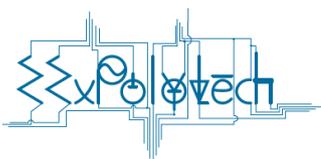
Venue

The conference will be held in Peter the Great Polytechnic University. The university is situated in Russia, Saint-Petersburg, st. Polytechnicheskaya, 29. Saint-Petersburg is a Russian port city on the Baltic Sea and one of the world's most beautiful cities.

Contacts

Email: eexpolytech@spbstu.ru
website: eexpolytech.spbstu.ru

The conference is organized by the
— Institute of Physics, Nanotechnology and Telecommunications of Peter the Great St.Petersburg Polytechnic University, Saint-Petersburg, Russia
— IEEE Russia NW Section



НАУЧНОЕ ИЗДАНИЕ
«НАУЧНО-ТЕХНИЧЕСКИЕ ВЕДОМОСТИ
САНКТ-ПЕТЕРБУРГСКОГО
ГОСУДАРСТВЕННОГО ПОЛИТЕХНИЧЕСКОГО УНИВЕРСИТЕТА.
ИНФОРМАТИКА. ТЕЛЕКОММУНИКАЦИИ. УПРАВЛЕНИЕ»

«ST. PETERSBURG STATE POLYTECHNICAL UNIVERSITY JOURNAL.
COMPUTER SCIENCE. TELECOMMUNICATIONS AND CONTROL SYSTEMS»

Том 12, № 2, 2019

Учредитель – Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Санкт-Петербургский государственный политехнический университет»

Журнал зарегистрирован Федеральной службой по надзору в сфере информационных технологий и массовых коммуникаций (Роскомнадзор).
Свидетельство о регистрации ПИ № ФС77-51457 от 19.10.2012 г.

Редакция журнала

д-р техн. наук, профессор *А.С. Коротков* – главный редактор
Е.А. Калинина – литературный редактор, корректор
Г.А. Пышкина – ответственный секретарь, выпускающий редактор

Телефон редакции (812)552-62-16, 297-18-21

E-mail: infocom@spbstu.ru

Компьютерная верстка *Е.А. Корнукова*

Перевод на английский язык *А.С. Колгатина*

Лицензия ЛР № 020593 от 07.08.97

Подписано в печать 28.06.2019. Формат 60×84 1/8. Бум. тип. № 1."
Печать офсетная. Усл. печ. л. 9,76. Уч.-изд. л. 9,76. Тираж 1000. Заказ

Санкт-Петербургский политехнический университет Петра Великого
Издательство Политехнического университета
член Издательско-полиграфической ассоциации университетов России
Адрес университета и издательства: 195251, Санкт-Петербург, ул. Политехническая, д. 29.

УСЛОВИЯ ПУБЛИКАЦИИ СТАТЕЙ

в журнале «Научно-технические ведомости Санкт-Петербургского государственного политехнического университета. Информатика. Телекоммуникации. Управление»

1. ОБЩИЕ ПОЛОЖЕНИЯ

Журнал «Научно-технические ведомости Санкт-Петербургского государственного политехнического университета. Телекоммуникации. Управление» является периодическим печатным научным рецензируемым изданием. Зарегистрировано Федеральной службой по надзору в сфере информационных технологий и массовых коммуникаций (Роскомнадзор). Свидетельство о регистрации ПИ № ФС77-51457 от 19 октября 2012 г. С 2008 года выпускается в составе сериального периодического издания «Научно-технические ведомости СПбГПУ» (ISSN 1994-2354).

Издание с 2002 года входит в Перечень ведущих научных рецензируемых журналов и изданий (перечень ВАК) и принимает для печати материалы научных исследований, а также статьи для опубликования основных результатов диссертаций на соискание ученой степени доктора наук и кандидата наук по следующим основным научным направлениям: **ИНФОРМАТИКА, ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА, РАДИОТЕХНИКА И СВЯЗЬ, ЭЛЕКТРОНИКА, ИЗМЕРИТЕЛЬНАЯ ТЕХНИКА, УПРАВЛЕНИЕ В СОЦИАЛЬНЫХ И ЭКОНОМИЧЕСКИХ СИСТЕМАХ**. Научные направления журнала учитываются ВАК Минобрнауки РФ при защите докторских и кандидатских диссертаций в соответствии с Номенклатурой специальностей научных работников.

Сведения о публикациях представлены в РИНЦ, в Реферативном журнале ВИНИТИ РАН, в международной справочной системе «Ulrich's Periodical Directory».

Периодичность выхода журнала – 4 номера в год.

2. ТРЕБОВАНИЯ К ПРЕДОСТАВЛЯЕМЫМ МАТЕРИАЛАМ

2.1. Оформление материалов

1. Рекомендуемый объем статей для авторов с ученой степенью доктора наук, званием профессора, соискателей ученой степени доктора наук (докторантов) 12–20 страниц формата А-4 с учетом графических вложений. Количество графических вложений (диаграмм, графиков, рисунков, таблиц, фотографий и т. п.) не должно превышать 4.

2. Рекомендуемый объем статей для преподавателей, авторов без ученой степени, соискателей ученой степени кандидата наук – 8–15 страниц формата А-4; аспирантов – 8 страниц формата А-4 с учетом графических вложений. Количество графических вложений (диаграмм, графиков, рисунков, таблиц, фотографий и т. п.) не должно превышать 3.

3. Авторы должны придерживаться следующей обобщенной структуры статьи: вводная часть (0,5–1 стр., актуальность, существующие проблемы); основная часть (постановка и описание задачи, изложение и суть основных результатов); заключительная часть (0,5–1 стр., предложения, выводы), список литературы (оформление по ГОСТ 7.05.-2008).

4. Число авторов статьи не должно превышать трех человек.

5. Набор текста осуществляется в редакторе **MS Word**, формул – в редакторе **MathType**. Таблицы набираются в том же формате, что и основной текст.

6. Шрифт – TNR, размер шрифта основного текста – 14, интервал – 1,5; таблицы большого размера могут быть набраны 12 кеглем. Параметры страницы: поля слева – 3 см, сверху, снизу – 2,5 см, справа – 2 см, текст размещается без переносов. Абзацный отступ – 1 см.

2.2. Предоставление материалов

Вместе с материалами статьи должны быть обязательно предоставлены:

- номер УДК в соответствии с классификатором (в заголовке статьи);
- аннотация на русском и английском языках;
- ключевые слова (5–7) на русском и английском языках;
- сведения об авторах на русском и английском языках: ФИО, место работы, должность, ученое звание, ученая степень, контактные телефоны, e-mail;
- аспиранты представляют документ отдела аспирантуры, заверенный печатью;
- акт экспертизы о возможности опубликования материалов в открытой печати.

С авторами статей заключается издательский лицензионный договор.

Предоставление всех материалов осуществляется в электронном виде через личный кабинет **ЭЛЕКТРОННОЙ РЕДАКЦИИ** по адресу <http://journals.spbstu.ru>

2.3. Рассмотрение материалов

Предоставленные материалы (п. 2.2) первоначально рассматриваются редакционной коллегией и передаются для рецензирования. После одобрения материалов, согласования различных вопросов с автором (при необходимости) редакционная коллегия сообщает автору решение об опубликовании статьи. В случае отказа в публикации статьи редакция направляет автору мотивированный отказ.

При отклонении материалов из-за нарушения сроков подачи, требований по оформлению или как не отвечающих тематике журнала материалы не публикуются и не возвращаются.

Редакционная коллегия не вступает в дискуссию с авторами отклоненных материалов.

Публикация материалов аспирантов очной бюджетной формы обучения осуществляется бесплатно в соответствии с очередностью.

При поступлении в редакцию значительного количества статей их прием в очередной номер может закончиться **ДОСРОЧНО**.

Более подробную информацию можно получить:

на сайте журнала <http://ntv.spbstu.ru>

по телефону редакции +7(812) 552-62-16 с 10⁰⁰ до 18⁰⁰ Галина Александровна

или по e-mail: infocom@spbstu.ru