



НАУЧНО-ТЕХНИЧЕСКИЕ ВЕДОМОСТИ

САНКТ-ПЕТЕРБУРГСКОГО ГОСУДАРСТВЕННОГО
ПОЛИТЕХНИЧЕСКОГО УНИВЕРСИТЕТА

Информатика. Телекоммуникации.
Управление

Том 11, № 4
2018

Издательство Политехнического университета
Санкт-Петербург
2018

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:

Ицыйсон 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;

Шкодырев 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 ВИНИТИ 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, 2018



ST. PETERSBURG STATE POLYTECHNICAL UNIVERSITY JOURNAL

Computer Science.
Telecommunications and Control Systems

Vol. 11, No. 4
2018

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

Заборовский В.С., Джан Л., Скиба В.Ю., Стрекалов С.В. Цифровая информационно-логистическая платформа оперативного управления внешнеторговой деятельностью поставщиков высокотехнологичной продукции	7
Попов С.Г., Лисенкова А.А. Алгоритмы динамической генерации MDX-запросов к многомерным OLAP-кубам	21

Телекоммуникационные системы и компьютерные сети

Сорокин А.А., Чан Куок Т. Адаптивное нечеткое управление заполнением буфера сетевого устройства	36
--	----

Программное обеспечение вычислительных, телекоммуникационных и управляющих систем

Верещагин К.А., Белых И.Н. Решение задачи построения ультразвуковых изображений методом волновой инверсии с использованием алгоритмов распараллеливания вычислений	49
Шевцов Ю.Д., Дудник Л.Н., Чернуха Ю.В., Фадеев Е.Д., Никифоров А.М. Алгоритм диагностики технического состояния двигателя по параметрам частотных характеристик масляного фильтра	63
Тимофеев Д.А., Маслов М.Ю. Архитектура системы сбора и обработки данных для повышения производительности труда разработчиков программного обеспечения	71
Попов С.Г., Фридман В.С. Обзор методов динамического распределения данных в распределенных системах управления базами данных	82

Аппаратное обеспечение вычислительных, телекоммуникационных и управляющих систем

Антонов А.П., Филиппов А.С., Киселев И.О. Разработка аппаратно-реконфигурируемого вычислителя с поддержкой стандарта OpenCL	108
--	-----

Системный анализ и управление

Байдина Т.А., Бурдаков С.Ф., Шагниева О.Б. Управление импульсным сглаживанием фрикционных автоколебаний при контактном взаимодействии робота с обрабатываемой поверхностью	119
---	-----

Интеллектуальные системы и технологии

Никитин К.В. Исследование и анализ динамики импульсных рекуррентных нейронных сетей в контексте задачи распознавания образов	130
Станкевич Л.А., Аманбаева С.С., Самочадин А.В. Оценка уровня умственной работоспособности учащихся на основе анализа сигналов ЭЭГ	151
Тимофеев Д.А., Самочадин А.В. Извлечение моделей бизнес-процессов из обучающих материалов	162

Contents

Information Technologies

- Zaborovsky V.S., Zhang Lei, Skiba V.Yu., Strekalov S.V.** Digital information and logistic platform for operational management of foreign trade activities of high-tech product suppliers 7
- Popov S.G., Lisenkova A.A.** Algorithms for MDX-query generation in multidimensional OLAP-cubes 21

Telecommunication Systems and Computer Networks

- Sorokin A.A., Tran Quoc T.** Adaptive fuzzy control for buffer loading regulation of network node 36

Software of Computer, Telecommunications and Control Systems

- Vereshchagin K.A., Belykh I.N.** Ultrasound image reconstruction based on waveform inversion method using parallel calculation algorithms 49
- Shevtsov Yu.D., Dudnik L.N., Chernukha Yu.V., Fadeev E.D., Nikiforov A.M.** Algorithm for diagnosing engine health by parameters of frequency characteristics of oil filter 63
- Timofeev D.A., Maslov M.Yu.** Architecture of data acquisition and processing system for improving productivity of software developers 71
- Popov S.G., Fridman V.S.** Review of methods for dynamic distribution of data in distributed database management systems 82

Hardware of Computer, Telecommunications and Control Systems

- Antonov A.P., Filippov A.S., Kiselev I.O.** Design of reconfigurable computer supporting OpenCL standard 108

System Analysis and Control

- Baidina T.A., Burdakov S.F., Shagnie O.B.** Control of friction self-oscillations pulse smoothing in contact interaction of a robot with the working surface 119

Intellectual Systems and Technologies

- Nikitin K.V.** Analysis of spiking recurrent neural networks dynamics in context of pattern recognition problem 130
- Stankevich L.A., Amanbaeva S.S., Samochadin A.V.** Evaluation of students' mental performance level based on EEG signal analysis 151
- Timofeev D.A., Samochadin A.V.** Process extraction from educational texts 162

DOI: 10.18721/JCSTCS.11401
УДК 65.428

ЦИФРОВАЯ ИНФОРМАЦИОННО-ЛОГИСТИЧЕСКАЯ ПЛАТФОРМА ОПЕРАТИВНОГО УПРАВЛЕНИЯ ВНЕШНЕТОРГОВОЙ ДЕЯТЕЛЬНОСТЬЮ ПОСТАВЩИКОВ ВЫСОКОТЕХНОЛОГИЧНОЙ ПРОДУКЦИИ

В.С. Заборовский¹, Лей Джан², В.Ю. Скиба³, С.В. Стрекалов³

¹Санкт-Петербургский политехнический университет,
Санкт-Петербург, Российская Федерация;

²Восточно-китайский педагогический университет,
КНР, г. Шанхай;

³Московский государственный технический университет имени Н.Э. Баумана,
Москва, Российская Федерация

Предложена новая архитектура и организационно-функциональная модель цифровой информационно-логистической платформы индустриальной экосистемы, обеспечивающей взаимодействие предприятий и органов государственного контроля внешнеэкономической деятельности. С учетом приоритетов развития экономики России рассмотрены возможности использования новых бизнес-моделей, информационных и логистических технологий, расширяющих перспективы государственно-частного партнерства, как эффективного инструмента повышения доли высокотехнологичной продукции в структуре российско-китайских внешнеэкономических связей. Показано, что применение современных информационно-технологических подходов и решений с использованием смарт-контрактов на базе распределенных реестров позволяет сформировать единое информационное пространство защищенных транзакций между предприятиями-производителями, компаниями-экспортерами, заказчиками, финансовыми институтами и государственными органами контроля, что обеспечивает в автоматическом режиме мониторинг внешнеэкономической деятельности как отдельных предприятий, так и национальной экономики в целом.

Ключевые слова: информационно-логистическая платформа, цифровая экономика, индустриальная экосистема, внешнеэкономическая деятельность, информационное взаимодействие, государственно-частное партнерство, внешнеэкономическая деятельность.

Ссылка при цитировании: Заборовский В.С., Лей Джан, Скиба В.Ю., Стрекалов С.В. Цифровая информационно-логистическая платформа оперативного управления внешнеэкономической деятельностью поставщиков высокотехнологичной продукции // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2018. Т. 11. № 4. С. 7–20. DOI: 10.18721/JCSTCS.11401

DIGITAL INFORMATION AND LOGISTIC PLATFORM FOR OPERATIONAL MANAGEMENT OF FOREIGN TRADE ACTIVITIES OF HIGH-TECH PRODUCT SUPPLIERS

V.S. Zaborovsky¹, Lei Zhang², V.Yu. Skiba³, S.V. Strekalov³

¹Peter the Great St. Petersburg Polytechnic University,
St. Petersburg, Russian Federation;

²East China Normal University,
China, Shanghai;

³Bauman State Technical University,
Moscow, Russian Federation

The article proposes a new architecture and functional model of the digital information and logistics operations within modern industrial ecosystem, which ensures the interaction of enterprises and custom control of foreign trade activities. Taking into account the

priorities of the Russian economy digital transformation, the possibilities of using new business models, new information and logistics technologies that expand the prospects of public-private partnership as an effective tool to increase the share of high-tech products in the structure of Russian-Chinese foreign trade relations are considered. It is shown that the use of intelligent applications, security approaches and smart contracts on the basis of distributed registers allows to create a robust information space of trade transactions between manufacturing enterprises, exporting companies, customers, financial institutions and state control bodies, which provides automatic monitoring of all operations as individual enterprises and the national economy as a whole.

Keywords: information and logistics platform, digital economy, industrial ecosystem, foreign trade relations, information interaction, public-private partnership.

Citation: Zaborovsky V.S., Lei Zhang, Skiba V.Yu., Strekalov S.V. Digital information and logistic platform for operational management of foreign trade activities of high-tech product suppliers. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2018, Vol. 11, No. 4, Pp. 7–20. DOI: 10.18721/JCSTCS.11401

Всё должно быть изложено так просто,
как только возможно, но не проще.

Альберт Эйнштейн

Введение

Скорость изменения технологий постоянно возрастает, что приводит к необходимости освоения новых навыков и знаний, позволяющих учитывать в используемых экономических моделях такие факторы, как нестабильность, неопределенность, сложность и неоднозначность. Концепция развития экономики в условиях VUCA (акроним от слов *volatility*, *uncertainty*, *complexity* и *ambiguity*) приобретает все большую популярность [1], что оказывает непосредственное влияние на процессы управления внешнеторговой деятельностью (ВТД). В этих условиях ВТД приобретает новую специфику, которую мы предлагаем рассматривать с позиций сокращения временных затрат на информационно-логистические операции за счет устранения «фальшивых разногласий» с помощью создания сервисов для реализации торговых операций в формате смарт-контрактов; формирования индикаторных показателей, непосредственно характеризующих интегративные аспекты торговых транзакций в рамках ВТД.

Предлагаемое нами решение основано на использовании схемы-модели CUCA — communication (коммуникации в аспекте реализации бизнес-процессов), understanding (понимание цели в аспекте знания законов и регламентов), clarity (ясность в аспекте текущего состояния ресурсов), agility (динамичность в аспекте полномочий принятия решений).

Анализ особенностей ВТД проводится для предприятий и организаций, которые производят продукцию с высокой долей добавленной стоимости, используют сквозные технологии защиты интеллектуальной собственности на производимую продукцию. Такие организации заинтересованы в минимизации финансовых, временных и логистических издержек, связанных с ведением ВТД, что требует оперативного учета и адаптации к изменяющимся соотношениям цен на производимые товары на внутреннем и международных рынках. При этом преимущество в абсолютных товарных ценах, включая затраты на транспортировку, не является достаточным условием взаимовыгодного долгосрочного внешнеторгового обмена. В условиях VUCA обеспечение конкурентных преимуществ за счет включения в процессы ВТД оборота прав на интеллектуальную собственность становится необходимым условием роста экспортного потенциала национальной экономики.

С учетом вышесказанного, в статье анализируются требования к цифровой информационно-логистической платформе (ЦИЛП), обеспечивающей развитие ВТД предприятий, ориентированных на внешние рынки. ЦИЛП обеспечивает согласованное взаимодействие предприятий и контролирующих органов в рамках бизнес-моделей класса B2B, B2G и G2G, с гарантированным выполнением требований по сертификации, транспортному сопровождению и таможенному регулированию ВТД. При создании ЦИЛП в условиях VUCA появляется возможность контроли-

ровать все регламентные операции: от заключения контракта до успешного закрытия сделки, последующего гарантийного обслуживания и соблюдения прав на используемую интеллектуальную собственность, обработки и хранения в распределенных реестрах в формате смарт-контрактов. Автоматическое исполнение смарт-контрактов снижает риски логистических ошибок, унифицирует процесс документооборота, визуализирует в режиме реального времени процессы перемещения товаров и статус таможенных документов.

Схема-модель CUCA рассматривается на примере развития российско-китайских экономических связей, с точки зрения увеличения сегмента товаров с высокой долей добавленной стоимости в условиях постоянных технологических изменений и поддержки бизнес-моделей ВТД. CUCA позволяет радикально повысить надежность и безопасность цепочки внешнеторговых операций, уровень автоматизации систем учета и контроля, сократить вре-

менные и финансовые издержки предприятий-экспортеров, а также внедрить принцип однократности предоставления данных (принцип «единого окна») [2, 3].

ВТД в эпоху цифровой экономики

Интерес к использованию цифровых технологий в сфере международной торговли постоянно растет. Для минимизации последствий влияния т. н. процессов VUCA и повышения эффективности управления на основе обработки, хранения и транспортировки данных о товарах, предлагается использовать схему-модель взаимодействия CUCA. Данная схема-модель состоит в согласовании операций: передачи данных при реализации бизнес-процессов, определении цели в аспекте знания законов и регламентов ВТД, наблюдаемости текущего состояния ресурсов и контролируемого распределения полномочий принятия решений (рис. 1).

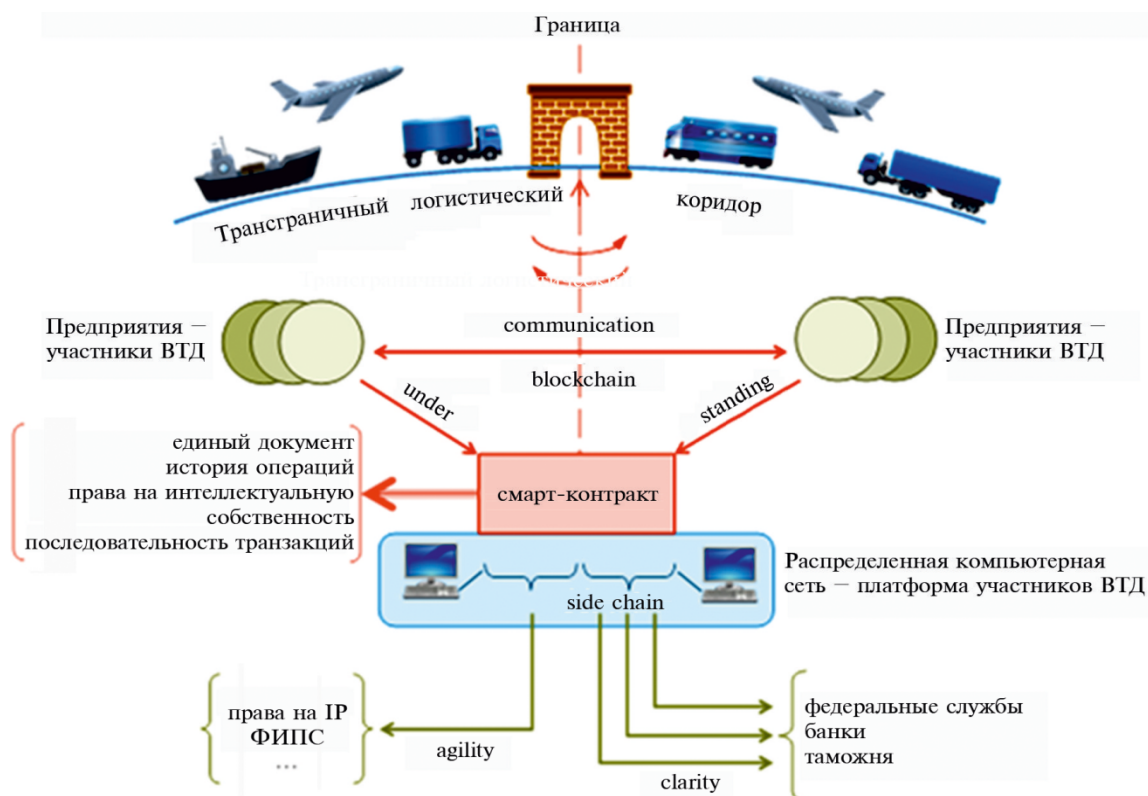


Рис. 1. Схема-модель CUCA

Fig. 1. CUCA model diagram

Использование CUSA значительно снижает потенциальную возможность мошенничества, нарушения регламентов ответственности, правил сертификации и использования ложной информации. Благодаря одноранговой архитектуре базы данных CUSA, информационное пространство ВТД наделяется новыми функциями, что позволяет экспортеру и импортеру напрямую обмениваться юридически значимыми документами, организованными в форме криптографически защищенной цепочки блоков (blockchain — блокчейн). Корректность обновления информации подтверждается всеми узлами сети блокчейн, а специально организованные цепочки записей (sidechain) позволяют органам таможенного и валютного контроля автоматически получать необходимую информацию, что сокращает административные издержки и ускоряет все процессы ВТД.

Учитывая, что распределенные реестры — это децентрализованная сетевая технология хранения и обработки данных, включающая мета-информацию о предыдущих операциях, новая операция добавления с использованием реестров выполняется с учетом контекста предыдущих операций. Это обеспечивает верификацию новых операций в будущем.

Каждый результат успешной операции называется «умным контрактом», хранится в распределенном реестре и многократно дублируется в системе, что обеспечивает автоматизированный контроль его корректности, радикально снижающий число нарушений при подготовке документации и таможенном оформлении, снижая логистические затраты за счет:

- обработки всех ВТД транзакций в общем криптографически защищенном информационном пространстве предприятий-производителей, лицензированных предприятий-экспортеров, заказчиков высокотехнологичной продукции, финансовых организаций и государственных органов таможенного контроля;
- создания распределенной трансграничной цифровой инфраструктуры оперативного управления и планирования внешнеторговой деятельности;

- интеграции производственных и логистических возможностей проведения экспортно-импортных операций, характерных для предприятий, занятых в сфере высокотехнологичного производства.

Рассмотрим, как с помощью современных информационно-вычислительных технологий реализовать полный набор регламентных требований, включая оперативный контроль различных факторов неопределенности, влияющих на скорость перемещения товаров через государственные границы.

Анализ структуры товарооборота

Согласно данным Главного таможенного управления КНР, товарооборот России с Китаем в январе–сентябре 2018 года увеличился на 25,7 % (против 22,4 % за аналогичный период 2017 до 77 148,5 млн долл. США, в том числе экспорт из России в КНР вырос до 41 947,6 млн долл. (+39,2 % против +28,5 % годом ранее), импорт из КНР в Россию — до 35 200,8 млн долл. (+12,7 % против +17,0 %) (рис. 2). Активное торговое сальдо составило 6 746,8 млн долл. против пассивного сальдо в 1 379,54 млн долл. по итогам первого полугодия 2017 года. Профицит торгового баланса достиг 8,74 % от совокупного двухстороннего товарооборота (против дефицита 2,24 % в январе–сентябре 2017).

По итогам девяти месяцев 2018 года Россия в рейтинге 20 основных торговых партнеров Китая улучшила свои позиции, заняв 11-е место (в 2017 — 12-е место). Планируется, что некоторое снижение доли поставок сырьевых товаров может произойти в результате реализации межгосударственных программ по созданию в России китайских перерабатывающих производств, ориентированных на экспорт готовой продукции в Китай. До тех пор, пока минеральные (в том числе углеводороды, цветные металлы) и природные ресурсы (древесина, морепродукты) доминируют в российском экспорте, задача расширения объемов экспорта отечественной наукоемкой продукции будет являться актуальной.

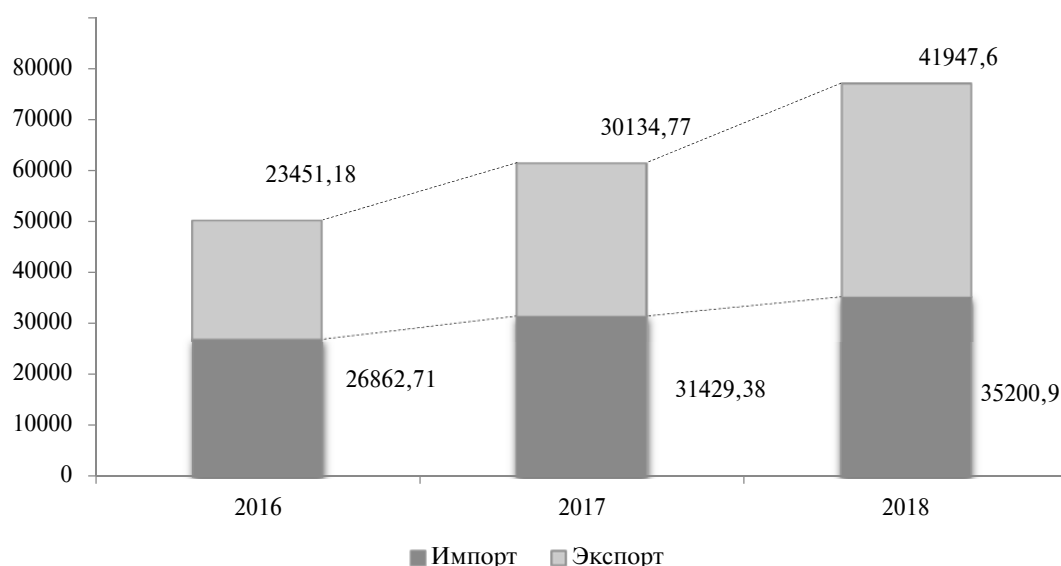


Рис. 2. Динамика товарооборота КНР–Россия, январь–сентябрь 2016–2018 гг., млн долл. США

Fig. 2. Dynamics of the trade turnover of the PRC–Russia, January–September 2016–2018, million dollars

В настоящее время для изменения сложившейся диспропорции в структуре отечественного экспорта имеются хорошие возможности. Учитывая, что наступающий этап индустриального развития характеризуется повсеместным применением в различных секторах экономики, включая финансы, здравоохранение, логистику, прикладные науки и образование, интеллектуальных человеко-машинных систем и облачных технологий, экспорт продукции, созданной на базе «умных» технологий является приоритетным товаром на глобальном рынке Hi-Tech продукции. Поэтому на изменение структуры экспорта может активно повлиять капитализация результатов интеллектуальной деятельности отечественных ученых и инженеров, что, в свою очередь, потребует развития государственной системы контроля за соблюдением лицензионных прав на объекты интеллектуальной собственности при реализации товаров в рамках контрактов на проведение ВТД. Создание такой системы позволит:

- в рамках концепции «единого окна» и технологии распределенных реестров существенно образом повышать скорость перемещения материальных и нематери-

альных активов через государственные границы без снижения уровня таможенного контроля;

- привлекать к процессу управления и планирования ВТД как представителей предприятий-экспортеров, так и лиц, осуществляющих профессиональную деятельность в сферах контроля прав интеллектуальной собственности и таможенного дела;
- развивать возможности организационно-правовых и информационно-технологических форм государственно-частного партнерства (ГЧП) [4, 5] с учетом интересов участников внешнеэкономической деятельности и органов государственного финансового и таможенного контроля.

Таким образом, товары с большой долей добавленной интеллектуальной собственности, такие как сервисы и средства IoT в медицине, логистике и торговле, суперкомпьютерное программное обеспечение, приборы на базе квантовых технологий, средства интеллектуального анализа больших данных, компьютерные системы предсказательного моделирования материалов, лекарств, климата и др., должны стать приоритетным направлением развития экспортных возможностей отечественных предприятий.



Рис. 3. Информационно-логистическая модель

Fig. 3. Information and logistic model

Ниже анализируются возможности развития ГЧП, основанные на использовании сквозных технологий, стимулирующих трансформацию экономических отношений между различными субъектами ВТД, и бизнес-моделей, реализуемых с использованием ресурсов цифровой информационно-логистической платформы.

Бизнес-модели цифровой трансформации ВТД

В рассматриваемом контексте суть цифровой трансформации ВТД сводится к использованию возможностей сквозных информационно-вычислительных систем и сетей класса IoT для стимулирования деятельности промышленных предприятий к кардинальному повышению объемов экспорта высокотехнологичной продукции, включая изменения бизнес-моделей взаимоотношений с партнерами и конкурентами за счет внедрения смарт-контрактов.

При этом следует учитывать, что согласно докладу Всемирной торговой организации, устранение барьеров в области международных перевозок товаров увеличит мировой ВВП на 5 %, а общий объем пере-

возок — на 15 %.* Информационно-логистическая модель внешнеторговой деятельности, использующая технологию контроля транзакций с помощью распределенных реестров и смарт-контрактов, представлена на рис. 3.

Важными задачами применения цифровых технологий в процессах контроля и управления ВТД являются [6]:

- удешевление логистики, чтобы ее доля в стоимости товара была минимальной;
- повышение уровня доверия между участниками сделки и органами таможенного и иного государственного контроля;
- увеличение объемов добавленной интеллектуальной собственности в номенклатуре экспортируемых товаров и сервисов;
- учет прав интеллектуальной собственности в заключенных лицензионных соглашениях.

Для этого все регламентные финансово-логистические операции, вплоть до успешного закрытия сделки и последующего гарантийного обслуживания, предлагается записывать и хранить в распределенных защищенных реестрах. В этом случае основой

* <https://blockchain3.ru/blokchejn/blokchejn-v-logistike/>

ВТД становятся смарт-контракты (заявка, описание, аккредитив и т. д.), которые автоматически гарантируют нормативно корректное урегулирование требований в соответствии с принятой компанией и утвержденной таможенными органами моделью внешнеторговой деятельности (смарт-контракт ВТД).

Следует заметить, что смарт-контракт реализует алгоритм, выполняемый в локальной или облачной вычислительной среде, который описывает согласованную поставщиком и заказчиком последовательность выполнения контрактных обязательств. При создании информационной платформы осуществления российско-китайской внешнеторговой деятельности надо учитывать, что обрабатываемые данные, используемые, например, для сертификации товаров в соответствии с национальными стандартами качества, находятся в «постоянном движении», траектория которых может быть наглядно отображена на многослойной телематической карте [7].

В современных условиях формирование реестра «обязательства плюс регламент», который характеризует временную последовательность выполнения контрактных обязательств, с помощью традиционных деклараций на бумажных носителях существенно снижает скорость проведения торговых операций и одновременно повышает количество неточных сведений, связанных с влиянием «человеческого фактора». Именно поэтому в основу рассмотренной информационно-логистической модели внешнеторговых операций положены цифровые технологии, позволяющие создать сетевую инфраструктуру нового класса для осуществления внешнеторговой деятельности или «интернет» экономически значимых ценностей. Такая инфраструктура может быть создана на основе интеллектуальной цифровой информационно-логистической платформы (ЦИЛП), осуществляющей и контролирующей материальные и информационные потоки, формируемые при реализации экспортных контрактных обязательств и регламентов таможенного контроля.

Организационно-функциональная модель ЦИЛП

Как было отмечено выше, бизнес-процессы, происходящие при реализации внешнеторговой деятельности, могут использоваться для проектирования смарт-контрактов на нормативно-алгоритмическом уровне регламентирующих последовательность как подготовительных, так и таможенно-логистических операций [8, 9]. При этом к подготовительным операциям относятся процессы продажи и закупки готовой продукции или материалов, оформление соответствующих внешнеторговых контрактов и осуществление банковских транзакций. К таможенно-логистическим процессам относятся процессы получения разрешительных документов (сертификатов соответствия и пр.), прохождения таможенных формальностей, транспортная обработка грузов на всем пути следования или передача товаров с помощью информационных носителей через компьютерные сети.

В настоящее время именно на участке использования национальной транспортной инфраструктуры и проведения таможенно-логистических операций наблюдаются самые крупные временные издержки [10, 11], которые напрямую связаны с финансовыми потерями предприятий-экспортеров продукции. Поэтому при разработке архитектуры ЦИЛП основное внимание уделяется управлению процессами между участниками внешнеторговой деятельности (рис. 4):

- формату B2B (Business to Business) – процессам взаимодействия промышленных предприятий и логистических операторов, страховых компаний и других участников логистической цепи, включая сервисы отслеживания грузов в реальном времени, базы данных перевозчиков и экспедиторских компаний, брокерские и консультационные услуги, включая сервис «доверенной третьей стороны», обеспечивающий обмен электронными документами, имеющими юридическую силу;



Рис. 4. Управление процессами ЦИЛП

Fig. 4. Control of DILP processes

- формату B2G (Business to Government) — процессам подачи промышленными предприятиями таможенных деклараций, предварительной информации, статистических форм учета и необходимых сведений в электронном виде. Функциональные возможности платформы должны повысить скорость прохождения процедур таможенного оформления и контроля для добросовестных участников ВЭД [11], а при выполнении определенного набора критериев на протяжении определенного периода времени, компания попадет в список благонадежных, и процесс таможенного оформления возможно будет осуществляться автоматически информационными системами таможенных органов без участия должностных лиц;
- формату G2G (Government to Government) — процессам межведомственного взаимодействия ФОИВ (Федеральная таможенная служба, Россельхознадзор, Росаккредитация, Роспотребнадзор, Феде-

ральная налоговая служба и др.), связанным с формированием запросов в информационные системы взаимодействующих органов. При этом все информационные системы федеральных и иных контролирующих органов, используемые для предоставления государственных услуг, должны быть зарегистрированы и подключены к ЦИЛП непосредственно или с использованием системы межведомственного электронного взаимодействия, созданной в Российской Федерации.

В ЦИЛП распределенный реестр фактически выполняет функцию информационного интегратора, в который поступают сведения как от участников экспортных сделок, так и от оборудования, формирующего транспортную инфраструктуру класса IoT, используемую логистическим оператором или провайдером сервиса IaaS. В качестве источников обрабатываемых в процессе таможенного сопровождения смарт-контракта

данных могут использоваться RFID-датчики, цифровые видеокамеры, датчики температуры, влажности, GPS-навигаторы и другие телематические устройства, передающие информацию о текущем состоянии, маршруте следования, местоположении на складе или других логистических атрибутов перемещаемого товара.

При этом технология распределенных реестров на алгоритмическом уровне закрепляет перечень требований к товару, его перевозке и хранению, который обязателен для выполнения поставщиком и компанией-логистическим оператором. В дальнейшем смарт-контракт автоматически подготавливает финансовые потоки и формирует нейтральный счет, куда получатель товара перечисляет оплату по проведенной товарной сделке. При этом денежные средства находятся на данном счете, пока товар не пересечет таможенную границу и не станет объектом контроля смарт-контракта.

Организационная структура ЦИЛП базируется на принципах государственно-частного партнерства, обеспечивающих тесное взаимодействие заинтересованных государственных структур и представителей бизнес-сообщества. Для реализации проекта создания ЦИЛП потребуется определить уполномоченный орган, который будет отвечать за выполнение регламентных требований и общую координацию работы информационных систем государственных органов. На подготовительном этапе проекта необходимо провести оценку необходимых инвестиций, постоянных затрат, планируемых выгод для государственных структур и участников ВЭД. Для привлечения инвестиционных активов со стороны частного сектора потребуется также разработка детального бизнес-плана с прогнозными сценариями доходности инвестиций в проект. Важно подчеркнуть, что ключевым фактором успеха согласно международным рекомендациям является наличие государственной стратегии, ориентированной на поддержку процессов цифровой трансформации национальной экономики.

Можно выделить два типа потенциальных пользователей ЦИЛП. Это предприя-

тия, уже ведущие внешнеторговую деятельность, и предприятия, только собирающиеся начать экспортировать свою продукцию на внешние рынки или импортировать сырье и материалы.

В первом случае для начала работы с ЦИЛП предприятию необходимо произвести оценку планируемых объемов внешне-торговой деятельности и по результатам оценить требуемый объем поддержки таможенно-логистических и импортно-экспортных операций. В случае низких значений индикативных показателей заключается контракт с ЦИЛП на полный доступ ко всем сегментам платформы. При высоких показателях, по результатам оценки, заключается соглашение с доступом только к сегменту B2G, позволяющему оптимизировать взаимодействие предприятия и государственных органов контроля. После заключения соглашения предприятием вносятся соответствующие изменения в регламент сопровождения внешнеторговых сделок.

Анализ структуры ЦИЛП показывает, что используемая в ней информация, являясь носителем событий и фактов, образует распределенный реестр или упорядоченный по определенным правилам набор данных, которые наделены атрибутами, характеризующими как права, так и факт их владения. Эти атрибуты обеспечивают «высокую точность» учета и управления данными, что открывает принципиально новые возможности применения распределенных реестров для осуществления и контроля внешнеторговых транзакций. Повышение точности учета данных снижает неопределенность в процессе планирования и экономические риски логистических транзакций, что создает предпосылки к радикальному изменению бизнес-моделей, основанных на исполнении взаимных обязательств. Это особенно важно при осуществлении внешнеторговой деятельности, т. к. позволяет автоматизировать процессы контроля за исполнением обязательств, анализируя на основе согласованных алгоритмов реестры финансовых ресурсов поставщиков, клиентов, интеллектуальной собственности и применяемых стандартов.

Принимая во внимание, что реестры-контракты или смарт-контракты могут быть полными (учитывают любые возможные ситуации) или открытыми (могут быть дополнены или изменены в зависимости от ситуации), оперативное управление экономическими процессами в рамках внешнеторговой деятельности может рассматриваться как коллективная и не авторитарная технология, позволяющая моментально и публично подтвердить точность и подлинность предоставляемых данных.

Методика оценки индикаторных показателей

Набор ключевых групп показателей, учитываемых в рамках ЦИЛП, дает наиболее полное представление об уровне организации внешнеторговой деятельности промышленного предприятия. Отличительной особенностью рассматриваемой методики является система картирования показателей и фокусировка на анализе таможенно-логистических процессов при реализации внешнеторговой деятельности предприятия.

Результат оценки представляется в виде графика, на котором показаны значения индикаторных показателей конкретного

предприятия и целевые значения, отражающие нормативные ограничения. Образец графического представления результатов оценки представлен на рис. 5.

Первая группа — показатели, направленные на аудит затрат на организацию таможенно-логистических процессов. Показатели определяются отношением фактических значений конкретного предприятия и среднеотраслевых показателей.

1. Коэффициент стоимости оформления одной экспортной таможенной декларации ($c_э$) [6]:

$$c_э = \frac{дт}{дт_{ср}}, \quad (1)$$

где $дт$ — стоимость оформления одной экспортной таможенной декларации в рублях (при использовании услуг брокерской компании, в случае наличия штатных сотрудников-декларантов стоимость определяется как среднее время в часах на оформление одной декларации, умноженное на зарплату сотрудника в рублях за час); $дт_{ср}$ — среднерыночная стоимость оформления одной таможенной декларации в рублях.

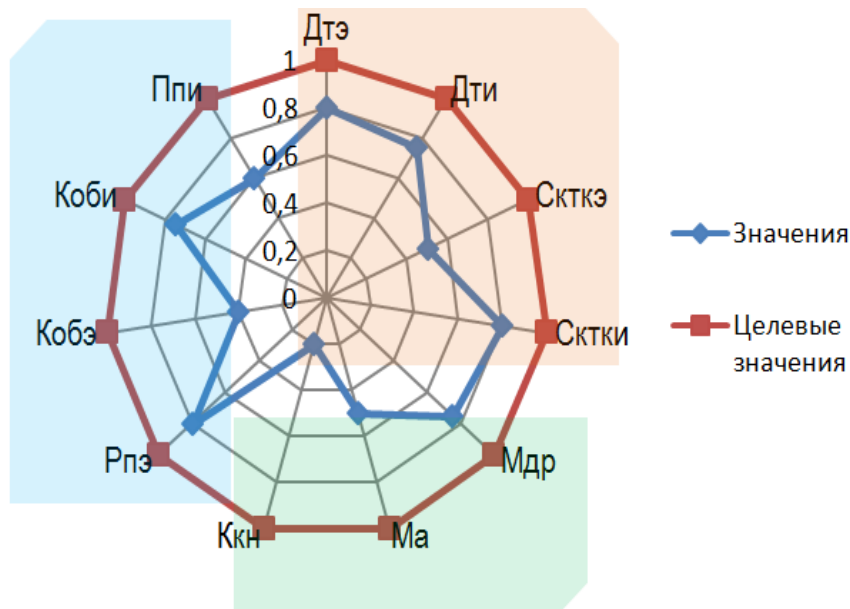


Рис. 5. Визуализация показателей ВТД предприятия с помощью радиальной диаграммы

Fig. 5. Visualization of the company's foreign trade activities of enterprises indicators using a radial chart

При осуществлении предприятием импортных операций по закупке комплектующих и материалов также рассчитывается показатель стоимости оформления импортой декларации $c_{дн}$.

2. Коэффициент стоимости экспедирования одного контейнера ($c_{кткэ}$):

$$c_{кткэ} = \frac{КТК}{КТК_{ср}}, \quad (2)$$

где $ктк$ — затраты в рублях на экспедирование одного контейнера до обозначенного таможенного поста в стране экспорта продукции предприятия; $ктк_{ср}$ — среднерыночная стоимость экспедирования контейнера до обозначенного таможенного поста в стране экспорта продукции предприятия.

При осуществлении импортных операций таким же методом рассчитывается коэффициент стоимости экспедирования контейнера по импорту $c_{ктки}$.

В случае если предприятие не использует контейнерные перевозки, аналогичным способом анализируются затраты на экспедирование других типов перевозок. Дополнительно могут быть проанализированы затраты на оплату труда сотрудников отдела внешнеэкономической деятельности, затраты на получение сертификатов на товары при осуществлении ВТД, составленные на основе анализа предложений нескольких сертифицирующих организаций, а также аналитические прогнозные оценки, полученные с помощью ЦИЛП на основе технологии обработки «больших данных».

Вторая группа — показатели маркетинговой активности.

1. Оценивается доля целевого сегмента внешнего рынка $M_{др}$, %:

$$M_{др} = \frac{V_{пп}}{V_{общ}} \times 100 \%, \quad (3)$$

где $V_{пп}$ — объем продаж продукции предприятия на внешнем рынке, у.е.; $V_{общ}$ — общий объем целевого рынка, выражаемый в приведенных единицах стоимости.

Коэффициент конкурентоспособности продукции $K_{кн}$ определяется методом аддитивной свертки потребительских и эконо-

мических показателей, которым присваиваются весовые коэффициенты методом экспертных оценок [6, 7].

2. Рассчитывается коэффициент маркетинговой активности M_a , %:

$$M_a = \frac{N_{п}}{N_{об}} \times 100 \%, \quad (4)$$

где $N_{п}$ — количество профильных мероприятий (выставок), посещенных предприятием на целевом рынке; $N_{об}$ — общее количество профильных мероприятий на целевом.

Дополнительно могут быть проанализированы годовые затраты на маркетинг по продвижению продукта на целевом внешнем рынке, а также исследована динамика экспортных поставок продукции за три года в соотношении с затратами на маркетинг.

Третья группа — показатели финансовой эффективности предприятия.

1. Расчет дохода от продаж по экспорту на единицу продукции $P_{пэ}$, %:

$$P_{пэ} = \frac{ЧП_э}{V_э} \times 100 \%, \quad (5)$$

где $ЧП_э$ — чистый доход от реализации товара за границу, руб.; $V_э$ — количество реализуемой продукции в денежном выражении.

2. Определение коэффициента оборачиваемости активов $K_{обэ}$:

$$K_{обэ} = \frac{B_э}{СОК}, \quad (6)$$

где $B_э$ — выручка от экспорта (импорта), руб.; $СОК$ — собственный оборотный капитал.

3. Показатель экономического эффекта импорта рассчитывается с учетом специфики продукции предприятия. Для продукции, выступающей в качестве сырья или материалов, эффективность импорта $K_{эим}$ оценивается в сравнении с затратами по закупке аналогичной продукции на внутреннем рынке; для импортных товаров, направленных на дальнейшую перепродажу, критерием эффективности служит наличие прибыли от продажи $P_{пи}$.

Практическое применение методики способствует повышению эффективности

внешнеторговой деятельности предприятия с учетом следующего:

- графическое представление динамики показателей удобно для мониторинга;
- комплекс показателей позволяет контролировать их текущее состояние в режиме реального времени;
- комплекс показателей может использоваться для создания программного продукта, позволяющего в автоматическом или автоматизированном виде проводить оценку внешнеторговой деятельности предприятий.

Разработанный набор индикаторных показателей позволяет повысить эффективность применения ЦИЛП для сокращения затрат на проведение экспортно-импортных операций, динамически корректируя проблемные участки ВТД предприятия, используя для этого возможность радиальной диаграммы как инструмента, позволяющего проводить визуальное сравнение между множественными количественными оценками в пределах заданного набора показателей.

Используя ЦИЛП, предприятия-экспортеры получают возможность оперативно корректировать процессы реализации ВТД в строгом соответствии с таможенными регламентами, что расширяет перспективы развития государственно-частного партнерства как эффективного инструмента повышения объема продукции, с существенной долей добавленной интеллектуальной собственности в структуре отечественного экспорта.

Заключение

Положенная в основу ЦИЛП схема-модель CUCA позволяет повысить эффек-

тивность оперативного управления логистических операций и автоматизировать процессы контроля ВТД со стороны таможенных органов. При этом для предприятий-экспортеров высокотехнологичной продукции открываются возможности использования новых бизнес-моделей внешнеторговой деятельности, повышающих надежность, безопасность и скорость торговых операций.

Применение сквозных цифровых технологий защиты прав на интеллектуальную собственность в сочетании с принципом «единого окна» и обработкой данных о ВТД с помощью инфраструктуры распределенных реестров позволяет сформировать единое информационное пространство защищенных транзакций между предприятиями-производителями, компаниями-экспортерами, заказчиками, финансовыми институтами и государственными органами контроля, что обеспечивает в автоматическом режиме мониторинг и оценку эффективности ВТД как отдельных предприятий, так и национальной экономики в целом.

В этом свете изучение возможностей применения современных информационно-технологических подходов и решений, детализированное описание их интеграций является приоритетным направлением в дальнейших исследованиях

Работа выполнена при поддержке гранта РФФИ 18-29-03250 «Методы робастного синтеза интеллектуальных транспортных систем для коалиции киберфизических объектов на основе байесовской концепции вероятности и аппарата модальных логик».

СПИСОК ЛИТЕРАТУРЫ

1. **Sunnie Giles** How VUCA is reshaping the business environment, and what it means for innovation // URL: <https://www.forbes.com/sites/sunniegiles/2018/05/09/how-vuca-is-reshaping-the-business-environment-and-what-it-means-for-innovation/> (Режим доступа: 06.12.2018).
2. **Скиба В.Ю., Скрабовский В.Э.** Предварительное информирование в Евразийском экономическом союзе: состояние и перспективы развития // От Евразийской интеграции к дальневосточному вектору политико-экономических

интересов России: сб. науч. трудов. Владивосток: РИО Владивостокского филиала Российской таможенной академии, 2015. С. 13–17.

3. **Скиба В.Ю., Галанаматис А.Н.** Изменение парадигмы информационного взаимодействия между таможенными и иными контролирующими органами при внедрении принципов и механизмов «единого окна» // От Евразийской интеграции к дальневосточному вектору политико-экономических интересов России: сб. науч. трудов. Владивосток: РИО Владивосток-



ского филиала Российской таможенной академии, 2015. С. 18–28.

4. **Ponten J.** Single window – best practice and the way forward // URL: http://www.uncitral.org/pdf/english/colloquia/EC/Ponten_SW_Best_Practice_and_the_Way_Forward.pdf (Accessed: 15.06.2016).

5. **Скиба В.Ю., Стрекалов С.В.** Реализация концепции «единого окна» в европейском союзе и странах азиатско-тихоокеанского региона: организационный и финансовый аспекты // Таможенная политика России на Дальнем Востоке. 2016. № 3 (76). С. 32–44.

6. Государственно-частное партнерство в России 2016–2017: текущее состояние и тренды, рейтинг регионов. М.: Ассоциация «Центр развития ГЧП», 2016. 32 с.

7. **Зубаков Г.В., Стрекалов С.В.** Актуальные вопросы информационного взаимодействия участников логистических цепей при реализации трансграничных грузовых авиаперевозок // Академический Вестник Ростовского филиала Российской таможенной академии. 2016. С. 21–24.

8. **Моторин Д.Е., Попов С.Г.** Алгоритм многокритериального поиска траекторий движения робота на многослойной карте // Информационно-управляющие системы. 2018. № 3. С. 45–53. DOI:10.15217/ISSN1684-8853.2018.3.45

9. **Motorin D.E., Popov S.G., Chuvatov M.V., Kurochkin M.A., Kurochkin L.M.** A study of the evaluation function for the cost of transport operations in distribution of purpose in a group of robots // XX IEEE Internat. Conf. on Soft Computing and Measurements. St. Petersburg, 2017. Pp. 536–538. DOI: 10.1109/SCM.2017.7970642

10. **Glazunov V., Kurochkin L., Popov S., Chuvatov M.** Prototype of the telematics map cloud service // 17th Conf. of Open Innovations Association. Yaroslavl, 2015. Pp. 50–55. DOI: 10.1109/FRUCT.2015.7117970

11. **Glazunov V.V., Utkin L.V., Ryabinin M.A., Popov S.G.** The rules selection algorithm for network traffic of robot groups in intelligent transportation systems // XX IEEE Internat. Conf. on Soft Computing and Measurements. St. Petersburg, 2017. Pp. 533–535. DOI: 10.1109/SCM.2017.7970641

Статья поступила в редакцию 23.11.2018.

REFERENCES

1. **Sunnie Giles** How VUCA is reshaping the business environment, and what it means for innovation. Available: <https://www.forbes.com/sites/sunniegiles/2018/05/09/how-vuca-is-reshaping-the-business-environment-and-what-it-means-for-innovation/> (Accessed: 06.12.2018).

2. **Skiba V.Yu., Skrabovskiy V.E.** Predvaritelnoye informirovaniye v Yevraziyskom ekonomicheskom soyuze: sostoyaniye i perspektivy razvitiya [Preliminary informing in the Eurasian Economic Union: State and development prospects]. *From Eurasian Integration to the Far Eastern Vector of Russia's Political and Economic Interests: Collection of Scientific Works*. Vladivostok: Vladivostokskii filial Rossiyskoy tamozhennoy akademii Publ., 2015, Pp. 13–17. (rus)

3. **Skiba V.Yu., Galanamatis A.N.** Izmeneniye paradigmy informatsionnogo vzaimodeystviya mezhdru tamozhennymi i inymi kontroliruyushchimi organami pri vnedrenii printsipov i mekhanizmov «yedinogo okna» [Changing the paradigm of information interaction between customs and other regulatory authorities in the implementation of the principles and mechanisms of the "single window"]. *From Eurasian Integration to the Far Eastern Vector of Russia's Political and Economic Interests: Collection of Scientific Works*. Vladivostok: Vladivostokskii filial Rossiyskoy tamozhennoy akademii Publ., 2015, Pp. 18–28. (rus)

4. **Ponten J.** Single window – best practice and the way forward. Available: http://www.uncitral.org/pdf/english/colloquia/EC/Ponten_SW_Best_Practice_and_the_Way_Forward.pdf (Accessed: 15.06.2016).

5. **Skiba V.Yu., Strekalov S.V.** Implementing the “Single Window” concept in the European Union and Asian-Pacific Region: Organizational and financial aspects. *Tamozhennaya politika Rossii na Dalnem Vostoke* [Russian customs policy in the Far East], 2016, No. 3 (76), Pp. 32–44. (rus)

6. *Public-Private Partnership in Russia 2016–2017: Current State and Trends, Rating of Regions*. Moscow: Assotsiatsiya «Tsentr razvitiya GChP» Publ., 2016, 32 p. (rus)

7. **Zubakov G.V., Strekalov S.V.** Aktualnyye voprosy informatsionnogo vzaimodeystviya uchastnikov logisticheskikh tsepey pri realizatsii transgranichnykh gruzovykh aviaperevozok [Topical issues of information interaction of participants in logistics chains in the implementation of cross-border air cargo]. *Akademicheskii Vestnik Rostovskogo Filiala Rossiyskoy Tamozhennoy Akademii* [Academic Bulletin of the Rostov branch of the Russian Customs Academy], 2016, Pp. 21–24. (rus)

8. **Motorin D.E., Popov S.G.** Multi-criteria path planning algorithm for a robot on a multilayer map. *Information and Control Systems*. 2018, No. (3), Pp. 45–53. Available: <https://doi.org/10.15217/issn1684-8853.2018.3.45> (rus)

9. **Motorin D.E., Popov S.G., Chuvatov M.V., Kurochkin M.A., Kurochkin L.M.** A study of the evaluation function for the cost of transport operations in distribution of purpose in a group of robots. *XX IEEE International Conference on Soft Computing and Measurements*, St. Petersburg, 2017, Pp. 536–538. DOI: 10.1109/SCM.2017.7970642

10. **Glazunov V., Kurochkin L., Popov S., Chuvatov M.** Prototype of the telematics map

cloud service. *17th Conference of Open Innovations Association*, Yaroslavl, 2015, Pp. 50–55. DOI: 10.1109/FRUCT.2015.7117970

11. **Glazunov V.V., Utkin L.V., Ryabinin M.A., Popov S.G.** The rules selection algorithm for network traffic of robot groups in intelligent transportation systems. *XX IEEE International Conference on Soft Computing and Measurements*, St. Petersburg, 2017, Pp. 533–535. DOI: 10.1109/SCM.2017.7970641

Received 23.11.2018.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

ЗАБОРОВСКИЙ Владимир Сергеевич

ZABOROVSKY Vladimir S.

E-mail: vlad2tu@yandex.ru

ЛЕЙ Джан

LEI Zhang

E-mail: lzhang@ce.ecnu.edu.cn

СКИБА Владимир Юрьевич

SKIBA Vladimir Yu.

E-mail: vskiba69@mail.ru

СТРЕКАЛОВ Сергей Васильевич

STREKALOV Sergey V.

E-mail: s_strekalov@mail.ru

DOI: 10.18721/JCSTCS.11402
УДК 004.655:004.514;004.657

АЛГОРИТМЫ ДИНАМИЧЕСКОЙ ГЕНЕРАЦИИ MDX-ЗАПРОСОВ К МНОГОМЕРНЫМ OLAP-КУБАМ

С.Г. Попов, А.А. Лисенкова

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

При работе с многомерными базами данных возникает проблема длительного времени доступа к данным. Большинство серверов, поддерживающих технологию OLAP, хранят данные на дисках, тем самым увеличивая время доступа к требуемым данным. Альтернативным способом хранения является использование оперативной памяти, которое, однако, накладывает ограничение на объем памяти. Сервер Palo организован по принципам MOLAP и в качестве хранилища данных использует оперативную память. На данный момент доступ к данным и метаданным сервера осуществляется только с помощью атомарных веб-запросов. Применение MDX расширяет функционал доступа к серверу, а также унифицирует протокол обращения к данным. Результатом работы является подсистема генерации MDX-запроса на выборку данных из многомерных кубов OLAP-сервера Palo с интерактивным графическим интерфейсом. В проекте решены задачи анализа программных средств для работы с MDX-запросами, исследования синтаксиса MDX-запроса на выборку данных, разработки алгоритмов извлечения данных и метаданных из OLAP-кубов, а также реализации подсистемы генерации MDX-запроса на выборку данных из OLAP-куба с интерактивным графическим интерфейсом. Результаты могут использоваться для построения подсистем удаленного доступа к OLAP-серверу Palo.

Ключевые слова: OLAP, MOLAP, MDX, PALO, многомерные кубы, базы данных, метаданные, запросы, динамическая генерация, хранилище данных.

Ссылка при цитировании: Попов С.Г., Лисенкова А.А. Алгоритмы динамической генерации MDX-запросов к многомерным OLAP-кубам // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2018. Т. 11. № 4. С. 21–35. DOI: 10.18721/JCSTCS.11402.

ALGORITHMS FOR MDX-QUERY GENERATION IN MULTIDIMENSIONAL OLAP-CUBES

S.G. Popov, A.A. Lisenkova

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

When working with multidimensional databases, there is a problem of long time access to data. Most servers that support OLAP-technology store data on disks, thus increasing the access time to the required data. An alternative method to store data is using RAM; this, however, imposes a limitation on the amount of memory. The Palo server is organized in accordance with MOLAP principles, and uses random access memory as data storage. Access to the data and metadata of the server is currently provided only through atomic Web requests. Use of MDX extends the functionality of access to the server, and also unifies the protocol of access to data. The result of the study is the subsystem generating MDX-queries for data retrieval from multidimensional cubes of OLAP-server Palo with interactive graphical interface. The project solved the tasks of analyzing software

tools for working with MDX-queries, examining the syntax of an MDX-query for data retrieval, developing algorithms for extracting data and metadata from OLAP-cubes, and implementing the MDX-query generation sub-system for retrieving data from an OLAP-cube with interactive graphical interface. The results can be used to build remote access subsystems to the Palo OLAP-server.

Keywords: OLAP, DBMS, MOLAP, PALO multidimensional cubes, metadata, MDX, queries, dynamic generation, data storage.

Citation: Popov S.G., Lisenkova A.A. Algorithms for MDX-query generation in multidimensional OLAP-cubes. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2018, Vol. 11, No. 4, Pp. 21–35. DOI: 10.18721/JCSTCS.11402.

Введение

Технологии OLAP предоставления данных обеспечивают хранение и предоставление данных для комплексного анализа, состоящее в подготовке обобщенных данных больших массивов, структурированных на основе многомерного принципа. Совокупность естественных и интуитивно понятных моделей, организованных в виде многомерных кубов, обеспечивают чтение данных прикладными программами пользователей. Это позволяет использовать OLAP-системы в области прикладных программ для подготовки данных для бизнес-отчетности, предполагающих представление данных в различных срезах и разных уровнях иерархии, таких как отчеты по продажам, различные формы бюджетов государственных и коммерческих организаций, в том числе и в таких формах анализа данных, как интерактивное прогнозирование трендов [11, 12, 15], что требует быстрого доступа к данным OLAP-систем.

Доступ к данным может осуществляться при помощи нескольких видов интерфейсов, к которым относятся доступ к отдельным элементам многомерных кубов при помощи интерфейса MOLAP, реализованным, например, через web-запросы, или декларативных языков управления многомерными данными MDX. Различные OLAP-системы имеют свои программные интерфейсы для реализации запросов, и не во всех информационных системах существует возможность использования языка запросов MDX. В случае реализации типовых решений, основанных на выполнении запросов из специально разработанного программного обеспечения,

реализация методов как при помощи web-запросов, так и при помощи MDX определяется программной реализацией и не представляет сложности для пользователя, т. к. сложность реализации скрыта за интерфейсом предоставления данных. Однако в случае, если требуется выполнить единичный аналитический запрос, требующий выборки данных из куба, перед конечным пользователем, чаще всего аналитиком, встает проблема получения данных. Она связана с формулировкой результата и передачи её для выполнения специалисту-программисту, который выполнит эти запросы в консольном приложении или путем написания фрагмента кода. Указанный путь долг, т. к. требует участия нескольких специалистов и написания уникального программного кода для каждого запроса на получение данных. При этом аналитики, желающие получить данные из кубов, не будучи знакомыми с низкоуровневыми программными интерфейсами, знакомы с высокоуровневыми языками запросов, такими как MDX. Реализация нового приложения, обеспечивающего аналитика средствами доступа к данным куба средствами MDX-запросов, исключит трудоемкий процесс программирования запроса и ускорит получение данных. Реализация такого MDX-запроса возможна с использованием интерактивных графических форм, предоставляющих шаблоны запроса и требующие только дозаполнения данных о конкретных кубах. Применение интерактивных графических интерфейсов для доступа к многомерным OLAP-кубам позволяет расширить число пользователей, употребляющих технологию OLAP.

Цели и задачи исследования.

Схожие направления исследований

Цель настоящего исследования — разработка алгоритмов и реализация интерактивного графического интерфейса для динамической генерации MDX-запроса на выборку данных из многомерного OLAP-куба, располагающегося на MOLAP-сервере Palo, что предполагает выбор ограничений синтаксиса MDX-запроса на выборку, разработку алгоритма получения метаданных OLAP-куба для построения запроса на выборку, формирование интерактивной экранной формы для выбора атрибутов куба, разработку алгоритма формирования и выполнения MDX-запроса и исследование времени функционирования подсистемы генерации MDX-запроса на выборку данных из OLAP-куба с интерактивным графическим интерфейсом.

Исследование автоматической генерации запросов к OLAP-кубам из графических интерфейсов — активно развивающаяся область исследований. За последние пять лет основными направлениями исследований стали разработка языка обращения к многомерному пространству [8, 14], построение OLAP-кубов данных с помощью стандартных средств web-приложений [9], разработка технологий унифицированных языков доступа к данным куба [10, 13].

В работе [1] описано построение системы, работающей с многомерными аналитическими пространствами для обработки информации — OLAP-кубами данных, с помощью свободно распространяемых бесплатных инструментальных и программных средств для разработки web-приложений. Рассмотрены вопросы построения структуры БД, разработки системы кодирования и основанного на ней языка обращения к многомерному пространству, а также построения и проектирования транслятора, преобразующего выражения, написанные на языке обращений к многомерному пространству, в SQL-подобный язык запросов к БД. Использование описанного в этой статье подхода позволяет осуществлять разработку и построение OLAP-кубов данных

на стандартных средствах разработки web-приложений без привлечения дорогостоящих специализированных программных средств для работы с многомерными аналитическими пространствами, выраженными в форме OLAP-кубов данных.

В статье [2] изучена проблема общения оператора терминала с информационной системой на проблемно-ограниченном подмножестве естественного языка. Предложен механизм семантического анализатора, осуществляющего перевод запроса к информационной системе с естественного языка на SQL-подобный язык запросов среды 1С: Предприятие. Приведена блок-схема самообучения семантического анализатора. Вся совокупность запросов на естественном языке ограничена двумя классами предложений: простые вопросительные и побудительные предложения. Приведены примеры, на которых подробно рассмотрены этапы преобразования запросов с естественного языка.

Представление дизайна и прототипа фреймворка для представления OLAP-операций в виде набора RESTful Web Services рассмотрено в [3]. Web Services были разработаны для загрузки куба, среза, drill-down и других операций.

Представление eLog — BI решения для eBilling для отслеживания управления документами, оптимизации и анализа организовано в работе [4]. Представляет простой и интуитивно-понятный метаязык для неопытных пользователей. Для оптимизации MDX-запросов создали алгоритм для оптимальной идентификации куба среди всех кубов, доступных в хранилище.

В статье [5] приведено описание реляционной OLAP-системы архивных метеоданных, созданной на базе разработанной ROLAP-Mining СПО системы. Ядром системы является СУБД MySQL. MDX-запросы визуализируются с помощью плагина Saiku Analytics. В системе используется технология интеллектуального анализа данных.

В работе [6] предложена OLAP-структура, основанная на web-технологии, которая была разработана из-за недостатков системы OLAP в рамках обычной структу-

ры клиент-сервер. Изучив фактические примеры, была разработана многомерная модель аналитической системы с применением методов связи между Office Web Components и сервером анализа. Кроме этого, реализованы многие операции OLAP-анализа для многомерных данных.

После представления временной модели данных, поддерживающей исторические измерения и ведение таблицы фактов, анонсирована трехуровневая архитектура, основанная на web-сервисах SOAP и XML, что позволило эффективно запрашивать данные из web-источников. В этой архитектуре XML-метаданные хранятся на сервере приложения в виде XML-документов, содержащих структуру хранилищ данных. Это позволяет разрешить доступ для запросов без использования хранилища. Результаты работы представлены в [7].

В [8] рассмотрен системный дизайн компилятора MDX с помощью метода ADD. Это означает, что системные требования, включая функциональные и качественные требования и ограничения, рассматриваются как драйверы в процессе проектирования. Конечная архитектура удовлетворяет не только функциональным требованиям, но и важным качествам, которыми должна обладать система.

Синтаксис MDX-запроса к многомерному OLAP-кубу

Реализуемый запрос на выборку данных из многомерных OLAP-кубов с помощью языка MDX представляет собой набор команд, который имеет следующий вид:

$$\begin{aligned} & SELECT \left[\begin{array}{l} \langle SELECT \text{ query axis clause} \rangle \\ [, \langle SELECT \text{ query axis clause} \rangle \dots] \end{array} \right] \\ & FROM \left[\langle SELECT \text{ subcube clause} \rangle \right] \\ & \left[\langle SELECT \text{ slicer axis clause} \rangle \right] \end{aligned}$$

Оси запроса определяют границы набора ячеек, который возвращается инструкцией многомерных выражений SELECT. Определение границ набора ячеек позволяет сократить количество возвращаемых данных.

Для определения осей запроса используется инструкция $\langle SELECT \text{ query axis clause} \rangle$, связывающая набор данных с определенной осью запроса. Каждое значение в этой инструкции определяет одну ось запроса. Таким образом, число осей в наборе соответствует числу значений $\langle SELECT \text{ query axis clause} \rangle$ в инструкции SELECT.

Для определения оси в инструкции $\langle SELECT \text{ query axis clause} \rangle$ используется следующий синтаксис:

$$\begin{aligned} & \langle SELECT \text{ query axis clause} \rangle ::= [NON \text{ EMPTY}] Set \text{ Expression} \\ & \left[\langle SELECT \text{ dimension property list clause} \rangle \right] \left[\langle HAVING \text{ clause} \rangle \right] \\ & ON \{ \\ & \quad Integer \text{ Expression} | \\ & \quad AXIS (Integer \text{ Expression}) | \\ & \quad \{ COLUMNS | ROWS | PAGES | SECTIONS | CHAPTERS \} \\ & \} \end{aligned}$$

У каждой оси запроса есть номер: 0 для оси X, 1 для оси Y и т. д. В указанном ранее синтаксисе инструкции значение Integer Expression задает номер оси. В запросе многомерных выражений можно указать до 128 осей. Наиболее часто в запросах встречается до пяти осей, поэтому для них созданы специальные псевдонимы COLUMNS, ROWS, PAGES, SECTIONS, CHAPTERS.

Каждый запрос многомерных выражений выполняется для конкретно заданного куба. Контекст куба определяет элементы, которые вычисляются в выражениях запроса.

В инструкции SELECT контекст куба задается с помощью предложения FROM, где в качестве контекста может выступать не только куб целиком, но и его вложенный куб. Указание контекста позволяет расширять или ограничивать его с использованием дополнительных функций.

Предложение FROM определяется следующим синтаксисом:

$$\begin{aligned} & \langle SELECT \text{ subcube clause} \rangle ::= \\ & \quad Cube \text{ Identifier} | \\ & \quad (SELECT [*] \\ & \quad \left[\langle SELECT \text{ query axis clause} \rangle [, \langle SELECT \text{ query axis clause} \rangle \dots] \right] \\ & \quad FROM \langle SELECT \text{ subcube clause} \rangle \langle SELECT \text{ slicer axis clause} \rangle) \end{aligned}$$

В указанном раннее синтаксисе куб (вложенный куб), над которым производится операция выборки, описывается предложением *<SELECT subcube clause>*.

После выполнения инструкции SELECT ось среза производит фильтрацию данных. При этом в конечном итоге получают данные, имеющие пересечения с заданными элементами. Ось среза определяется в предложении WHERE инструкции SELECT.

Для явного определения оси среза используется следующий синтаксис:

(SELECT slicer axis clause) ::= WHERE Set Expression

В приведенном синтаксисе определения оси среза аргумент SET EXPRESSION может принимать либо кортежное выражение, интерпретируемое как набор при вычислении предложения, либо выражение набора. Если задано выражение набора, язык многомерных выражений пытается вычислить набор с использованием статистического вычисления по результирующим ячейкам в каждом кортеже набора. Иными словами, используется функция Aggregate для набора, при этом каждая мера статистически обрабатывается с помощью связанной статистической функции. Кроме того, если выражение набора нельзя выразить в виде перекрестного соединения элементов иерархии атрибутов, при вычислении в языке многомерных выражений ячейки, находящиеся за пределами выражения набора, с помощью которого определена ось среза, интерпретируются, как имеющие значения NULL.

Таким образом, упрощенный запрос на выборку данных из многомерного куба включает в себя следующие ключевые слова: SELECT, FROM, ON COLUMNS, ON ROWS. При этом в предложении FROM указывается имя куба, а каждый показатель оси представляется в виде *[Dimension].[Member]*. Если в выборке участвует несколько членов, то они должны быть сгруппированы и записаны согласно следующему синтаксису:

{[Dimension].[Member],[Dimension].[Member],[...]}

Реализуемый запрос на выборку данных из многомерных OLAP-кубов с помощью графического интерфейса реализуется в два этапа: получение данных и получение метаданных из системы управления данными куба. Реализация этапов приведена в алгоритмах получения метаданных и данных куба.

Алгоритмы интерактивного формирования MDX-запроса к данным OLAP-куба

Алгоритм получения метаданных куба для построения запроса на выборку. При динамической генерации MDX-запроса на выборку необходимо обеспечить корректную последовательность ввода элементов запроса. Прежде всего необходимо указать базу данных, с которой будет происходить взаимодействие. После чего можно приступить к формированию запроса.

Для того чтобы составить запрос, необходимо получить соответствующие метаданные. В контексте многомерных баз данных и, в частности, кубов, метаданными будут являться следующие компоненты: список кубов выбранной базы данных; список измерений, характеризующих куб; список членов, содержащихся в измерении. Учитывая, что минимально допустимый запрос на выборку имеет вид SELECT FROM Cube (где Cube – имя куба в выбранной базе данных), то для такого запроса необходимо получить только список кубов. Для развернутого запроса требуется получить и ограничить список мер, доступных в каждом кубе.

Общая схема получения метаданных для построения запроса на выборку состоит, с учетом особенностей реализации клиент-серверной технологии обращения к OLAP-серверу, в реализации следующих шагов.

1. Запросить список кубов, находящихся в выбранной базе данных.
2. Выбрать куб из списка, сформированного в п. 1, для заполнения предложения FROM.
3. Запросить список размерностей для куба из п. 2.
4. Выбрать размерность из списка, сформированного в п. 3.

5. Запросить меры для выбранной из п. 4 размерности.

6. Выбрать одну из мер, доступных в п. 5, сформировав тем самым одну из инструкций.

7. Повторить п. 3–6 требуемое количество раз для формирования инструкций вида *<SELECT query axis clause>* и, соответственно, определения осей результирующего запроса.

8. Повторить п. 3–6 требуемое количество раз для формирования инструкций вида *<SELECT slicer axis clause>* и, соответственно, определения осей среза.

Блок-схема алгоритма ввода метаданных представлена на рис. 1. Результатом работы алгоритма является сформированный MDX-запрос идентификаторов кубов, размерностей и мер для выбранной системы управления базами данных.

Алгоритм получения данных MDX-запроса. После формирования запроса (были указаны значения куба, размерностей и соответствующих осей), требуется проверить его на синтаксическую корректность, выполнить и продемонстрировать результаты. Эти этапы реализуются в алгоритме получения данных по ранее построенному запросу.

На первом этапе осуществляется проверка синтаксиса полученного запроса. Для выполнения простейшего запроса на выборку необходимо и достаточно присутствие трех лексем: ключевые слова *SELECT FROM* и имя куба. Следует отметить, что в конечном запросе должны участвовать все размерности куба. Поэтому, расширяя минимально допустимый запрос путем добавления описания осей, можно столкнуться с проблемой, когда какая-либо размерность не была указана. В этом случае существуют два способа устранения проблемы. Первый из них заключается в выборе одной меры по умолчанию для каждой неуказанной размерности. Второй использует в запросе все меры неуказанной размерности, т. е. предполагается, что пользователь заведомо хочет выбрать все данные из этой размерности. Выбор варианта реализации отбора мет определяется выбором пользователя для каждого запроса.

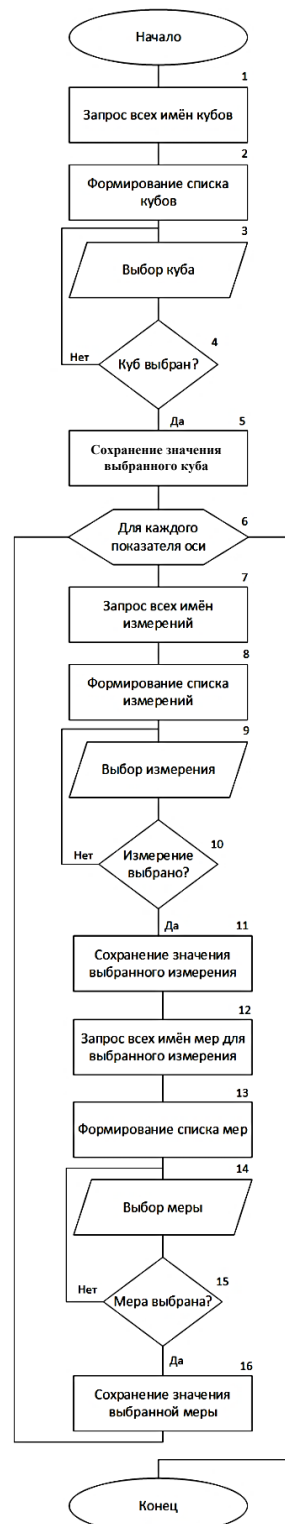


Рис. 1. Алгоритм получения метаданных для MDX-запроса

Fig. 1. The Algorithm of metadata creating for MDX-query

Общая схема получения данных в форме MDX-запроса на выборку состоит, с учетом особенностей реализации графического интерфейса представления OLAP-запроса и клиент-серверной технологии обращения к OLAP-серверу, из следующих шагов.

1. Запрос имени базы данных. Найти структуру в интерфейсе, отвечающую за выбор базы данных, и получить текущее значение имени БД.

2. Запрос имени куба. Найти структуру в интерфейсе, отвечающую за выбор куба (предложение FROM), и получить текущее значение имени куба.

3. Запрос идентификатора сессии. Отправить web-запрос подключения к удаленному серверу и забрать sid (идентификатор сессии).

4. Формирование кортежей, определяющих элементы куба.

5. Запрос результата выполнения MDX-запроса. Сформировать web-запрос на основе полученных данных и отправить его на выполнение к удаленному серверу.

6. Формирование результирующего набора данных. Синтаксический разбор полученного результата выполнения web-запроса для извлечения данных.

7. Формирование итогового MDX-запроса. Получить все пары <Измерение, Мера>, использованные при построении запроса, и составить синтаксически правильный MDX-запрос.

8. Вывод текста запроса и полученных данных. Вывод результирующего набора

данных и полученного MDX-запроса на экран.

Алгоритм получения данных из графического приложения MDX-запроса представлен на рис. 2 Псевдокод формирования текста запроса приведен в листинге.

Результатом работы алгоритма является текст MDX-запроса и данные, полученные в результате его исполнения.

Реализация подсистемы генерации MDX-запроса. Для реализации подсистемы генерации MDX-запроса разработан прототип программного обеспечения. Цель его функционирования состоит в предоставлении интерактивного графического интерфейса для генерации MDX-запроса на выборку данных из OLAP-сервера Palo, с последующим выводом результатов и возможностью просмотра сгенерированного текста запроса. Общая схема функционирования среды представлена на рис. 3. Для выполнения запроса выполняются следующие шаги.

1. Клиентское программное обеспечение эмулирует режим работы MDX, который реализует функциональную возможность выборки данных из многомерных OLAP-кубов.

2. С помощью web-запросов к MOLAP-серверу клиент получает метаданные из многомерных баз данных и формирует структуру запроса на выборку.

3. Конечный MDX-запрос преобразуется в web-запрос и посредством WEB API извлекает требуемые данные из выбранного многомерного OLAP-куба.

1 ПОЛУЧИТЬ список пар <Размерность, Мера> *PairList*;

2 СГРУППИРОВАТЬ меры из *PairList* по размерностям в *MeasureList*

3 for (*i* = 0; *i* < *MeasureList.size* (); *i*++)

4 ЕСЛИ (*MeasureList* [*i*] == null) ТО

5 ДОБАВИТЬ в *MeasureList* все идентификаторы *i*-й размерности;

6 СОСТАВИТЬ комбинации мер в установленном порядке для формирования кортежей, определяющих положение элемента в кубе.

Процедура формирования MDX-запроса
The MDX-query forming procedure

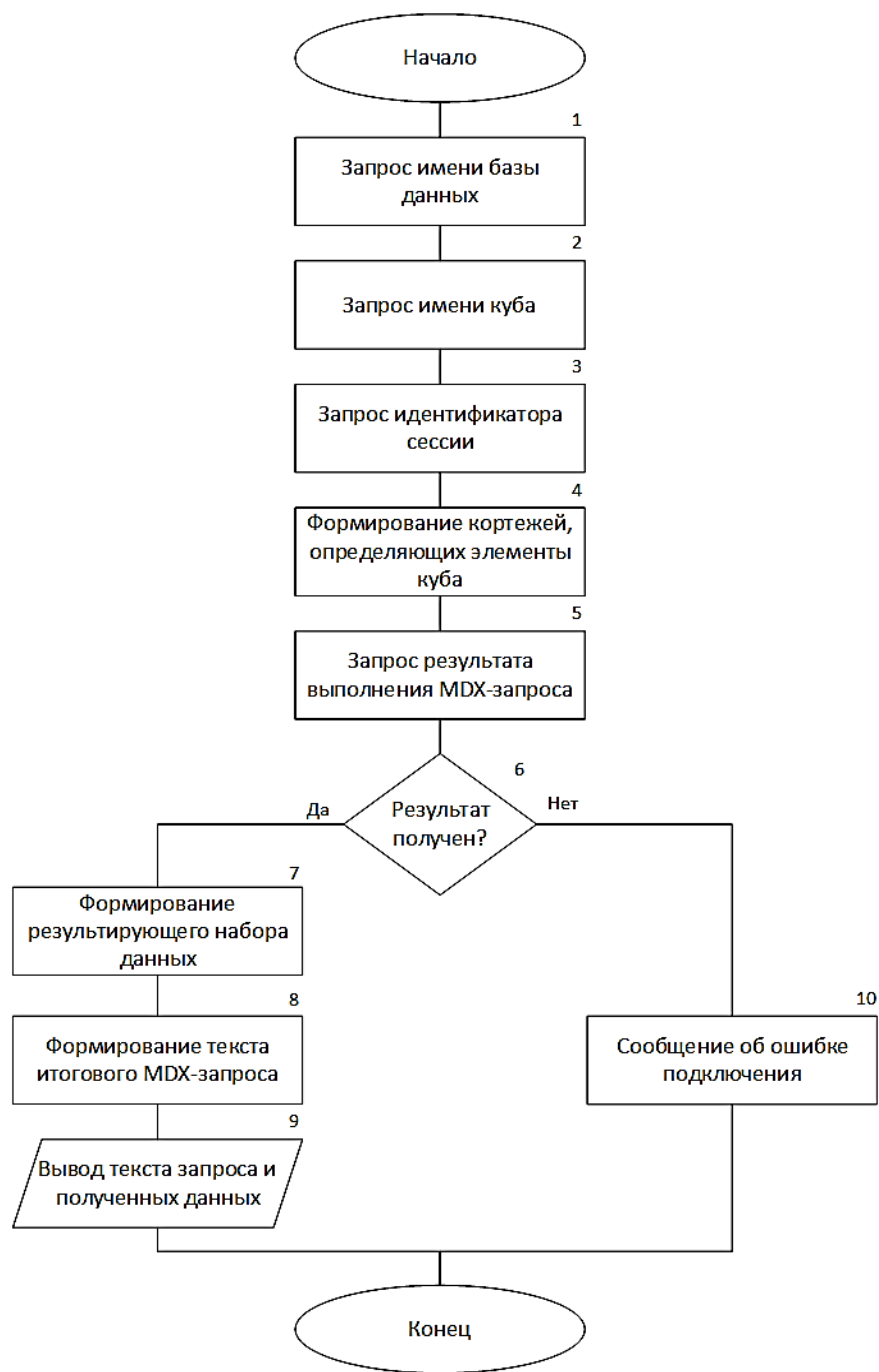


Рис. 2. Алгоритм получения текста и результата MDX-запроса

Fig. 2. The algorithm of text and results MDX-query receiving

Прежде всего для формирования запроса необходимо узнать список баз данных, доступных для работы на сервере. Подключение к MOLAP-серверу осуществляется автоматически при выполнении каждого запроса. Такой подход обеспечивает наличие актуального номера сессии (sid),

чтобы не приходилось его периодически обновлять по истечении тайм-аута. Таким образом, процедура формирования запроса начинается с выбора базы данных из выпадающего списка. После чего пользователю становится доступно основное окно запроса.

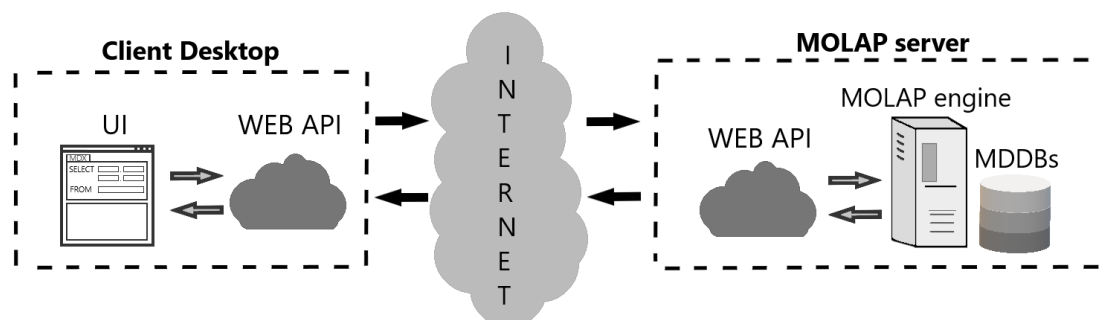


Рис. 3. Общая схема взаимодействия приложения с сервером

Fig. 3. The server and client interconnection general scheme

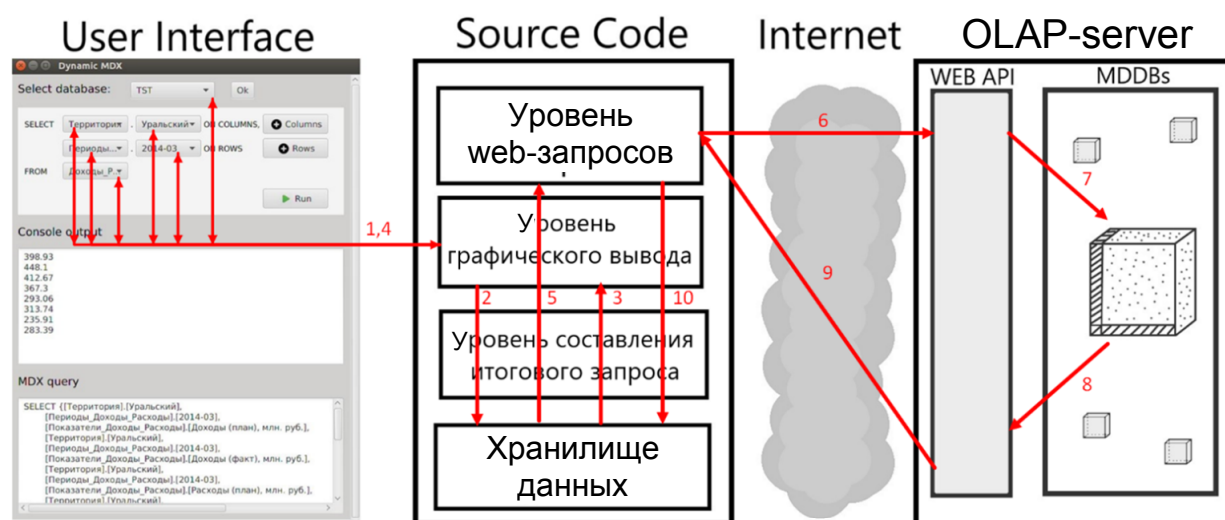


Рис. 4. Схема получения и представления метаданных запроса

Fig. 4. The query metadata getting and representation scheme

На рис. 4 представлена схема получения метаданных для построения запроса. Схема содержит следующие шаги.

1. Выбрать элемент запроса, для которого необходимо получить метаданные.
2. Проверить наличие запрашиваемых метаданных в хранилище.
3. Направить полученный список значений на уровень графического вывода.
4. Поместить полученный список значений в соответствующую структуру из п. 1.
5. Сформировать текст web-запроса.
6. Отправить сформированный запрос к OLAP-серверу.
7. Запросить метаданные из многомерного куба.
8. Передать данные в WEB API.
9. Получить ответ от OLAP-сервера.
10. Произвести синтаксический разбор ответа и сохранить значения в хранилище.

Получение метаданных может быть осуществлено двумя способами.

1. Если в локальном хранилище есть запрашиваемые данные, то маршрут движения по схеме будет определяться следующими шагами: 1, 2, 3, 4.

2. Если в локальном хранилище не оказалось запрашиваемых данных, то необходимо выполнять соответствующий web-запрос. В этом случае маршрут будет выглядеть так: 1, 2, 5, 6, 7, 8, 9, 10, 3, 4.

На рис. 5 представлена схема получения данных для сформированного MDX-запроса. Схема содержит следующие шаги.

1. Забрать текущие значения показателей осей из графического интерфейса.
2. Сохранить полученные значения в хранилище данных.
3. Направить полученные значения на уровень составления итогового запроса.

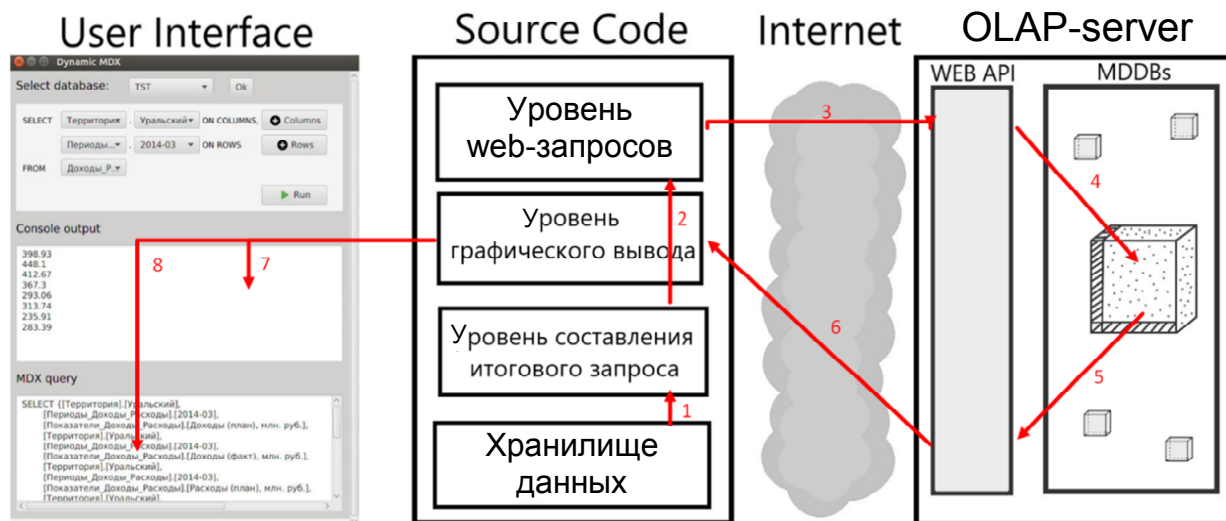


Рис. 5. Схема получения и представления данных запроса
Fig. 5. The data getting and representation scheme

4. Проверить запрос и составить кортежи, определяющие элементы куба.
5. Отправить web-запрос к OLAP-серверу.
6. Запросить данные из многомерного куба.
7. Передать данные в WEB API.
8. Получить ответ от OLAP-сервера.
9. Направить полученные результаты на уровень графического вывода.
10. Вывести на экран полученные значения ячеек.
11. Вывести на экран текст MDX-запроса.

На рис. 6 показан пример интерфейса, заполненный исходными метаданными текстом сформированного запроса и данными, полученными в результате его исполнения.

Согласно алгоритму получения метаданных для построения запроса на выборку, сначала необходимо получить список кубов и выбрать один из предложенных вариантов. Как только пользователь выбирает куб, в поля измерений автоматически подгружаются соответствующие метаданные.

Поле выбора меры измерения остается неактивным до тех пор, пока не будет выбрано соответствующее значение измерения. При этом, как только пользователь выбирает размерность, происходит формирование и отправка web-запроса к MOLAP-

серверу, который должен забрать из базы значения мер. Следует отметить, что значения мер не хранятся в локальном репозитории, а автоматически подгружаются из многомерной БД. Эта операция длится в среднем 40 мс и позволяет не хранить большие объемы метаданных локально.

После заполнения всех полей необходимо составить кортежи, состоящие из идентификаторов мер, для формирования конечного web-запроса. После его выполнения на экран пользователя выводятся результаты и сгенерированный текст MDX-запроса.

Для представления данных используются редактируемые и не редактируемые области. Редактируемые текстовые области: область 2 содержит выбор доступных кубов; область 3 — выбор показателей и мер; 5 — результат выполнения запроса. Области 1, 4 и 6 являются информационными и не подлежат редактированию.

Часть метаданных, полученная с удаленного сервера, хранится локально. К ней относятся имена БД, кубов, а также измерений. Такое решение позволяет сократить количество обращений с помощью web-запросов, тем самым уменьшая время отклика программы. Поскольку не все меры используются при построении запроса, то для каждого выбранного измерения формируется соответствующий запрос для получения списка имен мер.

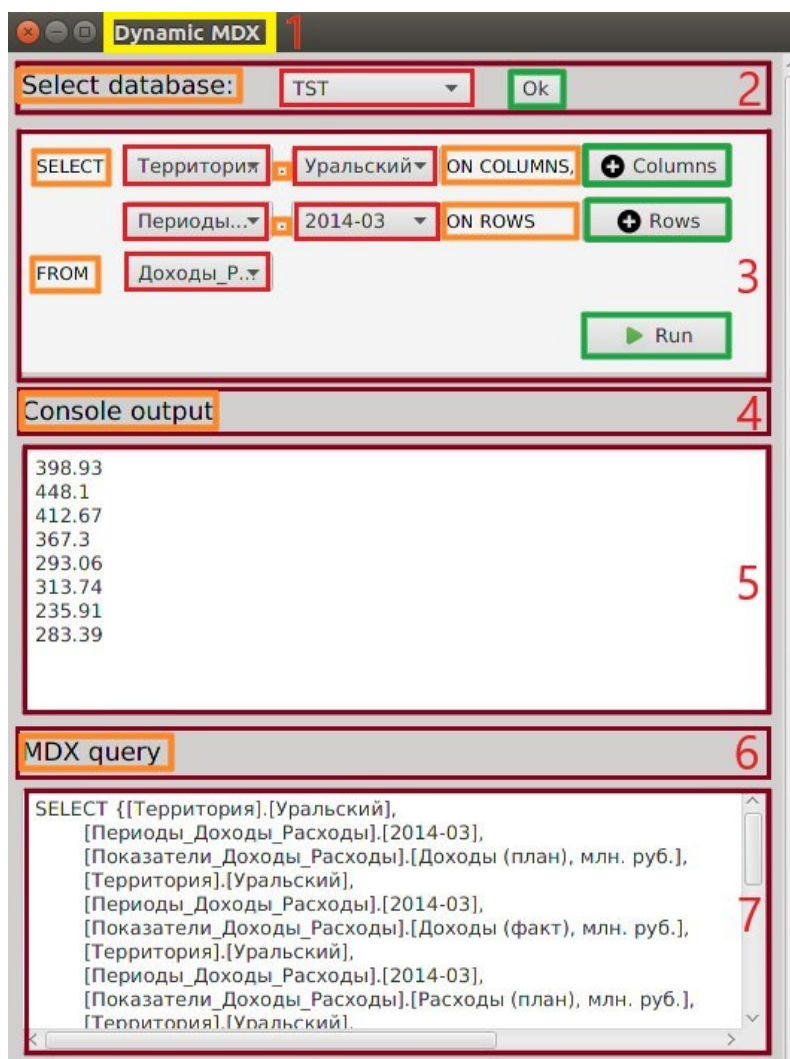


Рис. 6. Структурное содержание программного интерфейса
Fig. 6. The structure of user interface

Для хранения получаемых метаданных используются списки строкового типа. Что касается данных, поступающих после выполнения итогового MDX-запроса, то они направляются сразу в текстовое поле интерфейса без промежуточного хранения, откуда при необходимости могут быть извлечены.

Предложенный интерфейс обеспечива-ет графическое представление и формиро-вание запроса на выборку данных из мно-гомерного куба, результаты которого могут применяться для интерактивного отобра-жения в приложениях пользователя.

Исследование времени выполнения MDX-запроса

Для реализованного прототипа системы интерактивного запроса данных требуется получить и проанализировать временные зависимости выполнения MDX-запроса, что позволит оценить скорость формиро-вания запроса и получения данных с уда-ленного OLAP-сервера.

Постановка эксперимента. Для проведе-ния экспериментов была выбрана тестовая база данных и куб данных «Доходы Расхо-ды», структура которого представлена на рис. 7.

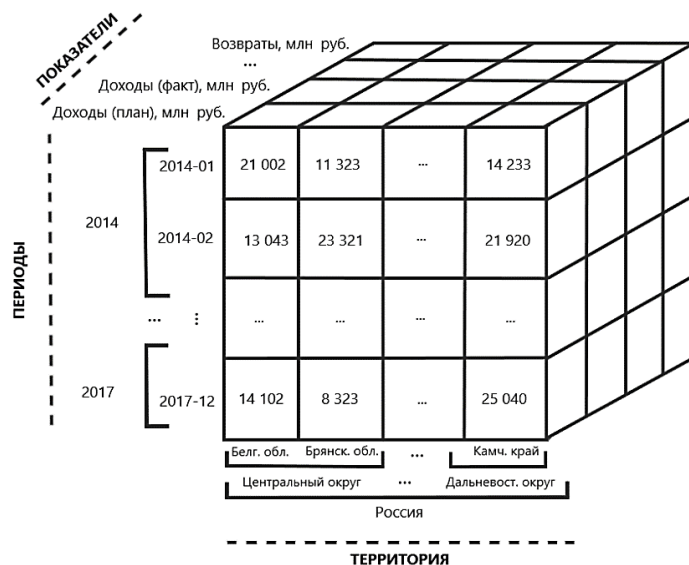


Рис. 7. Структура трехмерного куба данных «Доходы Расходы»

Fig. 7. The structure of thredementional «Income and expenses» cube

В ходе эксперимента необходимо определить время подготовки запроса и время выполнения запроса данных, а также общее время работы алгоритма, что позволит оценить объем данных, доступных для интерактивного запроса. Параметры и константы эксперимента приведены в таблице.

Параметры и константы эксперимента

The experiment constants and parameters

Наименование параметра	Значение
Сервер	
OLAP-сервер	Palo
Способ хранения данных	MOLAP
Тип интерфейса	WEB API
Куб	
Название	Доходы Расходы
Количество измерений	3
Общее количество мер	155
Общее число ячеек	32640
Количество заполненных ячеек	30515
Максимальное количество уровней иерархии	3

Выбранный куб содержит три измерения: «Территория», «Периоды», «Показатели», два из которых представляют собой

иерархию мер с максимальной глубиной, равной трем.

Для представленного куба формировался запрос, содержащий разное число мер и возвращающий различное число ячеек.

Обработка и анализ результатов. Для анализа времени выполнения программного кода были выделены участки программного кода: подготовки метаданных для составления web-запроса, выполнения запроса с учетом времени передачи и приема данных.

На рис. 8 представлена зависимость времени выполнения блоков кода от количества полученных данных и числа мер.

Одной из характеристик интерактивного графического интерфейса является время отклика. Для профессионального программного продукта крайне важно, чтобы большая часть операций выполнялась за минимально допустимое время. Анализируя график, изображенный на рисунке, можно сделать вывод о том, что краткосрочные запросы, которые выполняются менее 0,5 с, будут возвращать до 1650 значений. Запросы средней продолжительности, время выполнения которых варьируется от 0,5 до 2 с, возвращают до 6000 значений. Таким образом, для того чтобы выполнять запросы на выборку данных за приемлемое время, желательно извлекать не более 6000 значений из многомерного OLAP-куба.

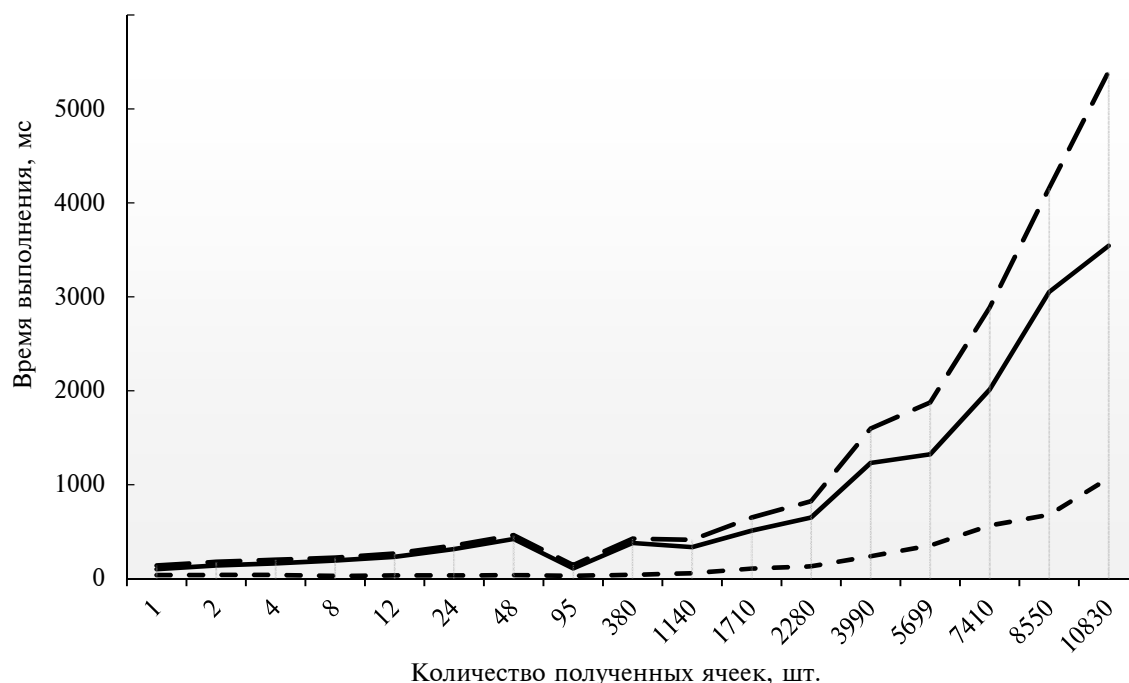


Рис. 8. Зависимость времени выполнения блока кода от количества полученных данных (—) подготовка запроса; (---) выполнение запроса; (- - -) общее время

Fig. 8. The dependence between code running time and data volume

Заключение

Рассмотрены технология генерации MDX-запроса на выборку данных из OLAP-кубов сервера Palo с интерактивным графическим интерфейсом, средства динамической генерации MDX-запросов, которые позволили выявить основные компоненты, необходимые для реализации графического приложения.

С помощью проведенного синтаксического анализа MDX-запроса на выборку данных разработаны алгоритмы получения метаданных и данных из многомерного куба на основе сгенерированного MDX-запроса.

Разработанные алгоритмы реализованы с использованием интерактивного графического интерфейса, позволяющего динамически генерировать текст MDX-запроса, выполнять его на удаленном OLAP-сервере, выводить полученные данные в специальное текстовое поле, а также просматривать текст результирующего MDX-запроса на выборку данных.

Реализован прототип подсистемы генерации MDX-запроса на выборку данных из OLAP-кубов сервера Palo с интерактивным графическим интерфейсом.

Для тестирования построенной подсистемы динамической генерации MDX-запросов выбраны условия проведения эксперимента, целью которого являлось определение объема получаемых данных, обеспечивающего приемлемую скорость выполнения запроса.

Реализованные алгоритмы получения данных и метаданных из многомерных кубов удаленного сервера продемонстрировали работоспособность прототипа в целом. Дальнейшим направлением исследования может стать расширение функциональных возможностей разработанного приложения.

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. Чибисов В.Н., Суслов И.А., Маркилов И.А. Построение OLAP-кубов с помощью стандартных средств разработки web-приложений // Программные продукты и системы. 2013. № 4.
2. Журавлев А.Е. Механизм семантического анализатора запросов к базам данных информационных систем // Вестник Санкт-Петербургского университета. Серия 10. Прикладная математика. Информатика. Процессы управления. 2012. № 4.
3. Patil A.D., Gangadhar N.D. OLaaS: OLAP as a Service // Cloud Computing in Emerging Markets, 2016 IEEE Internat. Conf. on. IEEE, 2016. Pp. 119–124.
4. Шмелева Д.В. и др. Реляционная OLAP-система архивных метеоданных на базе СПО (свободного программного обеспечения) // Научный альманах. 2017. № 5-3. С. 143–146.
5. Li J., et al. The design and implementation of web-based OLAP drilling analysis system // Fuzzy Systems and Knowledge Discovery, 2010 7th Internat. Conf. on. IEEE, 2010. Vol. 6. Pp. 2570–2573.
6. Vaisman A.A., Izquierdo A., Ktenas M. Web-enabled temporal OLAP // Web Congress, 2006. LA-Web'06. 4th Latin American. IEEE, 2006. Pp. 220–229.
7. Zhang P., Xi J. Attribute-driven design of MDX compiler // Computer Science and Software Engineering. 2008 Internat. Conf. on. IEEE, 2008. Vol. 2. Pp. 508–511.
8. Лобел Л. Разработка приложений на основе Microsoft SQL Server 2008. М.: Русская редакция, 2010.
9. Петров С.А. Метод реализации статистических отчетов с поддержкой детализации // Cloud of science. 2018. Vol. 5. No. 4. Pp. 765–777.
10. Чибисов В.Н., Суслов И.А., Маркилов И.А. Построение OLAP-кубов с помощью стандартных средств разработки web-приложений // Программные продукты и системы. 2013. № 4.
11. Наумов В.Н., Лычагина Е.Б., Шарабабаева Л.Ю. Использование BI-систем для обеспечения информационно-аналитической деятельности органов государственной власти // Управленческое консультирование. 2016. № 3(87). С. 144–153.
12. Bhowmik T., Sarkar A., Debnath N.C. OLAP umbrella: Visualization model for multidimensional databases // ACS/IEEE Internat. Conf. on Computer Systems and Applications. Hammamet, 2010. Pp. 1–8.
13. Трофимов С.П., Пономарева О.А. Формализация OLAP-кубов и реляционных связей в базах данных с помощью алгебры множеств // Компьютерный анализ изображений: Интеллектуальные решения в промышленных сетях – 2016. Екатеринбург, 2016. С. 230–233.
14. Артюшина Е.А., Сальников И.И. Технологии in-мемори для хранения, обработки и анализа больших объемов, структурированных и слабоструктурированных данных // XXI век: итоги прошлого и проблемы настоящего плюс. 2018. Т. 7. № 4. С. 147–152.
15. Ноженкова Л.Ф., Шайдуров В.В. OLAP-технологии оперативной информационно-аналитической поддержки организационного управления // Информационные технологии и вычислительные системы. 2010. № 2. С. 15–27.

Статья поступила в редакцию 12.11.2018.

REFERENCES

1. Chibisov V.N., Suslov I.A., Markilov I.A. Building OLAP-cubes using standard web-development applications. *Programmnyye produkty i sistemy [Software&Systems]*, 2013, No. 4. (rus)
2. Zhuravlev A.Ye. Queries semantic analyzer to information systems databases technol. *Vestnik of St. Petersburg State University. Series 10. Applied Mathematics. Computer Science. Control Processes*, 2012, No. 4. (rus)
3. Patil A.D., Gangadhar N.D. OLaaS: OLAP as a Service. *Cloud Computing in Emerging Markets, 2016 IEEE International Conference on*, IEEE, 2016, Pp. 119–124.
4. Shmeleva D.V., et al. Relational OLAP system is the archival of meteorological data on the basis of the SPO (the free software). *Science Almanac*, 2017, No. 5-3, Pp. 143–146. (rus)
5. Li J., et al. The design and implementation of web-based OLAP drilling analysis system. *Fuzzy Systems and Knowledge Discovery, 2010 7th International Conference on*, IEEE, 2010, Vol. 6, Pp. 2570–2573.
6. Vaisman A.A., Izquierdo A., Ktenas M. Web-enabled temporal OLAP. *Web Congress, 2006. LA-Web'06. 4th Latin American*. IEEE, 2006, Pp. 220–229.
7. Zhang P., Xi J. Attribute-driven design of MDX compiler. *Computer Science and Software Engineering, 2008 International Conference on*. IEEE, 2008, Vol. 2, Pp. 508–511.

8. **Lobel L.** *Razrabotka prilozheniy na osnove Microsoft SQL Server 2008 [Application Development Based on Microsoft SQL Server 2008]*. Moscow: Russkaya redaktsiya Publ., 2010. (rus)
9. **Petrov S.A.** Metod realizatsii statisticheskikh otchetov s podderzhkoy detalizatsii. *Cloud of science*, 2018, Vol. 5, No. 4, Pp. 765–777. (rus)
10. **Chibisov V.N., Suslov I.A., Markilov I.A.** The method of implementation of statistical reports with support for detail. *Programmnyye produkty i sistemy*, 2013, No. 4. (rus)
11. **Naumov V.N., Lychagina Ye.B., Sharababayeva L.Yu.** Use of BI-Systems for ensuring information and analytical activity of public authorities. *Administrative Consulting*, 2016, No. 3(87), Pp. 144–153. (rus)
12. **Bhowmik T., Sarkar A., Debnath N.C.** OLAP umbrella: Visualization model for multidimensional databases. *ACS/IEEE International Conference on Computer Systems and Applications – 2010*, Hammamet, 2010, Pp. 1–8.
13. **Trofimov S.P., Ponomareva O.A.** Formalization of OLAP-cubes and relational databases with the algebra of sets. *Computer image analysis: Intelligent solutions in industrial networks – 2016*, Yekaterinburg, 2016, Pp. 230–233. (rus)
14. **Artyushina Ye.A., Salnikov I.I.** In-memory technologies for storing, processing and analysing of large volumes of structured and weakly structured data. *XXI Century: Resumes of the Past and Challenges of the Present plus*, 2018, Vol. 7, No. 4, Pp. 147–152. (rus)
15. **Nozhenkova L.F., Shaydurov V.V.** OLAP-tehnologii operativnoy informatsionno-analiticheskoy podderzhki organizatsionnogo upravleniya [OLAP-technology operational information and analytical support for organizational management]. *Computer Science and Control*, 2010, No. 2, Pp. 15–27. (rus)

Received 12.11.2018.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

ПОПОВ Сергей Геннадьевич

POPOV Sergey G.

E-mail: popovserge@spbstu.ru

ЛИСЕНКОВА Анна Алексеевна

LISENKOVA Anna A.

E-mail: nutochka97@gmail.com

DOI: 10.18721/JCSTCS.11403

УДК 004.7

АДАПТИВНОЕ НЕЧЕТКОЕ УПРАВЛЕНИЕ ЗАПОЛНЕНИЕМ БУФЕРА СЕТЕВОГО УСТРОЙСТВА

А.А. Сорокин, Т. Чан Куок

Астраханский государственный технический университет,
г. Астрахань, Российская Федерация

Предложена методика управления процессом передачи информационного трафика за счет регулирования уровня загрузки буферов сетевых устройств. Методика основана на применении адаптивного нечеткого контроллера. Принцип работы контроллера основан на совместном использовании методов теории нечеткой логики и теории нечетких нейронных сетей. При помощи системы нечеткого вывода контроллер рассчитывает вероятности маркирования пакета меткой о перегрузке буфера, после чего пакет отправляется узлу-получателю. Узел-получатель при получении маркированного пакета отправляет команду на снижение интенсивности передачи трафика узлом-отправителем, а в случае отсутствия подобной команды узел-отправитель повышает интенсивность передачи трафика. Адаптация контроллера к изменению специфики передаваемого трафика производится с использованием нечеткой нейронной сети, при помощи которой анализируются корректные и некорректные управляющие воздействия, сформированные контроллером. На основании анализа формируются корректирующие воздействия на алгоритм работы контроллера. Проверка эффективности результатов исследований показала снижение вероятности потери информационных пакетов и увеличение уровня использования каналов в сети передачи данных.

Ключевые слова: сеть передачи данных, управление трафиком, нечеткий контроллер, система адаптации, нечеткая нейронная сеть, прецедент, обучение.

Ссылка при цитировании: Сорокин А.А., Чан Куок Т. Адаптивное нечеткое управление заполнением буфера сетевого устройства // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2018. Т. 11. № 4. С. 36–48. DOI: 10.18721/JCSTCS.11403

ADAPTIVE FUZZY CONTROL FOR BUFFER LOADING REGULATION OF NETWORK NODE

A.A. Sorokin, T. Tran Quoc

Astrakhan State Technical University,
Astrakhan, Russian Federation

The goal of this study is in developing a method for controlling the transmission of information packet traffic. The control is based on buffer loading regulation of network node. The fuzzy controller is used for regulation of buffer loading. The operating principles of the controller combine fuzzy logic theory methods and fuzzy neural networks theory. The controller relies on a system of fuzzy logic to calculate the probabilities of marking a packet with a buffer overload flag. After this the packet is sent

to the destination node. After receiving a packet with a buffer overload flag, the destination node sends the command to reduce the traffic intensity to sending node. When the controller starts to work incorrectly, the adaptation algorithm is used. The adaptation algorithm is based on using the fuzzy neural network that processes information about the control result of information packet traffic transmission. After this the fuzzy neural network generates commands for adjusting the controller. We have verified the effectiveness of the study's results, finding a reduced probability of loss of information packets and an increase in the level of using channels in the data network.

Keywords: data transmission network, traffic management, fuzzy controller, adaptation system, fuzzy neural network, precedent, training.

Citation: Sorokin A.A., Tran Quoc T. Adaptive fuzzy control for buffer loading regulation of network node. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2018, Vol. 11, No. 4, Pp. 36–48. 10.18721/JCSTCS.11403

Введение

В настоящее время Интернет является глобальной сетевой системой передачи данных (СПД). С каждым днем растет количество подключений и расширяется прикладное значение Интернета [1], в частности, в области внедрения технологии Интернет вещей (Internet of Things – IoT) [2]. Чаще всего повышение эффективности передачи трафика осуществляется двумя способами [3, 4]: увеличением производительности аппаратной части оборудования и разработкой программного обеспечения для повышения эффективности решения задач, возникающих в процессе работы сетевого устройства.

Увеличение производительности оборудования сдерживается ростом его стоимости и энергопотребления, что особенно критично, если в систему передачи входит большое количество узлов, в том числе предназначенных для длительной автономной работы. Кроме этого, рост объемов передаваемого трафика быстро нивелирует выигрыш повышения эффективности работы СПД за счет увеличения производительности аппаратной части оборудования. Повышение эффективности работы сетевого оборудования благодаря усовершенствованию программной части включает оптимизацию работы протоколов, используемых для управления работой сети в целом, и повышение эффективности работы алгоритмов управления очередью внутри сетевых узлов.

Как показал анализ [4, 5], разработка или модернизация сетевых протоколов

осложняется необходимостью решения организационных задач, связанных с согласованием работы оборудования различных поколений и производителей, анализом и устранением потенциальных угроз информационной безопасности, исследованием взаимного влияния работы нового протокола (или модернизированной версии используемого) с уже существующими системами и методами управления. Алгоритмы управления очередью ориентированы на работу «внутри» определенного сетевого узла, а формируемые ими управляющие воздействия ориентированы на применение в условиях использования существующих сетевых протоколов.

Цель данного исследования – повышение эффективности передачи трафика за счет разработки методики управления загрузкой буферов сетевых устройств.

Методика управления интенсивностью передачи трафика

Управление очередью производится непосредственно в узлах сети передачи данных за счет регулирования количества информационных пакетов в буфере устройства. Управляющим воздействием на интенсивность передачи трафика является величина вероятности удаления пакета из буфера или вероятность установки значения специальной служебной отметки – «маркера» у информационного пакета, что он является причиной перегрузки. Удаление пакетов с определенной вероятностью из буфера производится в случае, если

протокол транспортного уровня, используемый в сети передачи данных, не поддерживает функцию маркирования пакетов. Далее рассуждения ведутся при допущении того, что протокол транспортного уровня поддерживает функцию маркирования пакетов, если они вызывают перегрузку буфера сетевого узла. Допущение оправдано тем, что большинство используемых протоколов транспортного уровня, ориентированных на эксплуатацию сетей, работающих с использованием протоколов стека TCP/IP, поддерживают функцию маркирования пакетов [6].

Если узел-получатель принимает такой маркированный пакет, то им производится отправка служебного сообщения узлу-отправителю о необходимости снизить интенсивность передачи информации. После снижения интенсивности снижается загрузка буфера сетевого узла, что снижает вероятность или исключает маркирование информационных пакетов. Если узел-получатель принимает информационные пакеты без маркеров о перегрузке, то он отправляет команду, разрешающую узлу-отправителю повысить интенсивность передачи. Использование подобных принципов регулирования позволяет повысить эффективность передачи трафика по таким критериям, как вероятность потери пакетов, средняя задержка передачи пакета, эффективность использования каналов передачи данных и др. Повышение эффективности передачи трафика по приведенным критериям позволяет повысить качество услуг, оказываемых сетью.

Обзор работ [7, 8] показал, что перспективным направлением по созданию систем управления загруженностью буфера сетевого устройства является применение контроллеров, использующих для обработки информации о специфике интенсивности нагрузки методы теории нечетких множеств. Анализ [7, 8] показал, что ограничение предлагаемых методов — недостаточная способность к адаптации при возникновении изменений специфики передаваемого трафика. С учетом общетеоретических работ по созданию систем нечеткого вывода (СНВ) [9, 10] ограничения в

адаптивности вызываются тем, что разрабатываемые системы управления в качестве своей основы используют статические базы знаний, статические области определения функций принадлежности (ФП) входных и выходных переменных. Кроме того, авторы [7, 8] отдают предпочтение ФП треугольной и трапецеидальной формы, тогда как с учетом [11, 12] применение сигмоидальных, колоколообразных или гауссовых ФП более целесообразно с позиции точности обобщения экспертных знаний об изменении степени принадлежности значенных переменных к определенному терму.

Сущность предлагаемого принципа управления интенсивностью передачи трафика заключается в использовании адаптивного нечеткого контроллера для определения управляющего воздействия — вероятности маркирования информационного пакета, который стал причиной превышения рекомендованной длины очереди $P_m(\Delta q)$, где Δq — разница между рекомендованной длиной очереди в буфере устройства $q_{рек}$ и фактической длиной очереди в буфере $q_{факт}$. Адаптивная способность контроллера обеспечивается дополнительным модулем, анализирующим удачные и неудачные прецеденты управления процессом передачи информационных пакетов. На основании анализа прецедентов формируются корректирующие воздействия на контроллер, что изменяет его функциональный оператор, при помощи которого производится расчет управляющих воздействий на объект управления. Изменения функционального оператора происходят за счет изменения параметров ФП принадлежности. Применительно к колоколообразным ФП это коэффициент концентрации r , координата максимума v , коэффициент крутизны функции s . Соотношение для задания колоколообразной ФП с учетом работ [9, 10] имеет вид:

$$\mu(x) = \frac{1}{1 + [(x - v) / r]^2 \cdot s}, \quad (1)$$

где x — область значения переменной, которая проецируется на область определения функции принадлежности $\mu(x)$.

Получение значения $P_m(\Delta q)$ можно представить в виде соотношения:

$$\begin{aligned} P_m(\Delta q) &= F_{\Delta q}(\Delta q_{t_n}, \Delta q_{t_{n-1}}) : \Delta q_{t_n} = \\ &= q_{\text{рек}} - q_{\text{факт } t_n}, \Delta q_{t_{n-1}} = q_{\text{рек}} - q_{\text{факт } t_{n-1}}, \end{aligned} \quad (2)$$

где $F_{\Delta q}$ — оператор для обработки значений входных переменных Δq_{t_n} и $\Delta q_{t_{n-1}}$.

При этом цель управления — выполнение условия:

$$\Delta q = |q_{\text{рек}} - q_{\text{факт}}| \rightarrow \min. \quad (3)$$

Величину Δq можно отнести к критерию оптимизации, при этом границами того, что управление процессом передачи трафика оптимально, является:

$$\begin{aligned} q_{\text{факт}} \in (q_{\text{рек min}}, q_{\text{рек max}}) : q_{\text{рек min}} &= q_{\text{рек}} - q_{\text{доп}}, \\ q_{\text{рек max}} &= q_{\text{рек}} + q_{\text{доп}}, \end{aligned} \quad (4)$$

где $q_{\text{доп}}$ допустимая разница отклонения $q_{\text{факт}}$ от величины $q_{\text{рек}}$.

Оценка эффективности управления осуществляется по условиям:

- минимизации вероятности потери информационных пакетов $P_{\text{lost}}^{\text{pack}}$ в зависимости от интенсивности их передачи $\lambda_{\text{pack}}(t)$ в течение времени t :

$$P_{\text{lost}}^{\text{pack}}[\lambda_{\text{pack}}(t)] \rightarrow \min; \quad (5)$$

- повышение коэффициента загрузки линий передачи $L_{\text{line}}^{\text{pack}}$ в СПД в зависимости от интенсивности их передачи $\lambda_{\text{pack}}(t)$ в течение времени t :

$$L_{\text{line}}^{\text{pack}}[\lambda_{\text{pack}}(t)] \rightarrow \max; \quad (6)$$

- сохранение величины времени пребывания пакета в очереди $T_{\text{p in q}}^{\text{нов}}$ в пределах существующих значений $T_{\text{p in q}}^{\text{сущ}}$ в зависимости от интенсивности поступления пакетов $\lambda_{\text{pack}}(t)$ в течение времени t :

$$T_{\text{p in q}}^{\text{нов}}[\lambda_{\text{pack}}(t)] \leq T_{\text{p in q}}^{\text{сущ}}[\lambda_{\text{pack}}(t)]. \quad (7)$$

В рамках проводимых исследований перед описанием входных переменных

Δq_{t_n} и $\Delta q_{t_{n-1}}$ при ФП они проходят процедуру нормализации, для того чтобы их значения принадлежали интервалу $[-1; 1]$. Подобное нужно для повышения универсальности использования результатов. Так, если использовать абсолютные значения величин, потребуется корректировка области значений входных переменных для всех устройств, имеющих разный размер буфера $q_{\text{буфер}}$. Нормализация производится при помощи соотношения вида:

$$\begin{aligned} \Delta q_{t_n}^{\text{norm}} &= K \cdot \Delta q_{t_n}, \\ \Delta q_{t_{n-1}}^{\text{norm}} &= K \cdot \Delta q_{t_{n-1}}, \end{aligned} \quad (8)$$

где K_i — нормировочный коэффициент, определяющийся при помощи выражения:

$$K = \begin{cases} 1/q_{\text{рек}} & \text{если } q_{\text{факт } t_n} < q_{\text{рек}} \\ & \text{или } q_{\text{факт } t_{n-1}} < q_{\text{рек}}, \\ 1/(q_{\text{буфер}} - q_{\text{рек}}) & \text{если } q_{\text{факт } t_n} \geq q_{\text{рек}} \\ & \text{или } q_{\text{факт } t_{n-1}} \geq q_{\text{рек}}. \end{cases} \quad (9)$$

В результате использования (8) и (9) величины $\Delta q_{t_n}^{\text{norm}}$ и $\Delta q_{t_{n-1}}^{\text{norm}}$ принимают значения в диапазоне $[-1, 1]$. После нормирования производится преобразование значений переменных $\Delta q_{t_n}^{\text{norm}}$ и $\Delta q_{t_{n-1}}^{\text{norm}}$ от четкого значения к нечеткому:

$$\begin{aligned} \Delta q_{t_n}^{\text{norm}} &\xRightarrow{\text{FUZZ}} \Delta q_{t_n}^{\text{fuzzy}}, \\ \Delta q_{t_{n-1}}^{\text{norm}} &\xRightarrow{\text{FUZZ}} \Delta q_{t_{n-1}}^{\text{fuzzy}}. \end{aligned} \quad (10)$$

Для этого переменные $\Delta q_{t_n}^{\text{fuzzy}}$ и $\Delta q_{t_{n-1}}^{\text{fuzzy}}$ описываются при помощи терм множеств $T_{\Delta q_{t_n}} = \{t_{\Delta q_{t_n}}^k\}$ и $T_{\Delta q_{t_{n-1}}} = \{t_{\Delta q_{t_{n-1}}}^k\}$, где k — число элементов множества, состоящего из семи термов: *NB* (Negative Big) — очень низкий, *NM* (Negative Medium) — средне низкий, *NS* (Negative Small) — скорее средний, чем низкий, *ZE* (Zero Every) — средний, *PS* (Positive Small) — скорее средний, чем высокий, *PM* (Positive Medium) — скорее высокий, *PB* (Positive Big) — очень высокий. Каждому терму со-

ответствует функция принадлежности $\mu(\Delta q_{t_n})$, описываемая соотношением (1), со специфичными для каждого из термов параметрами r , v , s . На момент начала работы контроллера принимается, что ФП каждой из переменных Δq_{t_n} и $\Delta q_{t_{n-1}}$ имеют идентичные параметры. Общий вид ФП переменных Δq_{t_n} и $\Delta q_{t_{n-1}}$ показан на рис. 1.

Для обобщения значений входных переменных используется база знаний, состоящая из правил вида:

$$\vartheta: \text{if}(\Delta q_{t_n}^{\text{fuzz}} = t_{\Delta q_{t_n}}^{\kappa}) \wedge (\Delta q_{t_{n-1}}^{\text{fuzz}} = t_{\Delta q_{t_{n-1}}}^{\kappa}) \Rightarrow (P_m^{\text{fuzz}}(\Delta q) = y_{P_m(\Delta q)}), \quad (11)$$

$$(\kappa \in [1, 7]), v \in [1, 7]), \vartheta \in [1, \vartheta_{\max} = 49],$$

где ϑ — номер правила; $t_{\Delta q_{t_n}}^{\kappa}$ и $t_{\Delta q_{t_{n-1}}}^{\kappa}$ — элементы терм множеств; $y_{P_m(\Delta q)}$ — выходное значение каждого правила.

При этом справедливо, что

$$[\forall t_{\Delta q_{t_n}}^{\kappa} \exists \mu_{\Delta q_{t_n}}^{\kappa}(\Delta q_{t_n})] \wedge [\forall t_{\Delta q_{t_{n-1}}}^{\kappa} \exists \mu_{\Delta q_{t_{n-1}}}^{\kappa}(\Delta q_{t_{n-1}})] = 1, \quad (12)$$

где $\mu_{\Delta q_{t_n}}^{\kappa}(\Delta q_{t_n})$ и $\mu_{\Delta q_{t_{n-1}}}^{\kappa}(\Delta q_{t_{n-1}})$ — ФП соответствующих термов. В рамках проводимых исследований это колоколообразные функции, описываемые соотношением (1), с соответствующими значениями параметров r , v , и s ; $y_{P_m(\Delta q)}$ — элемент кортежа выходных значений.

В рамках работы количество элементов $y_{P_m(\Delta q)}$ в кортеже равно семи:

$$Y_{P_m(\Delta q)} = \langle y_{P_m(\Delta q)}^1 = 0, y_{P_m(\Delta q)}^2 = 0,167, y_{P_m(\Delta q)}^3 = 0,33, y_{P_m(\Delta q)}^4 = 0,5, y_{P_m(\Delta q)}^5 = 0,667, y_{P_m(\Delta q)}^6 = 0,834, y_{P_m(\Delta q)}^7 = 1,0 \rangle. \quad (13)$$

Обобщение правил вида (11) производится при помощи базы знаний и алгоритма нечеткого вывода, что можно представить в виде соотношения:

$$\bigcup_{\vartheta=1}^{49} \left[(\Delta q_{t_n}^{\text{fuzz}} = t_{\Delta q_{t_n}}^{\kappa}) \wedge (\Delta q_{t_{n-1}}^{\text{fuzz}} = t_{\Delta q_{t_{n-1}}}^{\kappa}) \Rightarrow \right] \xRightarrow{\text{defuzz}} P_m(\Delta q). \quad (14)$$

Соотношение (14) — формализованное представление соотношения (2), роль оператора $F_{\Delta q}$ выполняет алгоритм нечеткого вывода. Для выполнения операции defuzz — дефаззификации с учетом работ [9, 10] используется выражение:

$$P_m(\Delta q) = \left(\sum_{\vartheta=1}^{49} c_{\vartheta} \cdot y_{P_m(\Delta q)\vartheta} \right) / \sum_{\vartheta=1}^{49} c_{\vartheta}, \quad (15)$$

где

$$c_{\vartheta} = \min[\mu_{\Delta q_{t_n}}^{\kappa}(\Delta q_{t_n}) = c_{\Delta q_{t_n}}, \mu_{\Delta q_{t_{n-1}}}^{\kappa}(\Delta q_{t_{n-1}}) = c_{\Delta q_{t_{n-1}}}], \quad (16)$$

где $c_{\Delta q_{t_n}}$ и $c_{\Delta q_{t_{n-1}}}$ — значения входных переменных, полученных при помощи фаззификации ФП соответствующих термов.

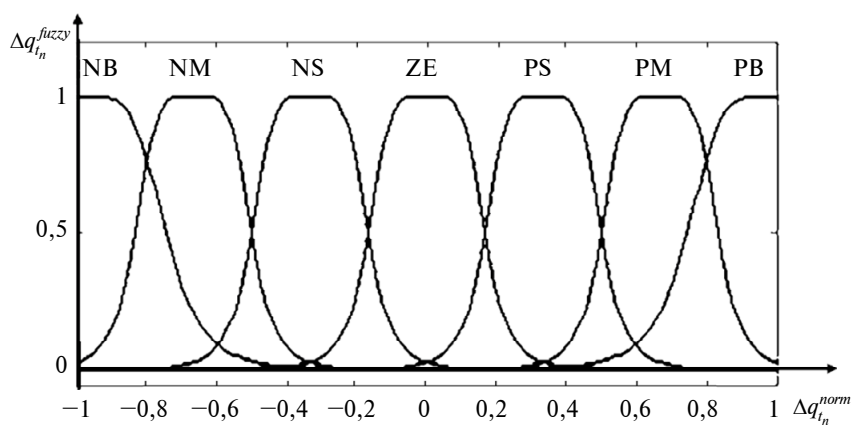


Рис. 1. Функции принадлежности входных переменных Δq_{t_n} и $\Delta q_{t_{n-1}}$

Fig. 1. Membership functions of input variables Δq_{t_n} and $\Delta q_{t_{n-1}}$

Как показывает исследование [13], перспективным направлением развития систем нечеткого вывода является разработка методов их адаптации к изменениям условий внешней среды. Из анализа соотношений (11)–(16) видно, что предложенная система нечеткого вывода имеет статичные функции принадлежности, базу знаний и области определения выходной переменной. Корректировка СНВ может производиться ручным или автоматическим методами. Формально задачу корректировки можно представить в виде:

$$\Delta x_{CY} = |x_{\text{реальн}} - x_{\text{ожид}}| \rightarrow \min, \quad (17)$$

где $x_{\text{реальн}}$ — результат работы системы после управляющего воздействия, $x_{\text{ожид}}$ — ожидаемая величина выхода системы после управляющего воздействия.

Применение ручных методов корректировки СНВ ограничивается длительной задержкой между моментом времени, когда система управления начинает выдавать неадекватные команды, и моментом, когда начинается производиться корректировка. Кроме того, учитывая, что топология систем связи может состоять из большого количества узлов, требуется разработка корректирующих воздействий на системы управления каждого из них. Причина этого заключается в том, что параметры потока информационных пакетов на каждом из узлов могут быть разные. Следовательно, необходимо использовать автоматические системы корректировки, позволяющие оптимизировать процесс обслуживания трафика на каждом из узлов по критерию (3) при соблюдении ограничений (5)–(7).

Метод адаптации нечеткого контроллера под изменение специфики передаваемого трафика

Анализ работ [9, 10, 13] показал, что корректировка СНВ, использующих алгоритм нечеткого вывода Сугено, предусматривается при помощи аппарата нечетких нейронных сетей (ННС). В процессе работы ННС формируется множество корректирующих воздействий Q_s . Множество Q_s состоит из двух подмножеств: 1) корректирующих воздействий, оказываемых на ФП

$Q_s^{\text{ФП}} = \{q_s^{\text{ФП}}\}$, где $q_s^{\text{ФП}}$ — кортеж корректирующих значений ФП определенного термина, применительно к колоколообразным функциям $q_s^{\text{ФП}} = \langle \Delta r_i, \Delta v_i, \Delta s_i \rangle$, где Δr_i — корректирующее значение коэффициента концентрации, Δv_i — корректирующее значение координаты максимума, Δs_i — корректирующее значение коэффициента крутизны; 2) уточненных значений области определения выходной переменной $Q_s^{\text{ВыхП}} = \{y_{P_m(\Delta q)}^{\text{new } j}\}$, которое является преобразованным множеством выходных значений, описываемых соотношением (13).

Формально подобное высказывание можно представить в виде:

$$Q_s = Q_s^{\text{ФП}} \cup Q_s^{\text{ВыхП}} : Q_s^{\text{ФП}} = \{q_s^{\text{ФП}}\}, \quad (18)$$

$$Q_s^{\text{ВыхП}} = \{y_{P_m(\Delta q)}^{\text{new } j}\} : q_s^{\text{ФП}} = \langle \Delta r_i, \Delta v_i, \Delta s_i \rangle.$$

Для запуска корректировки производится постоянный мониторинг наполняемости буфера информационными пакетами. Если при заданных значениях Δq_{t_n} , $\Delta q_{t_{n-1}}$ и формировании $P_m(\Delta q)$ через интервал времени τ уровень очереди находится в пределах допустимых границ $q_{\text{буф}} \in (q_{\text{рек min}}, q_{\text{рек max}})$, то формируется положительный прецедент p^+ вида:

$$\langle \Delta q_{t_n}, \Delta q_{t_{n-1}}, P_m(\Delta q) \rangle \Rightarrow p^+ : q_{\text{буф}} \in [q_{\text{рек min}}, q_{\text{рек max}}] : q_{\text{рек min}} \leq q_{\text{буф}} \leq q_{\text{рек max}}. \quad (19)$$

Если после обработки переменных Δq_{t_n} , $\Delta q_{t_{n-1}}$ и формировании $P_m(\Delta q)$ через интервал τ уровень очереди не находится в пределах допустимых границ $q_{\text{буф}} \notin (q_{\text{рек min}}, q_{\text{рек max}})$, то формируется отрицательный прецедент; если $q_{\text{буф}} < q_{\text{рек min}}$, прецедент обозначается как p_-^- :

$$\langle \Delta q_{t_n}, \Delta q_{t_{n-1}}, P_m(\Delta q) \rangle \Rightarrow p_-^- : q_{\text{буф}} \notin (q_{\text{рек min}}, q_{\text{рек max}}) : q_{\text{буф}} < q_{\text{рек min}}, \quad (20)$$

если $q_{\text{буф}} > q_{\text{рек max}}$, формируется отрицательный прецедент вида p_+^- :

$$\begin{aligned} <\Delta q_{t_n}, \Delta q_{t_{n-1}}, P_m(\Delta q) > \Rightarrow p_{>}^- : q_{\text{буф}} \notin \\ &\notin (q_{\text{рек min}}, q_{\text{рек max}}) : q_{\text{буф}} > q_{\text{рек min}}. \end{aligned} \quad (21)$$

Таким образом, в отдельной базе данных формируются записи прецедентов p^+ , $p_{<}^-$ и $p_{>}^-$, где n_{p^+} , $n_{p_{<}^-}$ и $n_{p_{>}^-}$ — количество соответствующих прецедентов. Максимальный объем записей в базе данных прецедентов равен n_{max} . Решение о запуске алгоритма адаптации $d_{\text{corr}} = 1$ принимается при выполнении условия:

$$d_{\text{corr}} = \begin{cases} 1, & \theta < (n_{p_{<}^-} + n_{p_{>}^-}) / n_{\text{max}}, \\ 0, & \theta \geq (n_{p_{<}^-} + n_{p_{>}^-}) / n_{\text{max}}. \end{cases} \quad (22)$$

Величины θ и n_{max} определяются эмпирически, в результате имитационного моделирования или последовательной самонастройки системы адаптации. В рамках исследований целесообразным признано, что $\theta = 10$, а $n_{\text{max}} = 100$.

После запуска процедуры обучения формируется обучающая выборка. В обучающей выборке положительные прецеденты не изменяются, а для отрицательных выполняется корректировка параметров $P_m(\Delta q)$. При выполнении условия (20) корректировка $P_m(\Delta q)$ производится при помощи добавления фиксированной величины P_m . В результате формируется кортеж вида:

$$\begin{aligned} <\Delta q_{t_n}, \Delta q_{t_{n-1}}, \Pi_m^>(\Delta q) > \Rightarrow \pi_{>}^- : \Pi_m^>(\Delta q) = \\ &= P_m(\Delta q) + \Delta P_m. \end{aligned} \quad (23)$$

При выполнении условия (21) корректировка $P_m(\Delta q)$ производится при помощи вычитания величины P_m . В результате формируется кортеж вида:

$$\begin{aligned} <\Delta q_{t_n}, \Delta q_{t_{n-1}}, \Pi_m^<(\Delta q) > \Rightarrow \pi_{<}^- : \Pi_m^<(\Delta q) = \\ &= P_m(\Delta q) - \Delta P_m. \end{aligned} \quad (24)$$

В рамках исследований принимается, что $\Delta P_m = 0,01$ (1 %). После формирования выборка подается на вход ННС. Окончательно выборка S^{HHC} имеет вид:

$$\begin{aligned} S^{\text{HHC}} &= \{Q^{\text{Stud}}, P^{\text{Stud}}\} : Q^{\text{Stud}} = \{q_m^{\text{Stud}}\}, \\ P^{\text{Stud}} &= P \cup \Pi^> \cup \Pi^< : q_m^{\text{Stud}} = \{\Delta q_{t_n}, \Delta q_{t_{n-1}}\}_m, \\ P &= \{P_m\}, \Pi^> = \{\Pi_m^>\}, \Pi^< = \{\Pi_m^<\} : \\ &\Pi_m^> \in p_{>}^-, \Pi_m^< \in p_{<}^-. \end{aligned} \quad (25)$$

Начальное состояние ННС аналогично состоянию СНВ контроллера, входные переменные описываются аналогичным набором термов, имеющих аналогичные области значения и определения, база знаний ННС содержит аналогичный набор правил, начальные значения выходных переменных также аналогичны СНВ контроллера. Значения q_m^{Stud} подаются на вход ННС, после они обрабатываются соотношениями, аналогичными (11)–(16), и формируется множество выходных значений $P_m^{\text{HHC}}(\Delta q)$:

$$\forall q_m^{\text{Stud}} \exists P_m^{\text{HHC}}(\Delta q) : q_m^{\text{Stud}} \xRightarrow{\text{Алгоритм Сугено}} P_m^{\text{HHC}}(\Delta q). \quad (26)$$

Затем определяется погрешность при помощи соотношения:

$$E_k = \frac{1}{2} [P_m^{\text{HHC}}(\Delta q) - P_m^{\text{Stud}}]^2. \quad (27)$$

Если E_k больше критического значения $E_{k \text{ min}}$, начинается процесс корректировки параметров нейронной сети. Сущность корректировки заключается в следующем. Как известно, величина $P_m^{\text{HHC}}(\Delta q)$ получена в результате обработки входного кортежа q_m^{Stud} совокупностью термов ФП входных переменных, базой правил. В рамках проводимых исследований ФП описываются соотношением (1), база знаний — (14), набор выходных значений, используемых в правилах, — (11). С учетом [13], при обучении ННС производится корректировка параметров ФП и области значений выходных переменных, используемых в правилах. Цель обучения ННС — минимизация ошибки E_k . Таким образом, задачу обучения нейронной сети аналогично (17), с учетом (27), можно представить в виде:

$$E_k = \frac{1}{2} [P_m^{\text{HHC}}(\Delta q) - P_m^{\text{Stud}}]^2 \rightarrow E_{k \text{ min}}, \quad (28)$$

с учетом (1), (13), (14), $P_m^{\text{HHC}}(\Delta q)$ при заданных q_m^{Stud} можно представить в виде:

$$\begin{aligned} P_m^{\text{HHC}}(\Delta q) &= F_m^{\text{HHC}}[\{s_i^{\text{in1}}\}, \{r_i^{\text{in1}}\}, \\ &\{v_i^{\text{in1}}\}, \{s_i^{\text{in2}}\}, \{r_i^{\text{in2}}\}, \{v_i^{\text{in2}}\}, \{y_{P_m(\Delta q)}\}], \end{aligned} \quad (29)$$

где $\{s_i^{in1}\}, \{r_i^{in1}\}, \{v_i^{in1}\}$ — параметры термов ФП переменной $\Delta q_{t_n}^{fuzz}$; $\{s_i^{in2}\}, \{r_i^{in2}\}, \{v_i^{in2}\}$ — параметры термов ФП переменной $\Delta q_{t_{n-1}}^{fuzz}$; $y_{P_m(\Delta q)}$ — начальное значение правил, с учетом (13), $y_{P_m(\Delta q)} \in Y_{P_m(\Delta q)}$; F_m^{HNC} — оператор для обработки входных значений, в роли которого выступают соотношения, описывающие форму ФП, и используемый алгоритм нечеткого вывода. Обучение сводится к итеративному подбору параметров соотношения (29) для выполнения (28):

$$E_k = \frac{1}{2} (F_m^{HNC}[\{s_i^{in1}\}, \{r_i^{in1}\}, \{v_i^{in1}\}, \{s_i^{in2}\}, \{r_i^{in2}\}, \{v_i^{in2}\}, y_{P_m(\Delta q)}] - P_m^{Stud})^2 \rightarrow E_{k \min}. \quad (30)$$

Определение новых значений $\{s_i^{in1}\}, \{r_i^{in1}\}, \{v_i^{in1}\}, \{s_i^{in2}\}, \{r_i^{in2}\}, \{v_i^{in2}\}$ и $y_{P_m(\Delta q)}$ с учетом [13], проводится методом градиентного спуска. В результате соотношения имеют вид:

$$y_{P_m(\Delta q)}^{s+1} = y_{P_m(\Delta q)}^s - \eta \partial E_k / \partial y_{P_m(\Delta q)} = y_{P_m(\Delta q)}^s - \eta (P_m^{HNC}(\Delta q) - P_m^{Stud}) \alpha_j / \left(\sum_{\pi=1}^{49} \alpha_{\pi} \right), \quad (31)$$

где η — коэффициент обучения (в рамках исследований, по рекомендации экспертов $\eta = 0,2$), s — число итераций обучения, $j \in [1, 49]$; для нахождения параметров ФП первой входной переменной $\Delta q_{t_n}^{fuzz}$:

$$s_i^{in1 s+1} = s_i^{in1 s} - \eta \partial E_k / \partial s_i^{in1}, \quad (32)$$

$$r_i^{in1 s+1} = r_i^{in1 s} - \eta \partial E_k / \partial r_i^{in1}, \quad (33)$$

$$v_i^{in1 s+1} = v_i^{in1 s} - \eta \partial E_k / \partial v_i^{in1}, \quad (34)$$

где $\partial E_k / \partial y_{P_m(\Delta q)j}$, $\partial E_k / \partial s_i^{in1}$, $\partial E_k / \partial r_i^{in1}$, $\partial E_k / \partial v_i^{in1}$ — градиенты функции изменения погрешности относительно определенного параметра ФП или области значения выходной переменной.

Для нахождения параметров ФП переменной $\Delta q_{t_{n-1}}^{fuzz}$ используются соотношения, аналогичные (32)–(34). Градиенты по-

грешности коэффициента концентрации ФП s_i^j переменной $\Delta q_{t_n}^{fuzz}$:

$$\frac{\partial E_k}{\partial s_i^{in1}} = (P_{q \text{ out}}^k - P_q^k) \frac{\partial}{\partial s_i^{in1}} (P_{q \text{ out}}^k ([\{s_i^{in1}\}, \{r_i^{in1}\}, \{v_i^{in1}\}], [\{s_i^{in2}\}, \{r_i^{in2}\}, \{v_i^{in2}\}], \{P_{qj}\})); \quad (35)$$

градиент погрешности коэффициента крутизны ФП r_i^j переменной $\Delta q_{t_n}^{fuzz}$:

$$\frac{\partial E_k}{\partial r_i^{in1}} = (P_{q \text{ out}}^k - P_q^k) \frac{\partial}{\partial r_i^{in1}} (P_{q \text{ out}}^k ([\{s_i^{in1}\}, \{r_i^{in1}\}, \{v_i^{in1}\}], [\{s_i^{in2}\}, \{r_i^{in2}\}, \{v_i^{in2}\}], \{P_{qj}\})); \quad (36)$$

градиент погрешности координаты максимума ФП v_i^j переменной $\Delta q_{t_n}^{fuzz}$:

$$\frac{\partial E_k}{\partial v_i^{in1}} = (P_{q \text{ out}}^k - P_q^k) \frac{\partial}{\partial v_i^{in1}} (P_{q \text{ out}}^k ([\{s_i^{in1}\}, \{r_i^{in1}\}, \{v_i^{in1}\}], [\{s_i^{in2}\}, \{r_i^{in2}\}, \{v_i^{in2}\}], \{P_{qj}\})). \quad (37)$$

Для нахождения градиентов погрешности параметров ФП переменной $\Delta q_{t_{n-1}}^{fuzz}$ соотношения аналогичны (32)–(34).

С учетом формул (31)–(37) получается управляющее воздействие Q_s , которое оказывает на нечеткий контроллер:

$$Q_s = (\{r_i^j\}, \{s_i^j\}, \{v_i^j\}, \{y_{P_m(\Delta q)}\}), \quad (38)$$

где $i = 1, 2; j = 1, 2, \dots, 7; m = 1, 2, \dots, 49$; значения элементов Q_s указываются в соотношениях, описывающих выполнение работы нечеткого контроллера.

Результат корректировки ФП входных переменных контроллера показан на рис. 2. Также формируется множество скорректированных значений выходной переменной $P_m^{fuzz}(\Delta q)$. Учитывая особенности обучения ННС, количество элементов множества $Y_{P_m(\Delta q)}^{stud}$, при помощи которого описывается переменная $P_m^{fuzz}(\Delta q)$, составляет 49 (равно общему количеству правил в базе знаний). Результаты зависимости значений выходной переменной (вероятности маркировки пакета) $P_m^{fuzz}(\Delta q)$ от параметров Δq_{t_n} и $\Delta q_{t_{n-1}}$ до и после работы системы адаптации приведены на рис. 3. Как видно из ана-

лиза рисунков, после работы системы адаптации значения выходной переменной имеют более плавный характер, что позволяет реже применять команды на снижение интенсивности передачи трафика. Метод получил название AFRED (Adaptive Fuzzy Controller – Random Early Detection).

Завершающим этапом исследования стала проверка эффективности предложенной системы управления интенсивностью передачи трафика. Для моделирования процессов, связанных с генерацией трафика, его передачей и обслуживанием, использовался программный пакет Network Simulator (NS-2.35) [14].

Экспериментальная часть

Для реализации разработанных методов управления трафиком, для симулятора

NS-2.35 были написаны дополнительные модули на языке программирования высокого уровня C++. Схема моделируемой сети показана на рис. 4.

Исходные данные для экспериментов следующие: размер пакета – 1000 байт, размер буфера очереди – 500 пакетов, тип протокола TCP – TCP/NewReno, периоды проведения измерений $\Delta t_n = 0,006$ с, время моделирования – 100 с. Для сравнения использован «классический» метод управления интенсивностью передачи трафика на основе мониторинга очереди – RED (Random Early Detection) [15] и FEM (Fuzzy Explicit Marking) [7] – модернизированный метод RED, в котором вероятность маркировки пакета определяется с использованием нечеткого вывода. Проведено две серии экспериментов.

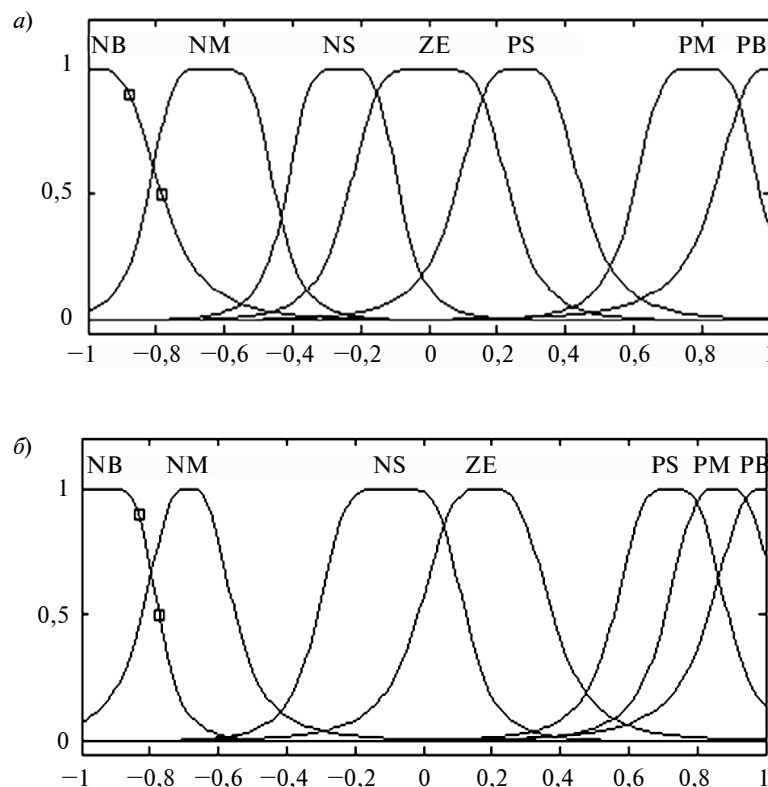


Рис. 2. Функции принадлежности переменных Δq_{t_n} (а) и $\Delta q_{t_{n-1}}$ (б)

Fig. 2. Membership functions of variables Δq_{t_n} (a) and $\Delta q_{t_{n-1}}$ (b) после работы системы адаптации нечеткого контроллера

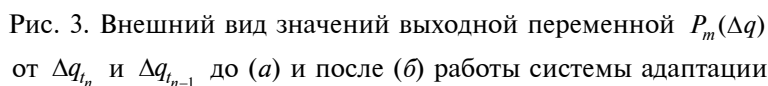


Рис. 4. Схема исследуемой сети в NS-2
Fig. 4. Scheme of the studied network in NS-2

Во время второй серии экспериментов измерялось количество входящих потоков трафика информационных пакетов в диапазоне $N_{\text{поток}}^{\text{вход}} \in [10, 300]$ при рекомендованной длине очереди $q_{\text{рек}} = 200$ пакетов. Критериями оценки эффективности обслуживания трафика информационных пакетов были вероятность потери пакетов $P_{\text{потер}}^{\text{пакет}}, \%$ и уровень использования линии передачи $L_{\text{лин. пер}}^{\text{использ}}, \%$. Результаты моделирования представлены на рис. 6.

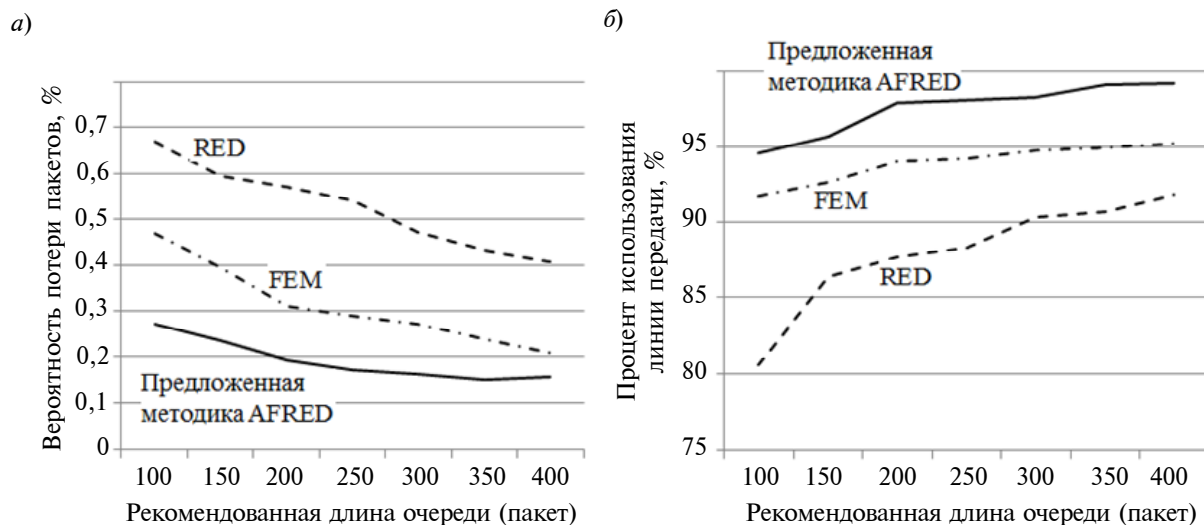


Рис. 5. Результаты передачи трафика при рекомендованных длинах очереди в буфере

Fig. 5. Results of traffic transmission with recommended queue lengths in the buffer

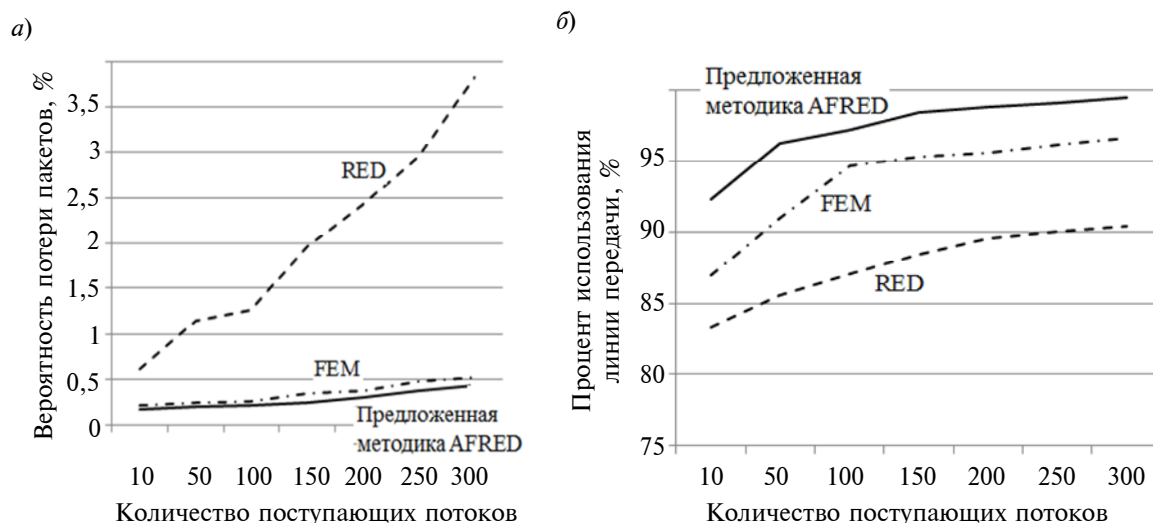


Рис. 6. Сравнение передачи трафика при изменении количества входящих потоков информационных пакетов

Fig. 6. Comparison of traffic when changing the number of incoming information packet flows

Обобщая анализ результатов моделирования, приведенных на рис. 5 и 6, можно сделать вывод, что в описанных условиях эксперимента предложенный метод управления трафиком информационных пакетов превзошел аналогичные решения RED и FEM. В результате проведенных экспериментов показано повышение эффективности передачи трафика информационных пакетов по сравнению с указанными методами по критерию вероятности потери па-

кета до 62,7 %, по критерию эффективности использования пропускной способности канала в среднем на 6,1 %.

Заключение

Предложена методика управления интенсивностью передачи трафика, основанная на использовании адаптивного нечеткого контроллера, при помощи которого производится определение вероятности маркирования пакета. Данный параметр влияет

на формирование команд, регулирующих скорость передачи информационных пакетов узлом-отправителем информации. Для реагирования на изменение особенностей передаваемого трафика в контроллере предусмотрена система адаптации, учитывающая адекватность управляющих воздействий, выработанных контроллером. Если количество ошибочных воздействий превысит критическое значение, система адаптации скорректирует механизмы работы нечеткого контроллера. Сущность корректировки заключается в формировании команд на изменение параметров функций принадлежности входных переменных и области значений выходной переменной, что в результате приводит к уточнению базы знаний нечеткого контроллера.

Эффективность использования предложенных теоретических положений проверена при помощи имитационного модели-

рования в программном комплексе Network Simulator (NS-2.35). Для сравнения эффективности использованы известные методы управления интенсивностью передачи трафика за счет регулирования уровня загрузки буферов сетевых узлов, такие как RED и FEM. В качестве критериев эффективности использовались вероятность потери информационных пакетов и доля использования пропускной способности канала. Относительно известных методов по критерию вероятность потери информационных пакетов доля потерянных пакетов сокращается до 62,7 %, а эффективность использования пропускной способности линии повышается до 6,1 %.

Таким образом, предложенные теоретические положения открывают возможности повышения эффективности передачи трафика в современных инфокоммуникационных системах.

СПИСОК ЛИТЕРАТУРЫ

1. Cisco прогнозирует трехкратное увеличение IP-трафика с 2014 по 2019 гг. // URL: https://www.cisco.com/c/ru_ru/about/press/press-releases/2015/06-01.html
2. Maraden Y., Hardjawana W., Vucetic B. Contention resolution algorithm for industrial Internet-of-Things networks // IEEE 4th World Forum on Internet of Things (WF-IoT). 2018. Pp. 493–498 DOI: 10.1109/WF-IoT.2018.8355192
3. Ибрагимов Б.Г., Гусейнов Ф.И. Исследование и анализ эффективности передачи мультимедийного трафика в сети NGN/IMS // T-Comm: Телекоммуникации и транспорт. 2015. Т. 9. № 12. С. 27–31.
4. Дмитриев В.Н., Сорокин А.А., Чан Куок Тоан, Фам Хак Чонг. Повышение эффективности управления трафиком в гетерогенных системах передачи данных в условиях неопределенности // Вестник Астраханского государственного технического университета. Управление, вычислительная техника и информатика. 2015. № 1. С. 67–77.
5. Деарт В.Ю., Кожухов И.С., Пилюгин А.В. Разработка экспериментальной платформы для исследования качества восприятия (QOE) услуг потоковой передачи видео // T-Comm: Телекоммуникации и транспорт. 2013. Т. 7. № 7. С. 32–35.
6. Deart V., Maslennikov A., Gaidamaka Y. A hysteretic model of queuing system with fuzzy logic active queue management // Proc. of 15th Conf. of Open Innovations Association FRUCT. 2014. Pp. 32–38. DOI: 10.1109/FRUCT.2014.6872419
7. Chrysostomou C., Pitsillides A., Hadjipollas G. Fuzzy logic congestion control in TCP/IP Best-Effort Networks. University of Cyprus, Monash University Melbourne, Australia, 2007. Pp. 2–5.
8. Xu C., Li F. A congestion control algorithm of fuzzy control in routers // 4th Internat. Conf. on Wireless Communications, Networking and Mobile Computing. 2008. Pp. 1–4. DOI: 10.1109/WiCom.2008.1078
9. Леоненков А.В. Нечеткое моделирование в среде MATLAB и fuzzyTECH. СПб.: БХВ-Петербург, 2005. 736 с.
10. Аль Б.М., Иванова О.Г., Дидрих В.Е. Нечеткая модель оценки функционирования сетевых систем интеллектуальной информационной системы // Вестник Воронежского института МВД России. 2014. № 1. С. 105–111.
11. Бобырь М.В., Кулабухов С.А., Конорева М.В. Сравнительный анализ использования сигмоидальных и треугольных функций принадлежности в алгоритмах нечетко-логического вывода // Современные инструментальные системы, информационные технологии и инновации: Сб. науч. трудов XII Междунар. науч.-практ. конф. 2015. С. 194–196.
12. Бобырь М.В., Нассер А.А., Абдулджабар М.А. Исследование свойств мягкого алгоритма нечетко-логического вывода // Известия Юго-Западного государственного университета. 2016. № 1 (64). С. 31–49.

13. Fuller R. Neural Fuzzy Systems. Abo Akademi University, 1995. 348 p.

14. Описание программного пакета NetworkSimulator – ns-2 // URL: <http://www.isi.edu/nsnam/ns/>

15. Al-Momni R.L., Karimi B. Gateway Feedback Congestion Control (GFCC) algorithm // 1st Internat. Scientific Conf. of Engineering Sciences – 3rd Scientific Conf. of Engineering Science. 2018. Pp. 7–12. DOI: 10.1109/ISCES.2018.8340519

Статья поступила в редакцию 08.05.2018.

REFERENCES

1. Cisco: press-release. Available: https://www.cisco.com/c/ru_ru/about/press/press-releases/2015/06-01.html
2. Maraden Y., Hardjawana W., Vucetic B. Contention resolution algorithm for industrial Internet-of-Things networks. *IEEE 4th World Forum on Internet of Things (WF-IoT)*, 2018, Pp. 493–498. DOI: 10.1109/WF-IoT.2018.8355192
3. Ibragimov B.G., Guseynov F.I. Research of the quality functioning of the NGN/IMS networks in establishing a multimedia session. *T-Comm*, 2015, Vol. 9, No. 12, Pp. 27–31. (rus)
4. Dmitriyev V.N., Sorokin A.A., Chan Kuok Toan, Fam Khak Chong. Increase of efficient management of traffic in heterogeneous systems of data transmission under uncertain conditions. *Vestnik of Astrakhan State Technical University. Management, Computer Science and Informatics*, 2015, No. 1, Pp. 67–77. (rus)
5. Deart V.Yu., Kozhukhov I.S., Pilyugin A.V. Experimental platform for QoE research of streaming video services. *T-Comm*, 2013, Vol. 7, No. 7, Pp. 32–35. (rus)
6. Deart V., Maslennikov A., Gaidamaka Y. A hysteretic model of queuing system with fuzzy logic active queue management. *Proceedings of 15th Conference of Open Innovations Association FRUCT*, 2014, Pp. 32–38. DOI: 10.1109/FRUCT.2014.6872419
7. Chryostomou C., Pitsillides A., Hadjipollas G. *Fuzzy logic congestion control in TCP/IP Best-Effort Networks*. University of Cyprus, Monash University Melbourne, Australia, 2007, Pp. 2–5.
8. Xu C., Li F. A Congestion control algorithm of fuzzy control in routers. *4th International Conference on Wireless Communications, Networking and Mobile Computing*, 2008, Pp. 1–4. DOI: 10.1109/WiCom.2008.1078
9. Leonenkov A.V. *Nechetkoye modelirovaniye v srede MATLAB i fuzzyTECH [Fuzzy simulation in MATLAB and fuzzyTECH]*. St. Petersburg: BKhV-Petersburg Publ., 2005, 736 p. (rus)
10. Al B.M., Ivanova O.G., Didrikh V.Ye. Fuzzy model for the estimation of functioning of network systems of the intellectual information system. *The Bulletin of Voronezh Institute of the Ministry of Internal Affairs of Russia*, 2014, No. 1, Pp. 105–111. (rus)
11. Bobyr M.V., Kulabukhov S.A., Konoreva M.V. Comparative analysis of the use of sigmoidal and triangular functions of membership in the algorithms of fuzzy logic inference. *Modern Instrumental Systems, Information Technologies and Innovations. Proceedings of Scientific Papers of the XII International Scientific and Practical Conference*, 2015, Pp. 194–196. (rus)
12. Bobyr M.V., Nasser A.A., Abduldzhabbar M.A. Investigation of the properties of soft algorithm fuzzy inference. *Proceedings of the Southwest State University*, 2016, No. 1 (64), Pp. 31–49. (rus)
13. Fuller R. *Neural Fuzzy Systems*. Abo Akademi University Publ., 1995, 348 p.
14. Description of the software package Network Simulator – ns-2. Available: <http://www.isi.edu/nsnam/ns/>
15. Al-Momni R.L., Karimi B. Gateway Feedback Congestion Control (GFCC) algorithm. *1st International Scientific Conference of Engineering Sciences – 3rd Scientific Conference of Engineering Science*, 2018, Pp. 7–12. DOI: 10.1109/ISCES.2018.8340519

Received 08.05.2018.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

СОРОКИН Александр Александрович

SOROKIN Alexandr A.

E-mail: alsorokin.astu@mail.ru

ЧАН Куок Тоан

TRAN Quoc Toan

E-mail: quoctoan.astu@mail.ru

DOI: 10.18721/JCSTCS.11404
УДК 519.63-688; 534-8, 534.222

РЕШЕНИЕ ЗАДАЧИ ПОСТРОЕНИЯ УЛЬТРАЗВУКОВЫХ ИЗОБРАЖЕНИЙ МЕТОДОМ ВОЛНОВОЙ ИНВЕРСИИ С ИСПОЛЬЗОВАНИЕМ АЛГОРИТМОВ РАСПАРАЛЛЕЛИВАНИЯ ВЫЧИСЛЕНИЙ

К.А. Верещагин, И.Н. Белых

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

Активно развивающееся направление ультразвуковой томографии требует разработки эффективных подходов решения прямых и обратных задач для волнового уравнения как на теоретическом, так и на прикладном уровне. В статье изучены методы, применяемые для решения задач реконструкции изображений из волнового поля, и предложен оригинальный алгоритм построения двухмерных срезов ультразвуковых изображений. Для акустически неоднородной модели упругой среды проведено решение прямой задачи для волнового уравнения в двухмерной постановке с учетом поглощения с целью получения полного поля отраженных волн. Представлен оптимизационный алгоритм 2D-волновой инверсии ультразвукового поля для решения обратной задачи во временной области. Приведены основные архитектурные элементы программного решения, рассмотрены способы распараллеливания вычислений для повышения производительности процесса моделирования. Обсуждены результаты тестирования и проверки на корректность различных этапов алгоритма. Достоинствами предложенного подхода являются точность и устойчивость решения при повышенной вычислительной производительности.

Ключевые слова: волновое уравнение, отраженные волны, волновая инверсия, оптимизация, параллельные вычисления, ультразвуковая томография.

Ссылка при цитировании: Верещагин К.А., Белых И.Н. Решение задачи построения ультразвуковых изображений методом волновой инверсии с использованием алгоритмов распараллеливания вычислений // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2018. Т. 11. № 4. С. 49–62. DOI: 10.18721/JCSTCS.11404

ULTRASOUND IMAGE RECONSTRUCTION BASED ON WAVEFORM INVERSION METHOD USING PARALLEL CALCULATION ALGORITHMS

K.A. Vereshchagin, I.N. Belykh

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

Ultrasound tomography is actively developing area that requires effective research and development efforts in forward and inverse problems solution for wave equation. In this work different methods of image reconstruction from elastic wave field are

investigated and original algorithm for two-dimensional ultrasound imaging is suggested. Full reflection wave field including absorption was modeled by means of forward problem for 2-D acoustic wave equation solution for inhomogeneous elastic media. Invers problem is solved by optimization algorithm for two-dimensional full waveform inversion of reflection wave field. The main features of the software application design are described, using parallel calculation techniques for the modeling process efficiency improvement. The testing results are discussed and algorithm steps are validated for correctness. The advantages of the proposed approach are accuracy and stability of the solution with improved computational efficiency.

Keywords: wave equation, reflection waves, waveform inversion, optimization, parallel calculations, ultrasound tomography.

Citation: Vereshchagin K.A., Belykh I.N. Ultrasound image reconstruction based on waveform inversion method using parallel calculation algorithms. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2018, Vol. 11, No. 4, Pp. 49–62. DOI: 10.18721/JCSTCS.11404

Введение

Ультразвуковые исследования (УЗИ) широко используются в медицине и неразрушающем контроле для дистанционного определения внутренней структуры изучаемых объектов. В настоящее время активно развиваются различные подходы к ультразвуковой томографии (УЗТ), являющейся, в отличие от компьютерной (КТ) и магнитореzonансной (МРТ) томографии, не инвазивным методом, позволяющим обследовать целевые объекты в трехмерном пространстве. Подходы УЗТ различаются по типам используемых волн — отраженных или проходящих, что достигается различной геометрией расположения источников и приемников ультразвуковых (УЗ) колебаний, а также методами реконструкции пространственных изображений из зарегистрированного волнового поля. В одном из развитых подходов трансмиссионной УЗТ на проходящих волнах [4] исследуемый объект располагается между источниками и приемниками, лежащими в одной плоскости, которая пошагово перемещается в перпендикулярном направлении. Такая геометрия расстановки датчиков накладывает ограничения на размер объекта.

В настоящей статье рассматривается алгоритм получения изображения трехмерной модельной среды, содержащей несколько объектов с различными акустическими свойствами, путем регистрации на ее поверхности отраженного волнового поля линейкой трансдюсеров и получения

наборов двухмерных срезов при ее перемещении в направлении перпендикулярном линии датчиков.

При распространении волн в гетерогенных упругих средах происходят различные эффекты, искажающие волновое поле. Это многократные отражения волн от границ раздела сред, рассеивание и затухание энергии в различных материалах, а также эффект дифракции волн на неоднородностях. В результате решения обратной задачи УЗТ эти эффекты минимизируются с целью получения итогового пространственного изображения внутренней структуры исследуемого объекта.

Большой практический интерес представляет эффективное решение обратной задачи в двухмерной постановке для отраженных волн при расположении линейки трансдюсеров на поверхности исследуемой среды с возможностью ее перемещения. Такая схема наблюдения не накладывает ограничений на размер объекта, являясь промежуточным этапом УЗТ, путем построения высокоразрешающих объемных УЗ-изображений набором двухмерных срезов.

В теоретических работах по сейсморазведке развито множество методов решения обратной задачи, идея большинства из которых заключалась в упрощении физической картины рассматриваемого явления распространения волн и использования [8] алгоритмов корректировки возникающих эффектов реверберации, дифракции и поглощения.



Устранение артефактов дифракции позволяет более точно определять форму объектов и идентифицировать мелкие объекты на ультразвуковых изображениях. Долгое время применялись, в основном, аппаратные методы устранения дифракционных эффектов. Теоретические подходы для разработки программных методов, основанные на расчете дифракционных искажений и их подавлении, исследованы в [14]. В основу этих методов легли идеи, использовавшиеся в задачах обработки сейсмических данных. В этой области наибольшее распространение получили различные методы сейсмической миграции, такие как миграция Кирхгофа, конечно-разностная миграция и миграция в Фурье-области [15]. Эти методы были популярны и активно развивались в то время, когда мощностей вычислительных систем не хватало для полного численного решения волнового уравнения.

Наиболее перспективный метод решения обратной задачи — метод волновой инверсии, основанный на решении прямой и обратной задач построения УЗ-изображений методом численного решения волнового уравнения, наиболее точно описывающего физическую картину явления. Эти методы ранее нашли применение в области инверсии сейсмических данных при решении задач сейсморазведки, являясь, однако, достаточно ресурсоемкими вследствие большого объема данных. Актуально исследование применимости этих методов в области построения УЗИ, где в силу повышенных требований к разрешающей способности объем данных соизмерим или может даже превосходить объемы трехмерных сейсмических данных.

В последние годы, в связи с быстрым развитием вычислительной техники, в частности, массивно-параллельных систем, таких как GPU-кластеры и суперкомпьютеры, все больше внимания стало уделяться возможностям применения сложных волновых методов как в сейсмике, так и в области ультразвуковых исследований.

Из аналитических трудов по сейсморазведке можно отметить [1, 9, 13], где прора-

ботана базовая теория, которая в дальнейшем легла в основу многочисленных исследований.

За последнее десятилетие существенного теоретического и практического продвижения в решении обратных задач с применением для УЗ медицинских приложений удалось достичь лаборатории Los Alamos в содружестве с коллективами из нескольких университетов США [7] и группе авторов из Московского государственного университета имени М.В. Ломоносова [16]. Большинство этих исследований посвящено тороидальной системе трансдьюсеров, позволяющей регистрировать полное волновое поле как проникающих, так и отраженных волн, но накладывающей ограничения на размер исследуемого объекта. В этих работах рассматривались варианты решения волнового уравнения разностными схемами во временной области [16] либо волновой инверсии в частотной области [7].

Из разнообразия рассматриваемых систем наблюдения можно отметить [4], где применялась модель двух параллельных плоскостей, реализованная пошаговым сканированием объекта двумя линейками трансдьюсеров, расположенных по разные стороны от него. Эта система принимает как проникающие, так и отраженные волны, но с потерями части волновой информации. В работе подробно рассмотрено влияние разных видов регуляризации на качество результирующих изображений.

В настоящей статье приводятся решения прямой и обратной задач построения ультразвуковых изображений методом инверсии поля отраженных волн во временной области для двухмерного случая.

Решение прямой задачи

Определим формально прямую задачу построения ультразвуковых изображений. Входные данные задачи:

- акустические параметры гетерогенной среды известны;

- линейка из N трансдьюсеров расположена на поверхности исследуемой области; форма исходного импульса задана.

Для каждого трансдьюсера s моделирование волнового процесса происходит следующим образом: трансдьюсер s , работая в режиме источника сигнала, излучает начальный импульс, после чего все N трансдьюсеров, включая s , работают в режиме приемника в течение некоторого времени T , принимая отраженные от среды волны. В результате каждого излучения получается ансамбль из N реализаций волнового процесса, в котором каждая i -я реализация представляет собой сигнал длительностью T , зарегистрированный одним приемником. Решением прямой задачи является набор из N таких ансамблей, рассчитанных для каждого из N трансдьюсеров.

В методах волновой инверсии решение прямой задачи осуществляется при помощи численного решения волнового дифференциального уравнения, имеющего вид:

$$\frac{1}{c^2(x)} \frac{\partial^2 p(x, t; x_s)}{\partial t^2} - \nabla^2 p(x, t; x_s) = s(x, t; x_s), \quad (1)$$

где x_s — положение источника импульса; $p(x, t; x_s)$ — амплитуда волны в точке x в момент времени t ; $c(x)$ — скорость звука в точке x ; $s(x, t; x_s)$ — первоначальный сигнал.

Существуют различные подходы к численному решению такого уравнения, рассмотрим самые распространенные из них.

Метод конечных разностей. Это дифференциальное уравнение относится к гиперболическому типу, поэтому для его численного решения на некоторой сетке удобно применять метод конечно-разностной аппроксимации с крестообразным или Т-образным шаблоном для аппроксимации производных второго порядка, используя известные соотношения для производных:

$$\frac{\partial^2 p}{\partial t^2} = \frac{p_j^{k+1} - 2p_j^k + p_j^{k-1}}{\tau^2} + O(\tau^2), \quad (2)$$

$$\frac{\partial^2 p}{\partial x^2} = \frac{p_{j+1}^k - 2p_j^k + p_{j-1}^k}{h^2} + O(h^2), \quad (3)$$

где (i, j) — узел сетки; τ и h — шаги сетки по оси времени и по пространственной оси.

Эта схема является явной, т. к. значения в узлах шаблона на нижних временных слоях должны быть известны, поэтому она более простая с вычислительной точки зрения.

Однако на практике, при реализации метода на тестовой 2D-модели были выявлены определенные недостатки: условия на устойчивость и сходимость метода требуют выбора очень маленького шага дискретизации по пространственной оси, около 10 точек на длину волны сигнала [11]. В результате задача моделирования распространения высокочастотных волн, таких как ультразвук, становится очень затратной с вычислительной точки зрения, что критично для реализации следующего важного этапа алгоритма обращения — решения обратной задачи, при котором моделирование прямой задачи происходит многократно и итеративно.

Псевдоспектральный метод. Рассмотрим альтернативные методы решения волнового уравнения во временной области, обладающие меньшей вычислительной сложностью, чем метод конечных разностей. Одним из самых популярных методов, используемых для решения подобной задачи на практике, является псевдоспектральный метод, активно применявшийся для моделирования акустических волн в задачах сейсморазведки [10].

Основная идея метода состоит в том, что вместо аппроксимации пространственных производных конечными разностями применяется метод Фурье, основанный на численном нахождении производных в спектральной области и использовании обратного преобразования Фурье. При этом производные по времени аппроксимируются конечной разностью. Для уменьшения количества точек по времени применяется метод k -пространств, поглощающие граничные условия реализуются методом PML (Perfect Matched Layer) [11]. Данный метод обладает заметным преимуществом по скорости работы.

Недостаток заключается в том, что метод Фурье, несмотря на свою скорость, допус-



кает заметные погрешности, выражающиеся в виде появления мелких осцилляций в пространстве (эффект Гиббса) в том случае, если в спектре исходного сигнала есть резкие перепады частот, превышающие частоту, поддерживаемую дискретной временной шкалой по теореме Котельникова–Найквиста [11]. Это накладывает определенные ограничения на среду моделирования и требует повышенного внимания к спектральной картине, а в некоторых ситуациях может быть необходимо применение оконных сглаживающих функций. В данной работе рассматриваемые сигналы почти полностью лежат в ограниченной полосе спектра, что избавляет от этих проблем и позволяет применять метод на практике.

Для использования метода в прикладных задачах в разных языках программирования существуют специальные библиотеки численного моделирования, например, библиотека k-Wave Toolbox для языка MATLAB.

Решение обратной задачи

Исходные данные обратной задачи построения ультразвуковых изображений – данные о волновом поле в результате решения прямой задачи: N ансамблей, каждый из которых состоит из N реализаций длительностью T каждая. Решением обратной задачи является восстановленное по исходному волновому полю пространственное распределение параметров неоднородной среды, через которую прошел излученный сигнал, в данном случае – значения скорости звука в каждой точке пространства.

Метод волновой инверсии во временной области. Основная идея метода волновой инверсии при решении обратной задачи состоит в использовании знаний о математической модели распространения волнового поля для численного моделирования физического процесса и нахождения тех параметров уравнения, которые позволяют наиболее точно смоделировать данные, полученные в результате физического эксперимента.

Задача волновой инверсии обычно формулируется в виде задачи минимизации некоторой целевой функции, характеризующей отклонение смоделированных данных от экспериментальных. В качестве целевой функции часто берут расстояние между данными, порожденное L_2 -нормой [4]:

$$E(c) = \frac{1}{2} \sum_s \sum_r \int_0^T [\delta p(x_r, t; x_s)]^2 dt, \quad (4)$$

$$\delta p(x_r, t; x_s) = p(x_r, t; x_s) - p_{obs}(x_r, t; x_s), \quad (5)$$

где s – номер источника сигнала; r – номер приемника сигнала; c – текущая модель скоростей; $p(x_r, t; x_s)$ и $p_{obs}(x_r, t; x_s)$ – модельные и реальные данные, полученные приемником r в момент времени t в результате эксперимента с источником s .

Задача для волнового уравнения решается численно. Для определения направления минимизации на каждой итерации вычисляется градиент функции E в аппроксимации $g^{(k)} \approx \nabla E(c^{(k)})$, после чего происходит сдвиг текущей модели скоростей по следующей формуле [3]:

$$c^{(k+1)} = c^{(k)} + \alpha^{(k)} p^{(k)}, \quad (6)$$

где $c^{(k)}$ – модель скоростей, построенная на k -й итерации; $\alpha^{(k)}$ – шаг минимизации; $p^{(k)}$ – направление минимизации, зависящее от оптимизационной стратегии, например, в методе градиентного спуска

$$p^{(k)} = -g^{(k)}, \quad (7)$$

а в методе сопряженных градиентов

$$p^{(k)} = -g^{(k)} + \beta^{(k)} p^{(k-1)}, \quad (8)$$

где $\beta^{(k)}$ – параметр, который может быть выбран по-разному, в зависимости от конкретизации метода.

Оба этих метода относятся к методам локальной оптимизации, т. е. в зависимости от выбора начального приближения модели скоростей $c^{(0)}$ алгоритм может сойтись к экстремуму, не являющемуся глобальным минимумом целевой функции, что особенно проявляется при решении нелинейных и невыпуклых задач, к кото-

рым относится и задача инверсии данных при помощи волнового уравнения.

Выполнение каждой итерации алгоритмов подразумевает вычисление двух основных параметров: вектора градиента и скалярного значения шага оптимизации.

Вычисление градиента целевой функции. Обычно для вычисления вектора частных производных используют конечно-разностные соотношения вида:

$$\frac{\partial f(x_1, \dots, x_n)}{\partial x_i} \approx \frac{f(x_1, \dots, x_i + h_i, \dots, x_n) - f(x_1, \dots, x_n)}{h_i}, \quad (9)$$

где h_i – шаг дискретизации для переменной x_i .

Однако для данной задачи такая схема является крайне неэффективной, т. к. для вычисления всего вектора градиента потребуется N полных решений прямой задачи, где N – количество переменных в модели скоростей.

Вместо этого в задачах такого масштаба для вычисления приближенного значения градиента может использоваться метод сопряженного состояния, в котором значение частной производной вычисляется по следующей формуле [1]:

$$\frac{\partial E}{\partial c(x_k)} = \frac{2}{c(x_k)^3} \times \sum_s \int_0^T \frac{\partial^2 p(x_k, t; x_s)}{\partial t^2} p^*(x_k, T-t; x_s) dt, \quad (10)$$

где $p^*(x_k, t; x_s)$ – решение прямой задачи для сопряженного уравнения:

$$\frac{1}{c^2(x)} \frac{\partial^2 p^*(x_k, T-t; x_s)}{\partial t^2} - \nabla^2 p^*(x_k, T-t; x_s) = \sum_g \delta p(x_g, t; x_s) \delta(x - x_g), \quad (11)$$

$$\delta p(x_g, t; x_s) = p(x_g, t; x_s) - p_{obs}(x_g, t; x_s). \quad (12)$$

Моделирование решения сопряженного уравнения происходит следующим образом: для каждого источника s на основе уже имеющихся данных от каждого приемника g , посчитанных при решении ос-

новной прямой задачи, вычисляется разность $\delta p(x_g, t; x_s)$, после чего происходит ее обращение во временной оси. Полученные данные трактуются как синхронный импульс для каждого трансдюсера, и для решения сопряженного волнового уравнения моделирование процесса производится с использованием уже готовой реализации.

Данный метод обладает заметным преимуществом по вычислительной производительности, т. к. требует только одного дополнительного решения прямой задачи для вычисления градиента. Недостатком, как видно из формулы, является необходимость моделирования и запоминания не только сенсорных данных, но и полного волнового поля решений основного и сопряженного уравнения, что существенно увеличивает объем сохраняемых в памяти данных.

Для упрощения дальнейшего вычисления шага минимизации можно считать градиент нормированным; направление от этого не меняется, а алгоритм вычисления шага становится более универсальным [2]. Формулы для вычисления направления на k -й итерации принимают следующий вид:

$$p^{(k)} = -\frac{g^{(k)}}{\|g^{(k)}\|^2} + \beta^{(k)} p^{(k-1)}, \quad (13)$$

где $g^{(k)}$ – градиент, $\beta^{(k)}$ – коэффициент, который может принимать разные значения, например [3]

$$\beta_{FR}^{(k)} = \frac{\|g^{(k)}\|^2}{\|g^{(k-1)}\|^2} \quad (14)$$

в методе Флетчера-Ривза, или

$$\beta_{PRB}^{(k)} = \frac{(g^{(k)})^T (g^{(k)} - g^{(k-1)})}{\|g^{(k-1)}\|^2} \quad (15)$$

в методе Полака-Рибьера.

Вычисление шага минимизации. После вычисления направления минимизации необходимо оценить размеры шага α , который нужно сделать в этом направлении для получения новой модели скоростей. Оценка вычисляемого шага обычно произ-

водится при помощи проверки выполнения определенных условий, гарантирующих, что шаг не будет слишком большим или маленьким. На практике часто используют условия Armijo-Goldstein [2]:

$$E(c + \alpha p) \leq E(c) + \beta \alpha p, \quad (16)$$

где $\beta \in (0,1)$, обычно берут значения порядка 10^{-4} – 10^{-3} .

Алгоритм вычисления шага выглядит следующим образом: вначале берется некоторый шаг $\alpha_{\max}$, после чего происходит проверка условий, в случае их невыполнения берется новый шаг $\tau \alpha$, где τ – коэффициент от 0 до 1, после чего алгоритм повторяется до тех пор, пока условия не будут выполнены.

Выполнение этого шага обычно требует дополнительного решения 1-2 прямых задач. Это увеличивает вычислительную сложность каждой итерации, но улучшает их точность и уменьшает их количество. Соблюдение этого баланса в целом – главная проблема при решении оптимизационных задач.

Данный алгоритм гарантирует только то, что шаг не будет слишком большим. Для более точных оценок при оптимизации часто используют более жесткие условия Вольфе [2]:

$$E(c + \alpha p) \leq E(c) + c_1 \alpha \nabla E^T p, \quad (17)$$

$$|\nabla E(c + \alpha p)^T p| \leq c_2 |\nabla E^T p|, \quad (18)$$

где $c_1 \in (0,1)$, $c_2 \in (c_1, 1)$.

Однако выполнение этих условий требует вычисления градиента на каждом этапе, что очень сильно усложняет процесс вычислений, поэтому в данной работе было решено использовать условия первого типа.

TV-регуляризация. Одним из способов преодоления неустойчивости оптимизационных задач является регуляризация – добавление некоторого дополнительного слагаемого к условию функции ошибки. Чаще всего регуляризация представляет собой некоторую функцию, зависящую от каких-то априорных данных о модели, в данном случае – изначальном распределении скоростей.

Таким образом, регуляризованная функция ошибки имеет следующий общий вид:

$$E_R(c) = E(c) + \lambda R(c), \quad (19)$$

где λ – коэффициент регуляризации, определяющий степень влияния регуляризующей функции на результат. Значение коэффициента определяется эмпирически для конкретной задачи. В качестве регуляризующей функции часто берут, например, L_1 или L_2 норму (регуляризация Тихонова). Для обращения волновых данных важно, чтобы регуляризующая функция, помимо прочего, сохраняла края объектов на изображении для большей разрешающей способности. Поэтому при построении томографических изображений часто используют TV-регуляризацию (total variation), функция которой имеет следующий вид [5]:

$$R_{TV}(c(x)) = \sum_x \sqrt{\varepsilon^2 + \|\nabla c(x)\|^2}, \quad (20)$$

где x – пространственная координата; $c(x)$ – распределение скорости; ε – малое слагаемое, обеспечивающее дифференцируемость при значении нормы, равной нулю во всей области.

Выражение для вычисления градиента этой функции выглядит следующим образом [6]:

$$\nabla R_{TV}(c(x)) = \nabla \cdot \frac{\nabla c}{\sqrt{\varepsilon^2 + \|\nabla c(x)\|^2}}. \quad (21)$$

Метод упорядоченных подмножеств. Как упоминалось ранее, вычисление градиента методом сопряженного состояния (adjoint-state method) осложнено необходимостью считывать и обрабатывать на каждой итерации огромные объемы данных, что при ограничениях на систему ввода-вывода существенно увеличивает время на т. н. «накладные расходы». Для ускорения процесса обращения данных в задачах томографии можно использовать метод упорядоченных подмножеств (ordered-subsets method) [5]. Метод состоит в том, что данные сенсоров делятся на несколько наборов, каждый из которых содержит опреде-

ленное число срезов (под срезом здесь подразумевается набор данных, полученных от одного источника). Для каждого такого набора производится шаг итерации, минимизирующий некоторую локальную функцию ошибки, определенную отдельно для каждого набора k . Для того чтобы минимизирующий алгоритм работал корректно для небольших подмножеств, функцию ошибки было решено нормализовать относительно реальных данных срезов данного подмножества. Таким образом, из формулы (4) для определенного далее нормализующего коэффициента получаем формулу:

$$E(c) = \frac{1}{2K(c)} \sum_{s \in M_k} \sum_r \int_0^T [\delta p(x_r, t; x_s)]^2 dt, \quad (22)$$

$$\delta p(x_r, t; x_s) = p(x_r, t; x_s) - p_{obs}(x_r, t; x_s), \quad (23)$$

$$K(c) = \sum_{s \in M_k} \sum_r \int_0^T (p_{obs}(x_r, t; x_s))^2 dt, \quad (24)$$

где индексы суммирования s и r обозначают, соответственно, номера источников

и приемников сигнала; M_k — индексное множество всех источников, принадлежащих k -му набору; c — текущая модель скоростей; $p(x_r, t; x_s)$ и $p_{obs}(x_r, t; x_s)$ — модельные и реальные данные, полученные приемником r в момент времени t в результате эксперимента с источником s ; $K(c)$ — функция нормализации ошибки.

Таким образом, каждая итерация требует работы со значительно меньшим объемом данных, это увеличивает количество шагов, сделанных при анализе полного набора, что существенно ускоряет процесс без особого ухудшения качества итогового изображения [5].

Программная реализация

Общая архитектура программы. Для программной реализации рассмотренных ранее методов был выбран язык MATLAB. На рис. 1 представлена UML-диаграмма основных классов, реализующих необходимые структуры данных и алгоритмы решения прямой и обратной задач.

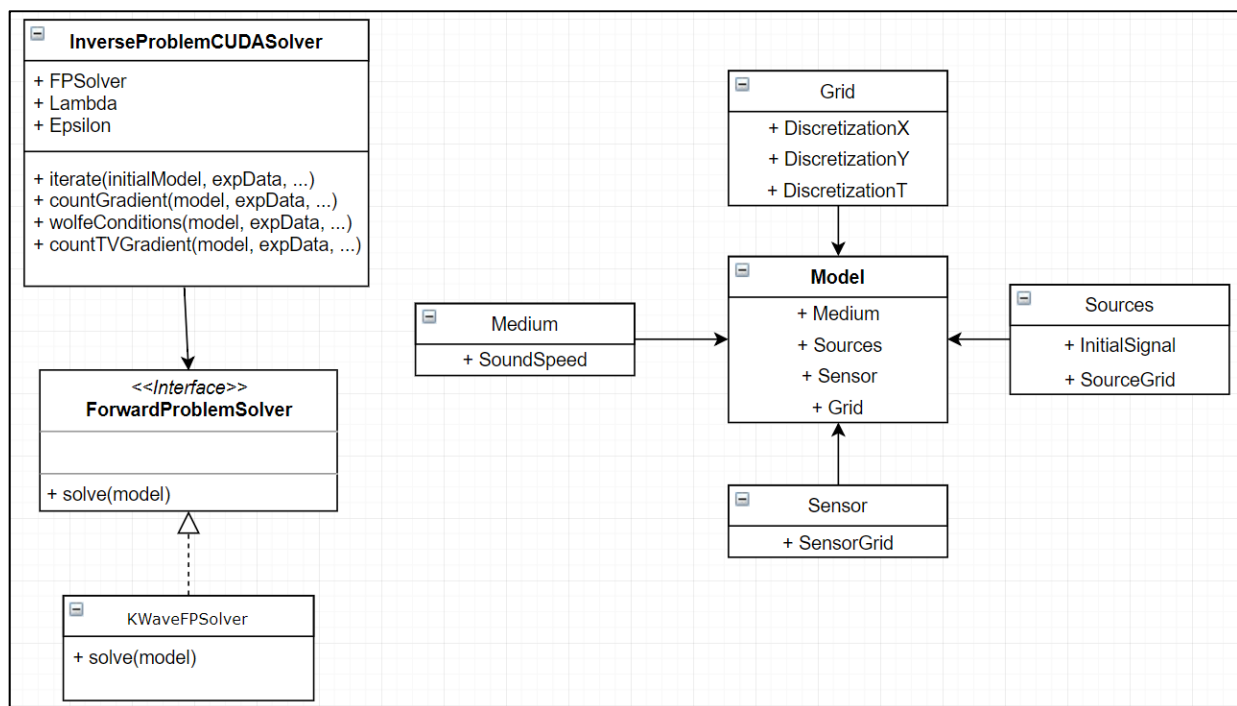


Рис. 1. Диаграмма взаимодействия основных классов проекта

Fig. 1. Diagram of project main classes interaction

Представленные на рисунке основные классы взаимодействуют следующим образом:

- **KWaveFPSolver** — реализует алгоритм решения прямой задачи для псевдоспектрального метода. Класс реализует интерфейс **ForwardProblemSolver**, содержащий метод **solve**, принимающий объект класса **Model** и возвращающий решение прямой задачи;

- **InverseProblemCUDASolver** — класс, реализующий весь необходимый функционал для решения обратной задачи: вычисление градиента, регуляризация, вычисление шага минимизации и т. д. При создании объекта в конструктор передается объект класса **KWaveFPSolver**, используемый далее в методах для решения прямой задачи, а также параметры регуляризации **Lambda** и **Epsilon**;

- **Model** — класс, агрегирующий в себе объекты, описывающие все основные параметры моделируемого эксперимента. Объекты этого класса передаются в качестве параметра всем методам, работающим с экспериментальной моделью. Полями объекта класса **Model** являются объекты классов **Medium**, **Sensor**, **Sources**, **Grid**. Это позволяет легко модифицировать модель при реализации разных алгоритмов;

- **Medium** — класс параметров среды, содержит поля значений скорости звука и плотности в каждой точке;

- **Sensor** — класс, задающий расположение приемников сигнала относительно исследуемой области;

- **Sources** — класс параметров источника сигнала, задает расположение источников и форму начального сигнала для каждого источника;

- **Grid** — класс параметров дискретизации модели, задает шаг дискретизации и количество отсчетов по осям **X**, **Y** и **T**.

Для моделирования волнового процесса при решении прямой задачи использовалась библиотека **k-Wave Toolbox**, неоднократно применявшаяся в научных исследованиях [10].

Распараллеливание вычислений на CPU и GPU. Последовательные реализации алгоритмов решения прямой и обратной задачи являются вычислительно затратными: этапы нахождения направления и шага оптимизации суммарно требуют дополнительного решения как минимум 5–6 прямых задач. Для моделируемой области средних размеров (около 200×300 дискретных значений) с более чем 100 трансдьюсерами, решение одной прямой задачи на типичном персональном компьютере (4 ядра, 2.6 ГГц, 8 Гб RAM, 500 Гб HDD), использованном для тестирования, занимает более получаса.

Кроме того, на этапе вычисления градиента методом сопряженного состояния требуется сохранять значения амплитуд не только в местах расположения трансдьюсеров, но и во всей остальной области в каждый момент времени. Объем обрабатываемых на каждой итерации данных для упомянутой модели составляет примерно 120 Гб. Данные такого объема невозможно хранить в оперативной памяти, поэтому при выполнении необходимых вычислений происходит нагруженный дисковый обмен. Суммарное время операций чтения, обработки и записи для одной итерации для HDD дисков составляет примерно 1,5 ч.

Вследствие высокой ресурсоемкости процесса моделирования становится актуальной задача оптимизации процесса решения прямой задачи с использованием технологий распараллеливания вычислений на CPU и GPU. Библиотека **k-Wave**, используемая для численного решения волнового уравнения при моделировании прямой задачи, содержит отдельные модули, предназначенные для более эффективного моделирования волнового процесса в 3D-областях большого масштаба. Эти модули написаны на языке программирования C++ с применением технологий **OpenMP** и **CUDA** и позволяют эффективно использовать вычислительные ресурсы компьютера.

Для возможности использования этих модулей в решении прямой задачи построения ультразвуковых 2D-изображений, двухмерная задача была представлена в ви-

де трехмерной с пренебрежительно малым шагом в третьем измерении. Результат моделирования для такого представления оказался абсолютно соответствующим обычной 2D-реализации, что подтверждает корректность такого использования модулей. Таким образом, в работе заложена возможность расширения предлагаемого двухмерного подхода до трех пространственных переменных в продолжение работы [8] для решения задачи получения томографических ультразвуковых изображений. Также для распараллеливания процесса решения уравнений для разных источников и чтения многочисленных файлов сенсорных данных использовался MATLAB Parallel Toolbox, позволяющий распределить нагрузку по вычислению данных каждого источника между несколькими worker-процессами при помощи операций выделения пула и распараллеливания циклов for.

В результате такой оптимизации время последовательного вычисления одной полной итерации для рассмотренной далее тестовой модели уменьшилось с 3,5 до 1,5 ч при использовании пула worker-процессов и до 27 мин при подключении библиотеки, использующей GPU. Распараллеливание проводилось с применением CPU Intel Core i5-6200U (4 ядра, 2.6 ГГц) и GPU NVIDIA GeForce GTX 950M (640 CUDA-ядер, 914 МГц).

Тестирование программы и анализ результатов

Тестирование минимизации функционала.

Проверим, что оптимизирующий алгоритм работает корректно, т. е. значение функции ошибки убывает с каждой итерацией.

Многократное осуществление итераций занимает очень много времени для крупных моделей, поэтому тестирование проводилось на небольшой, но структурно представительной модели, приведенной на рис. 2. Количество дискретных значений в пространстве $N_x = 256$, $N_y = 150$; шаг дискретизации в пространстве $dx = dy = 1 \cdot 10^{-4}$ м; дискретизация по времени $N_t = 1160$ с, $dt = 3 \cdot 10^{-8}$ с; скорость звука в среде составляет 1480 м/с, в центр помещен объект плотностью 1500 м/с. В объект добавлены вкрапления, скорость звука внутри которых равна, соответственно, 1570 и 1640 м/с. Трансдюсеры расположены вдоль оси X с шагом равным четным.

Возьмем в качестве начальной модели однородную среду плотностью 1480 м/с и запустим 45 итераций оптимизирующего алгоритма.

График зависимости функции ошибки от числа итераций изображен на рис. 3. По графику видно, что зависимость функции ошибки от количества итераций убывающая, что подтверждает корректность оптимизирующего алгоритма.

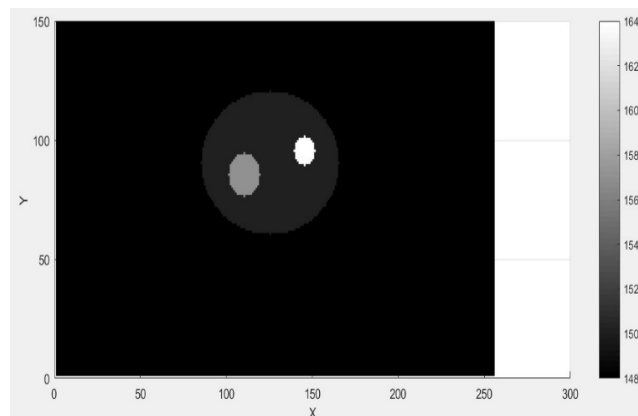


Рис. 2. Структура модели среды в виде пространственного распределения скорости звука в м/с, отображаемой в полутоновой шкале

Fig. 2. Model structure as the sound speed dimensional distribution represented in grayscale

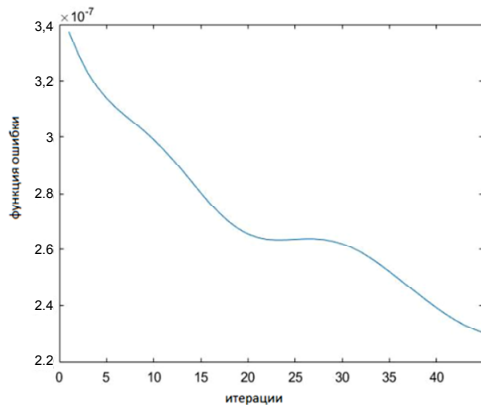


Рис. 3. Зависимость значения функции ошибки от числа итераций

Fig. 3. Error function dependency on number of iterations

Тестирование TV-регуляризации. Для проверки корректности TV-регуляризации и реализации ее градиента был запущен отдельно градиентный спуск для функции R_{TV} . Значение функции может вести себя по-разному в зависимости от параметра ϵ , поэтому было проведено два эксперимента с разными значениями ϵ .

Из рисунка 4 видно, что при очень малых значениях параметра ϵ градиент становится слишком чувствительным к величине шага, и процесс минимизации происходит некорректно. При достаточно большом ϵ алгоритм работает корректно.

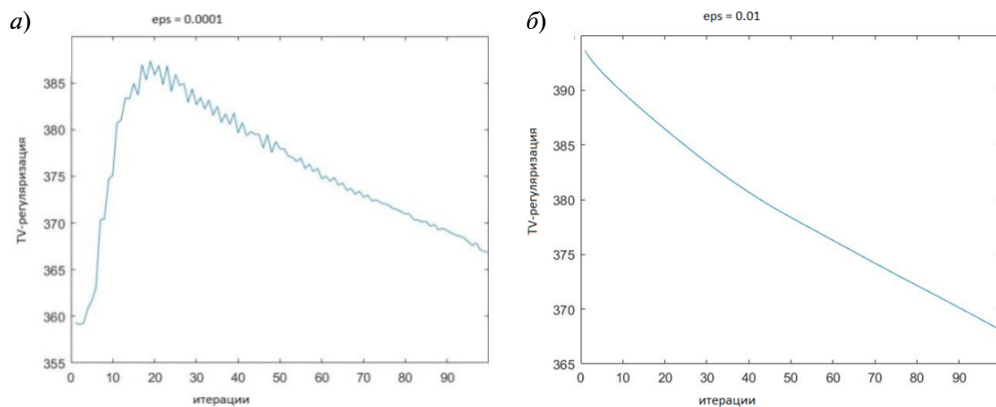


Рис. 4. Зависимость значения функции регуляризации от числа итераций для малого (а) и большого (б) параметра ϵ

Fig. 4. Regularization function dependency on number of iterations for small (a) and large (b) value of ϵ

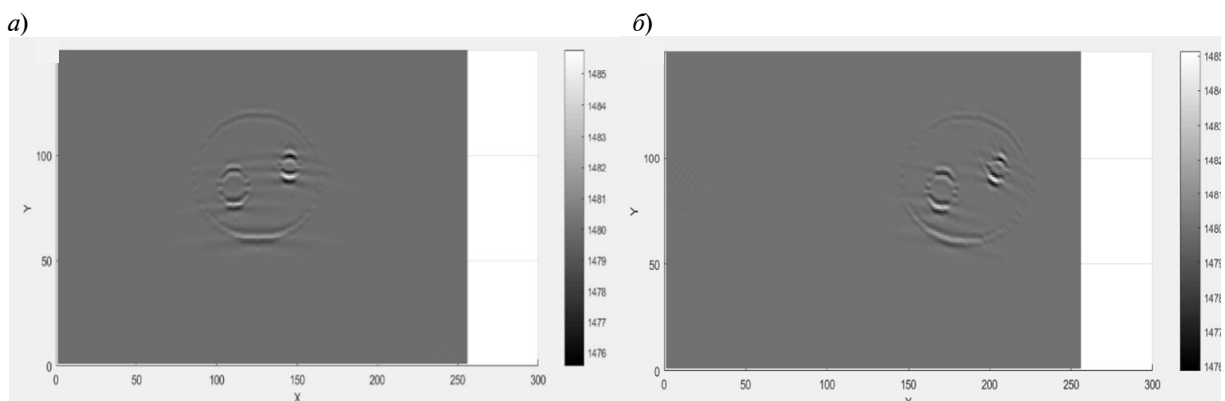


Рис. 5. Восстановленное пространственное распределение скоростей звука для исходной модели (а) и со сдвинутыми вправо по оси X внутренними объектами (б)

Fig. 5. Restored sound speed spatial distribution for initial model (a) and with internal objects shifted to the right along the X axis (b)

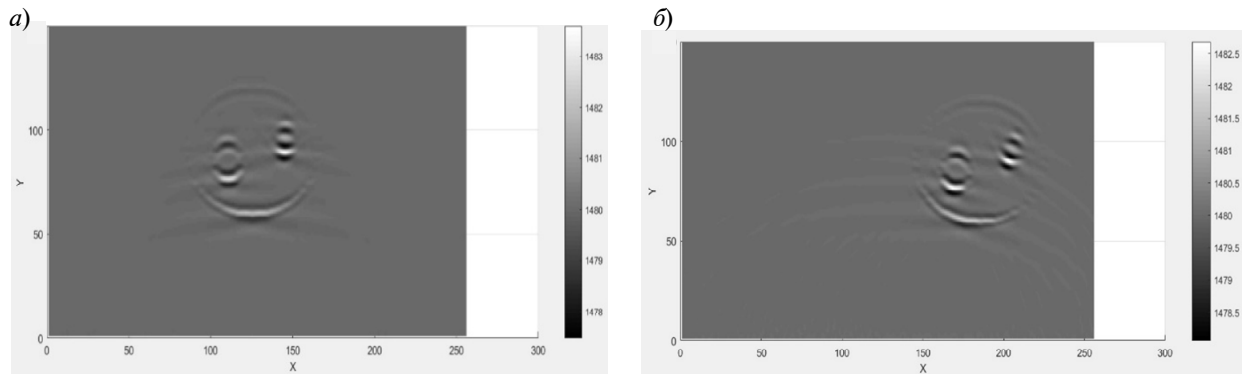


Рис. 6. Восстановленное пространственное распределение скоростей звука для исходной модели с поглощением (а) и со сдвинутыми вправо по оси X внутренними объектами (б)

Fig. 6. Restored sound speed spatial distribution for initial model with acoustic absorption (a) and with internal objects shifted to the right along the X axis (b)

Тестирование разрешающей способности алгоритма. Разрешающая способность алгоритма в первую очередь зависит от выбираемого им направления оптимизации.

Некоторые детали могут быть видны после первой итерации, а какие-то так и не проявляются. Для изучения этого вопроса рассмотрим несколько небольших представительных моделей и построим изображения восстановленного пространственного распределения скоростей звука после выполнения нескольких итераций.

Для начала возьмем модель, рассмотренную ранее. Из рисунка 5 видно, что четко выделяются передняя и задняя границы объекта среды и мелких вкраплений, при этом плохо прорисованы боковые участки объекта.

Это может быть связано с тем, что волны, отражающиеся от боковых субвертикальных участков границ, слабо улавливаются сенсорами, т. к. объект расположен строго в центре. Рассмотрим случай объекта, сдвинутого относительно линии сенсоров, переместив его на 60 единиц вправо по оси X.

На рис. 5 б видно, что небольшой сдвиг никак не повлиял на степень детализации боковых участков, из чего можно сделать вывод, что длина линии сенсоров должна значительно превосходить размеры предмета для его качественного отображения.

Теперь изучим влияние параметра поглощения среды на качество изображения

объекта. Будем считать, что поглощение происходит по экспоненциальному закону, зависящему от частоты сигнала [10].

На рис. 6 видно, что восстановленное изображение получается более размытым, менее четко прорисованы задние участки границ, т. к. часть волновой информации поглощается. Кроме того, сильнее проявляются поперечные артефакты в виде полос, особенно это видно по изображению со сдвинутым объектом. Такие артефакты могут быть подавлены дальнейшей обработкой полученных изображений и последующим применением алгоритмов сегментации.

Заключение

Предложен оригинальный алгоритм решения прямой и обратной задач построения плоских ультразвуковых изображений методом волновой инверсии отраженного поля во временной области. Алгоритм обладает рядом преимуществ по сравнению с другими алгоритмами, упомянутыми выше. Во-первых, предложенный алгоритм вычислительно более устойчив при наличии шумов, чем алгоритмы, использующие конечно-разностные схемы. Во-вторых, алгоритм работает только с отраженной частью волнового поля, регистрируемой на поверхности среды, что снимает ограничения на размеры изучаемых объектов. Это преимущество позволяет расширить область применения предлагаемого метода не



только для медицинских приложений, но и для задач неразрушающего контроля. Объемное изображение может быть представлено набором двумерных срезов. Недостатком подхода на отраженных волнах может быть отсутствие отражений при больших углах падения волн, что налагает определенные требования к размерам и геометрии системы наблюдения. Разработанный алгоритм программно реализован с распараллеливанием вычислений, что примерно на порядок повысило производительность процесса моделирования.

Тестирование алгоритма проведено на упрощенных, но акустически представительных моделях. Упрощение пространственной модели среды, состоящей из не-

скольких объектов относительно простой формы, принципиально не влияет на решение задачи и было обусловлено только вычислительными мощностями тестового компьютера.

Дальнейший интерес представляет решение задачи в трехмерной постановке для получения полномасштабных высокоразрешающих томографических ультразвуковых изображений сложно устроенных модельных сред. Прямые и обратные задачи в такой постановке необходимо алгоритмизировать и решать на высокопроизводительных параллельных системах, таких как суперкомпьютер или GPU-кластер, вследствие существенного роста объема данных и вычислений.

СПИСОК ЛИТЕРАТУРЫ

1. Boonyasiriwat, P. Valasek, P. Routh, W., et al. An efficient multiscale method for time-domain waveform tomography // *Geophysics*. 2009. Pp. WCC59–WCC68.
2. Nocedal J., Wright S. Numerical optimization. Springer, 2000. Pp. 30–63.
3. Liu Y., Teng J., Xu T., Badal J., Liu Q., Zhou B. Effects of conjugate gradient methods and step-length formulas on the multiscale full waveform inversion in time domain: numerical experiments // *Pure and Applied Geophysics*. 2017. Pp. 1983–2006.
4. Lin Y., Huang L., Zhang Z. Ultrasound waveform tomography with the total-variation regularization for detection of small breast tumors // *SPIE Medical Imaging*. 2012. Pp. 832002-1–832002-9.
5. Hudson H.M., Larkin R.S. Accelerated image reconstruction using ordered subsets of projection data // *IEEE Trans. Med. Imaging*. 1994. Pp. 100–108.
6. Peyré G. The numerical tours of signal processing – advanced computational signal and image processing // *IEEE Computing in Science and Engineering*. 2011. Pp. 94–97.
7. Sandhu G.Y., Li C., Roy O., Schmidt S., Duric N. Frequency domain ultrasound waveform tomography: breast imaging using a ring transducer // *Physics in Medicine and Biology*. 2015. Vol. 60. No. 14. Pp. 5381–5398.
8. Belykh I., Markov D. Volumetric ultrasound imaging: modeling of waveform inversion // *Computational Science Research Notes* 2703. 2017. Pp. 77–82.
9. Virieux J., Operto S. An overview of full waveform inversion in exploration geophysics // *Geophysics*. 2009. Pp. WCC1–WCC26.
10. Treeby B.E., Jaros J., Rendell A.P., Cox B.T. Modeling nonlinear ultrasound propagation in heterogeneous media with power law absorption using a k-space pseudospectral method // *The Journal of the Acoustical Society of America*. 2012. Pp. 4324–4336.
11. Langtangen P.H., Linge S. Finite difference computing with PDEs // *Texts in Computational Science and Engineering*. 2017. Pp. 93–205.
12. Li C., Duric N., Huang L. Clinical breast imaging using sound-speed reconstructions of ultrasound tomography data // *Medical Imaging Int. Society for Optics and Photonics*. 2008. Pp. 692009-1–692009-9.
13. Pratt R.G. Seismic waveform inversion in the frequency domain. Part 1: Theory and verification in a physical scale model // *Geophysics*. 1999. Vol. 64. Pp. 888–901.
14. Huang L., Williamson M. Detection of breast microcalcifications with super-resolution ultrasound imaging: A clinical study // *SPIE Medical Imaging*. 2013. Pp. 86751O-1–86751O-10.
15. Li J. Kirchhoff common offset migration velocity analysis via differential semblance // *Master's Thesis, Rice University*. 2007. Pp. 1–61.
16. Гончарский А.В., Романов С.Ю., Серёжников С.Ю. Обратные задачи послойной ультразвуковой томографии с данными на цилиндрической поверхности // *Вычислительные методы и программирование*. 2017. Т. 18. С. 267–276.

Статья поступила в редакцию 09.07.2018.

REFERENCES

1. Boonyasiriwat, P. Valasek, P. Routh, W., et al. An efficient multiscale method for time-domain waveform tomography. *Geophysics*, 2009, Pp. WCC59–WCC68.
2. Nocedal J., Wright S. *Numerical optimization*. Springer, 2000, Pp. 30–63.
3. Liu Y., Teng J., Xu T., Badal J., Liu Q., Zhou B. Effects of conjugate gradient methods and step-length formulas on the multiscale full waveform inversion in time domain: numerical experiments. *Pure and Applied Geophysics*, 2017, Pp. 1983–2006.
4. Lin Y., Huang L., Zhang Z. Ultrasound waveform tomography with the total-variation regularization for detection of small breast tumors. *SPIE Medical Imaging*, 2012, Pp. 832002-1–832002-9.
5. Hudson H.M., Larkin R.S. Accelerated image reconstruction using ordered subsets of projection data. *IEEE Trans. Med. Imaging*, 1994, Pp. 100–108.
6. Peyré G. The numerical tours of signal processing – advanced computational signal and image processing. *IEEE Computing in Science and Engineering*, 2011, Pp. 94–97.
7. Sandhu G.Y., Li C., Roy O., Schmidt S., Duric N. Frequency domain ultrasound waveform tomography: breast imaging using a ring transducer. *Physics in Medicine and Biology*, 2015, Vol. 60, No. 14, Pp. 5381–5398.
8. Belykh I., Markov D. Volumetric ultrasound imaging: modeling of waveform inversion. *Computer Science Research Notes* 2703, 2017, Pp. 77–82.
9. Virieux J., Operto S. An overview of full waveform inversion in exploration geophysics. *Geophysics*, 2009, Pp. WCC1–WCC26.
10. Treeby B.E., Jaros J., Rendell A.P., Cox B.T. Modeling nonlinear ultrasound propagation in heterogeneous media with power law absorption using a k-space pseudospectral method. *The Journal of the Acoustical Society of America*, 2012, Pp. 4324–4336.
11. Langtangen P.H., Linge S. Finite difference computing with PDEs. *Texts in Computational Science and Engineering*, 2017, Pp. 93–205.
12. Li C., Duric N., Huang L. Clinical breast imaging using sound-speed reconstructions of ultrasound tomography data. *Medical Imaging Int. Society for Optics and Photonics*, 2008, Pp. 692009-1–692009-9.
13. Pratt R.G. Seismic waveform inversion in the frequency domain. Part 1: Theory and verification in a physical scale model. *Geophysics*, 1999, Vol. 64, Pp. 888–901.
14. Huang L., Williamson M. Detection of breast microcalcifications with super-resolution ultrasound imaging: A clinical study. *SPIE Medical Imaging*, 2013, Pp. 86751O-1–86751O-10.
15. Li J. Kirchhoff common offset migration velocity analysis via differential semblance. *Master's Thesis, Rice University*, 2007, Pp. 1–61.
16. Goncharskiy A.V., Romanov S.Y., Seryozhnikov S.Y. Obratnyye zadachi posloynoy ultrazvukovoy tomografii s dannymi na tsilindricheskoy poverkhnosti [Inverse problems of layer-by-layer ultrasonic tomography with the data measured on a cylindrical surface]. *Vychislitelnyye metody i programmirovaniye [Numerical methods and programming]*, 2017, Vol. 18, Pp. 267–276. (rus)

Received 09.07.2018.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

ВЕРЕЩАГИН Константин Алексеевич
VERESHCHAGIN Konstantin A.
 E-mail: kostya.veresh@mail.ru

БЕЛЫХ Игорь Николаевич
BELYKH Igor N.
 E-mail: ibelykh.spb@gmail.com



DOI: 10.18721/JCSTCS.11405
УДК 004.9: 621.4

АЛГОРИТМ ДИАГНОСТИКИ ТЕХНИЧЕСКОГО СОСТОЯНИЯ ДВИГАТЕЛЯ ПО ПАРАМЕТРАМ ЧАСТОТНЫХ ХАРАКТЕРИСТИК МАСЛЯНОГО ФИЛЬТРА

Ю.Д. Шевцов¹, Л.Н. Дудник¹, Ю.В. Чернуха², Е.Д. Фадеев¹, А.М. Никифоров¹

¹ Кубанский государственный технологический университет,
г. Краснодар, Российская Федерация;

² Краснодарское высшее военное училище имени генерала армии С.М. Штеменко,
г. Краснодар, Российская Федерация

Получены и представлены амплитудные и фазовые частотные характеристики нормально работающего двигателя и с учетом возникновения возможных неисправностей в двигателе. Проведены эксперименты с использованием фильтроэлементов, проработавших в системе смазки различное количество времени. Произведена аппроксимация исходных графиков методом суммы синусов. Предложен алгоритм для проведения диагностики и оценки технического состояния фильтра и технического состояния двигателя. Наличие моделей фильтра, представленных в виде передаточных функций, позволяет, включив их в систему управления двигателем, изменять параметры системы смазки с целью улучшения эффективности его работы.

Ключевые слова: моделирование системы, техническое состояние, состояние фильтра, осциллограммы, управление двигателем, аппроксимация.

Ссылка при цитировании: Шевцов Ю.Д., Дудник Л.Н., Чернуха Ю.В., Фадеев Е.Д., Никифоров А.М. Алгоритм диагностики технического состояния двигателя по параметрам частотных характеристик масляного фильтра // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2018. Т. 11. № 4. С. 63–70. DOI: 10.18721/JCSTCS.11405

ALGORITHM FOR DIAGNOSING ENGINE HEALTH BY PARAMETERS OF FREQUENCY CHARACTERISTICS OF OIL FILTER

Yu.D. Shevtsov¹, L.N. Dudnik¹, Yu.V. Chernukha², E.D. Fadeev¹, A.M. Nikiforov¹

¹ Kuban State Technological University,
Krasnodar, Russian Federation;

² Krasnodar Higher Military School named after General of the Army S.M. Shtemenko,
Krasnodar, Russian Federation

We have obtained and discussed amplitude and phase frequency characteristics of a normally running engine and taking into account possible malfunctions in the engine based on a series of experiments using filter elements working in the lubrication system for different periods of time. Approximation of the initial schedules has been carried out by the sine sum method. We have offered an algorithm for performing diagnostics and assessment of the filter's and the engine's health. Including filter models in the form of transfer functions into the engine's control system allows to change the lubrication system parameters, improving the engine's efficiency.

Keywords: system modeling, technical status, filter health, oscillogram, engine management, approximation.

Citation: Shevtsov Yu.D., Dudnik L.N., Chernukha Yu.V., Fadeev E.D., Nikiforov A.M. Algorithm for diagnosing engine health by parameters of frequency characteristics of oil filter. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2018, Vol. 11, No. 4, Pp. 63–70. DOI: 10.18721/JCSTCS.11405

Введение

Один из важнейших показателей, оценивающих и ограничивающих ресурс непрерывной работы двигателей автономных и резервных энергетически установок, — время необслуживаемой работы элементов системы смазки: ограниченная периодичность замены масляного фильтра и моторного масла*.

Уровень развития современных технологий позволяет создавать системы контроля загрязнения масляного фильтра и элементов системы смазки. Чаще всего такие системы основаны на сравнении показаний датчиков, установленных в ДВС, с эталонными моделями масляных фильтров [1]. Так как пропускная способность фильтра зависит от многих факторов, создание достоверной модели, чувствительной к изменению параметров масляного фильтра, является нетривиальной и очень актуальной задачей. Существует несколько способов ее решения.

К числу наиболее эффективных способов моделирования системы смазки и ее элементов относятся модели, основанные на частотных методах, при помощи которых модели, в частности фильтров, представляются в виде передаточных функций [3–5].

Такие модели позволяют не только эффективно оценивать техническое состояние масляных фильтров, а по ним и двигателей, но и строить систему управления по изменяющимся параметрам системы смазки [2, 6].

Таким образом, для получения постоянной информации о техническом состоянии фильтра и двигателя в целом необходимо установить датчики давления на входе и выходе масляного фильтра, а в бортовом компьютере установить программное обеспечение, позволяющее обрабатывать получаемые осциллограммы давления и сравнивать их с эталонными характеристиками. В процессе эксплуатации могут возникать критические ситуации с параметрами исследуемых характеристик, связанных с «масляным голодани-

ем» на некоторых резонансных частотах. Наличие моделей фильтра, представленных в виде передаточных функций, позволяет, включив их в систему управления двигателя, изменять параметры системы смазки с целью улучшения эффективности его работы.

Результаты проведенного эксперимента

Предлагаемый метод синтеза эталонной математической модели масляного фильтра по исходным данным предполагает наличие экспериментальных графиков, полученных на специальной установке, при помощи которой создавались пульсации давления масла на входе и выходе из фильтра [2]. Экспериментальные графики пульсаций масла в системе смазки двигателя К-770 на одной из частот представлены на рис. 1.

На вход фильтра подавалось давление $P_{\text{вх}}$ пропорционально колебанию

$$y(t) = A_0 \sin \omega_i t, \quad (1)$$

где A_0 — постоянная амплитуда входного сигнала; ω_i — круговая частота, изменяющаяся в процессе эксперимента, на выходе фильтра регистрировался сигнал $P_{\text{вых}}$, пропорциональный

$$x(t) = A_1 \sin(\omega_i t + \varphi_i), \quad (2)$$

где A_1 — амплитуда выходного сигнала; φ_i — сдвиг фаз между выходным и входным сигналами.

По результатам обработки полученных осциллограмм определены и построены амплитудные и фазовые частотные характеристики.

Передаточную функцию системы смазки можно получить аналитически, составляя уравнения по их схемам замещения в электрических аналогах [1]. Например, передаточная функция масляного фильтра дизельного двигателя К-770 по давлению, с подставленными в уравнения коэффициентами системы, будет выглядеть следующим образом:

$$W(s) = \frac{8,039 \cdot 10^{-6} s^2 + 1,054 \cdot 10^{-3} s + 0,9804}{3,393 \cdot 10^{-11} s^4 + 9,76 \cdot 10^{-9} s^3 + \dots + 1,764 \cdot 10^{-5} s^2 + 1,602 \cdot 10^{-3} s + 1} \dots$$

* Кузнецов Е.С., Болдин А.П., Власов В.М. Техническая эксплуатация автомобилей: Учебник для вузов. 4-е изд. М.: Наука, 2011. 535 с.

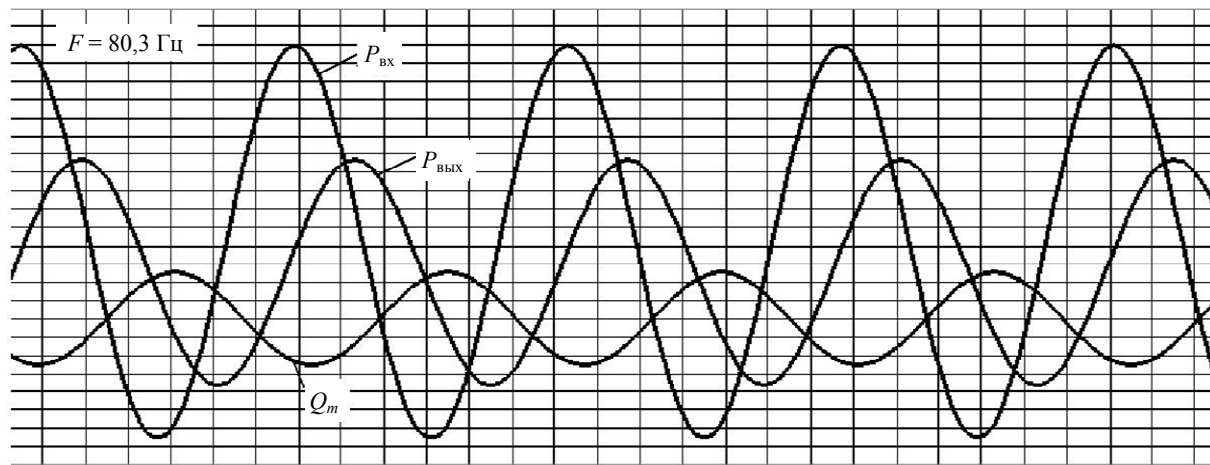


Рис. 1. Осциллограмма давлений масла на входе $P_{вх}$ и $P_{вых}$ фильтра и объемного расхода (подачи) Q_m на одной из частот ($f = 80$ Гц)

Fig. 1. Oscillogram of oil pressure at the inlet of the $P_{вх}$ and $P_{вых}$ filter and volumetric flow rate (supply) Q_m at one of the frequencies ($f = 80$ Гц)

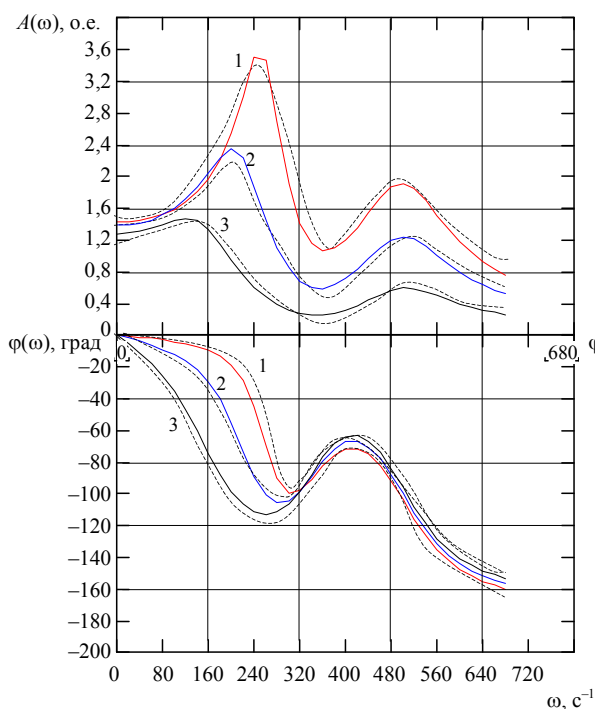


Рис. 2. АЧХ и ФЧХ масляного фильтра при различном времени эксплуатации:

1 – $t = 10$ ч; 2 – $t = 80$ ч; 3 – $t = 150$ ч;
(- -) теория; (—) эксперимент

Fig. 2. AFC and FCH of the oil filter at different operating times

В результате проведенной серии экспериментов с использованием фильтроэлементов, проработавших в системе смазки различное количество времени, получены и представле-

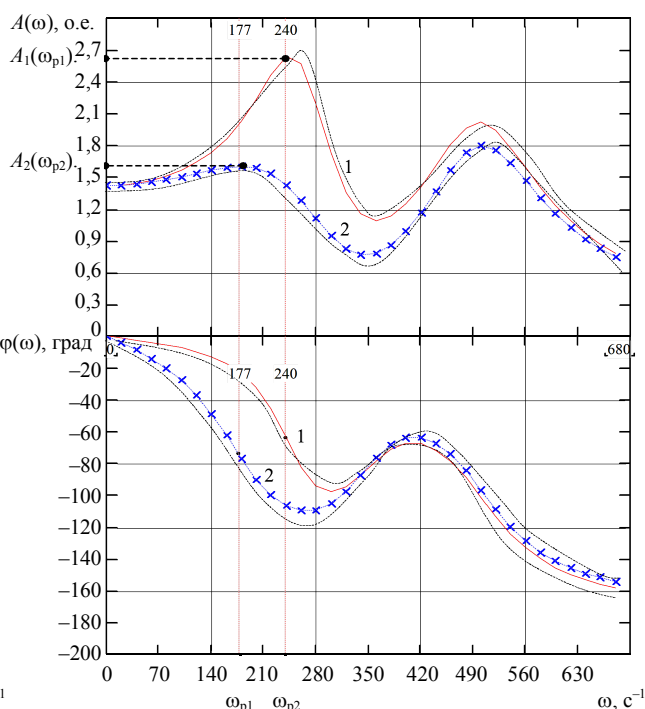


Рис. 3. АЧХ и ФЧХ масляного фильтра при $t = 35$ ч:

1 – при нормальной работе двигателя;
2 – при возникновении неисправности;

Fig. 3. AFC and FCH of the oil filter at $t = 35$ h

ны на рис. 2 и 3 амплитудные и фазовые частотные характеристики нормально работающего двигателя и с учетом возникновения возможных неисправностей в двигателе [7].

Другим способом при оценке технического состояния двигателя по параметрам его системы смазки является использование информации, получаемой в результате обработки осциллограмм давления масла, записываемых на входе и выходе фильтра при работающем двигателе (рис. 4).

С этой целью целесообразно применять разработанный способ оценки технического состояния двигателя внутреннего сгорания, согласно которому в качестве параметров пульсаций давления используются амплитудные и фазовые частотные характеристики, получаемые в заданном диапазоне частот в результате разложения в гармонический ряд несинусоидальных периодических сигналов на входе и выходе очистителя масла [2]:

$$P_{\text{вх}}(t) = \sum_{i=1}^n A_i \sin(\omega_i t + \varphi_i); \quad (3)$$

$$P_{\text{вых}}(t) = \sum_{k=1}^n A_k \sin(\omega_k t + \varphi_k), \quad (4)$$

где i, k – гармоники входного и выходного сигналов; A_i, A_k – амплитуды, фазы гармоник входного и выходного сигналов соответственно; n – количество делений среднестатистического периода пульсаций.

В качестве результата определяется величина и скорость отклонения по амплитуде и фазе данных характеристик от эталонных АЧХ и ФЧХ. Эталонными являются частотные характеристики, полученные в результате экспериментальных исследований при нормальной работе двигателя и при возникновении неисправностей в его узлах и системах [8, 9].

Для реализации этого способа предложено воспользоваться программой Matlab. Необходимо произвести аппроксимацию исходных графиков, заданных таблично, методом суммы синусов по семи гармоникам. Описывающее их уравнение представлено ниже:

$$f(x) = \sum_7^i a_i * \sin(b_i * x + c_i), \quad (5)$$

где a_i – амплитудное значение; b_i – круговая частота; c_i – фазы гармоник.

Используя программу Matlab, получаем графики пары вход-выход в электронном виде (рис. 4, 5).

Коэффициенты уравнения аппроксимации для входа по семи гармоникам представлены в табл. 1, для выхода – в табл. 2.

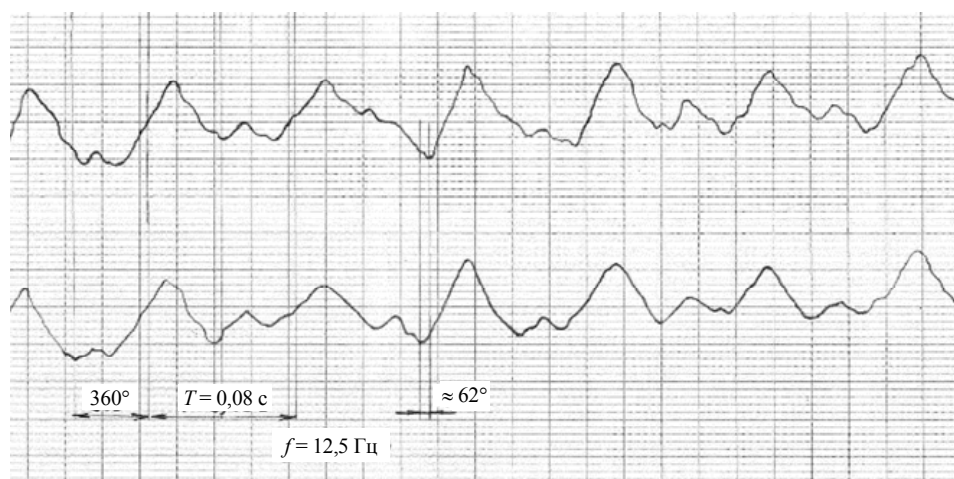


Рис. 4. Осциллограммы давления масла на входе и выходе масляного фильтра при работающем двигателе

Fig. 4. Oscillograms of oil pressure at the inlet and outlet of the oil filter with the engine running

Таблица 1

Коэффициенты уравнения аппроксимации для входа

Table 1

Approximation equation coefficients for an entrance

	1	2	3	4	5	6	7
a	0,5105	0,1109	0,07419	0,02054	0,02802	0,01286	0,007888
b	0,1502	0,3137	0,6411	1,571	1,26	0,9421	1,899
c	0,1036	0,8657	-1,857	1,498	-2,565	1,393	1,842

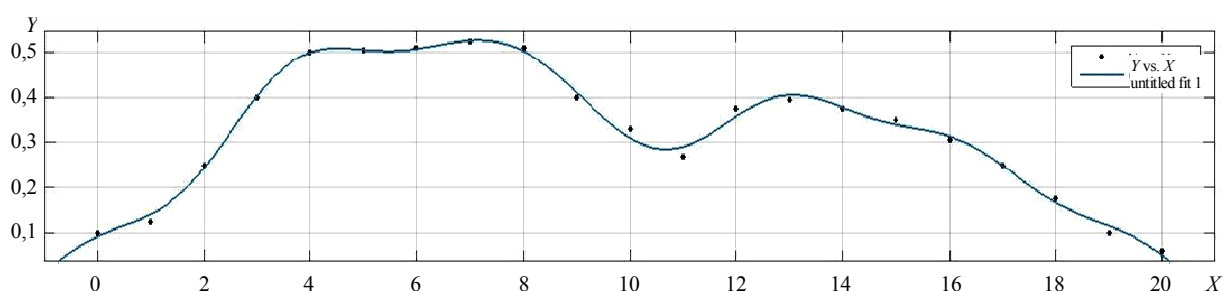


Рис. 4. График аппроксимированной функции на входе:

(•) заданные значения; (—) аппроксимированная функция

Fig. 4. Graph of the approximated function at the input (the specified values are indicated by dots, the approximated function is indicated by a line)

Таблица 2

Коэффициенты уравнения аппроксимации для выхода

Table 2

Approximation equation coefficients for an exit

	1	2	3	4	5	6	7
a_1	0,5503	0,04252	0,02434	0,1548	0,03185	0,01591	0,01019
b_1	0,166	0,6465	1,249	0,3475	0,8113	1,584	2,502
c_1	-0,0578	-2,712	2,299	0,02929	1,662	-0,04344	1,326

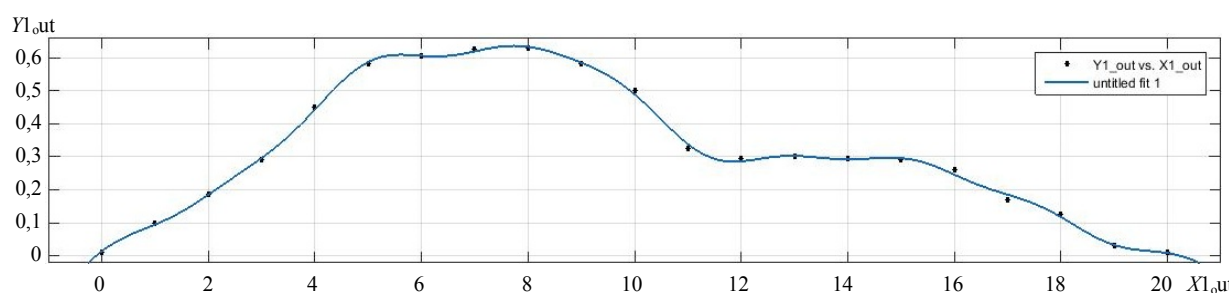


Рис. 5. График аппроксимированной функции на входе:

(•) заданные значения; (—) аппроксимированная функция

Fig. 5. Graph of the approximated function at the input (the given values are indicated by dots, the approximated function is indicated by a line)

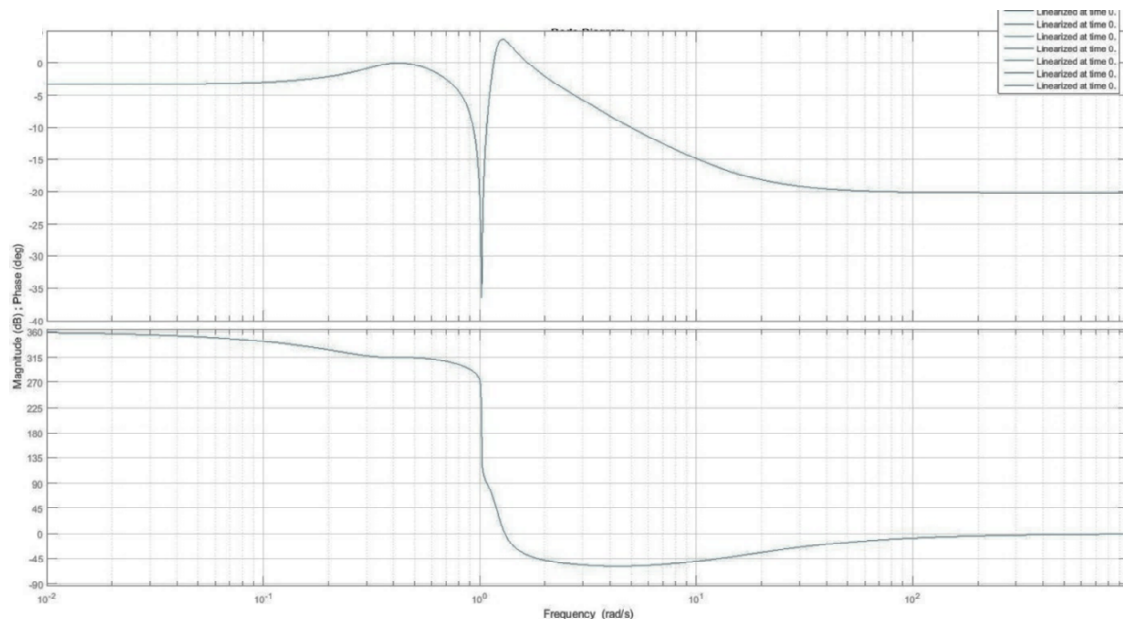


Рис. 6. АЧХ и ФЧХ передаточной функции масляного фильтра

Fig. 6. The AFC and FCH of the transfer function of the oil filter

На следующем этапе формулы, описывающие пульсации масла на входе и выходе из фильтра, преобразуются к виду изображений Лапласа. Отношение изображений Лапласа функций входа и выхода представляет собой математическую модель фильтра в виде передаточной функции:

$$W(s) = \frac{-25,1602 \cdot s^8 - 357,2744 \cdot s^7 + \dots}{-174,4378 \cdot s^8 - 1,6147 \cdot s^7 - \dots} + \frac{383,4296 \cdot s^6 - 597,3089 \cdot s^5 + 495,9023 \cdot s^4 - \dots}{-173,5009 \cdot s^6 + 10,5214 \cdot s^5 + 112,5220 \cdot s^4 - \dots} - \frac{244,6831 \cdot s^3 + 74,4339 \cdot s^2 - 13,0151 \cdot s - 1}{-98,5891 \cdot s^3 + 42,2884 \cdot s^2 - 9,8241 \cdot s + 1}.$$

АЧХ и ФЧХ передаточной функции представлены на рис. 6.

Из сравнения АЧХ и ФЧХ математических моделей, полученных методом схем замещения в электрических аналогах (рис. 4 и 5) и методом получения передаточной функции из аппроксимированных графиков (рис. 6), видно, что они существенно отличаются, поскольку первая получена для фильтра, а вторая учитывает все элементы системы смазки двигателя. Таким образом, передаточная функция всей системы равна $W_c(s) = W_\Phi(s) \cdot W_H(s)$.

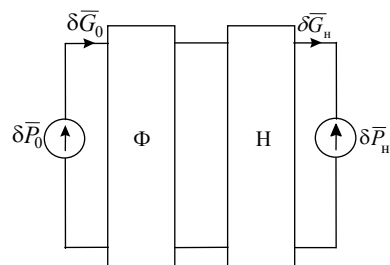


Рис. 7. Структурная схема передаточной функции системы смазки:

Φ – фильтр; Н – нагрузка

Fig. 7. Block diagram of the transfer function of the lubrication system

Она включает в себя передаточную функцию фильтра и передаточные функции остальных элементов системы смазки, представленных в виде передаточной функции нагрузки (рис. 7).

Заключение

Для проведения диагностики и оценки технического состояния фильтра необходимо последовательно реализовать следующий алгоритм.

1. Осциллографировать пульсации давления на входе и выходе фильтра для различных периодов эксплуатации нормального и аварийного функционирования двигателя.

2. Выделить из передаточной функции всей системы смазки $W_c(s)$ передаточную функцию фильтра.

3. Полученные из передаточной функции фильтра частотные характеристики сравнить с эталонными АЧХ и ФЧХ по величине и скорости отклонения амплитуд и фаз этих характеристик.

Таким образом, для получения постоянной информации о техническом состоянии фильтра и двигателя в целом необходимо установить датчики давления на входе и выходе масляного фильтра, а в бортовом ком-

пьютере установить программное обеспечение, позволяющее обрабатывать получаемые осциллограммы давления и сравнивать их с эталонными характеристиками. В процессе эксплуатации могут возникать критические ситуации с параметрами исследуемых характеристик, связанных с «масляным голоданием» на некоторых резонансных частотах. Наличие моделей фильтра, представленных в виде передаточных функций, позволяет, включив их в систему управления двигателя, изменять параметры системы смазки с целью улучшения эффективности его работы.

СПИСОК ЛИТЕРАТУРЫ

1. Ременцов А.Н., Фролов Ю.Н., Воронов В.П. Системы, технологии и организации услуг в автомобильном сервисе. М.: Изд. центр «Академия», 2014. 480 с.
2. Блекборн Дж., Ритхоф Г., Шерер Дж.Л. Гидравлические и пневматические силовые системы управления. Пер. с англ. М.: Изд. ИЛ, 1962. 614 с.
3. Гийон М. Исследования и расчет гидравлических систем. Пер. с фр. М.: Машиностроение, 1964. 387 с.
4. Гликман Б.Ф. Математические модели пневмогидравлических систем. М.: Наука. Физматлит, 1986. 386 с.
5. Гликман Б.Ф. Нестационарные течения в пневмогидравлических цепях. М.: Машиностроение, 1979. 256 с.
6. Атрошенко В.А., Шевцов Ю.Д., Яцынин П.В., Дьяченко Р.А., Педько М.Н. Технические воз-

можности повышения ресурса автономных электростанций энергетических систем. Краснодар: ИД «Юг», 2010. 192 с.

7. Григорьев М.А. Очистка масел двигателей внутреннего сгорания. М.: Машиностроение, 1983. 148 с.

8. Шевцов Ю.Д. К вопросу моделирования элементов систем смазки двигателей ДЭС как линейных электрических цепей // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2010. № 5(108). С. 179–182.

9. Атрошенко В.А., Шевцов Ю.Д., Василенко Н.В., Лысенко М.П., Кокорев В.В., Дьяченко Р.А. Способ оценки технического состояния двигателя внутреннего сгорания // Патент РФ на изобретение № 2259549. Бюллетень № 24. 27.08.2005.

Статья поступила в редакцию 09.11.2018.

REFERENCES

1. Rementsov A.N., Frolov Yu.N., Voronov V.P. *Sistemy, tekhnologii i organizatsii uslug v avtomobilnom servise* [Systems, technologies and organization of services in the automotive service]. Moscow: Akademiya Publ., 2014, 480 p. (rus)
2. Blekborn Dzh., Ritkhof G., Sherer Dzh.L. *Gidravlicheskiye i pnevmaticheskiye silovyye sistemy upravleniya* [Hydraulic and pneumatic power control systems]. Moscow: IL Publ., 1962, 614 p. (rus)
3. Giyon M. *Issledovaniya i raschet gidravlicheskikh sistem* [Research and calculation of hydraulic systems]. Moscow: Mashinostroyeniye Publ., 1964, 387 p. (rus)
4. Glikman B.F. *Matematicheskiye modeli pnevmogidravlicheskikh sistem* [Mathematical models of pneumohydraulic systems]. Moscow: Nauka. Fizmatlit Publ., 1986, 386 p. (rus)

5. Glikman B.F. *Nestatsionarnyye techeniya v pnevmogidravlicheskikh tsepyakh* [Unsteady flows in pneumatichydraulic circuits]. Moscow: Mashinostroyeniye Publ., 1979, 256 p. (rus)

6. Atroshchenko V.A., Shevtsov Yu.D., Yatsynin P.V., Dyachenko R.A., Pedko M.N. *Tekhnicheskiye vozmozhnosti povysheniya resursa avtonomnykh elektrostantsiy energeticheskikh sistem* [Technical possibilities of increasing the service life of autonomous power stations of energy systems]. Krasnodar: Yug Publ., 2010, 192 p. (rus)

7. Grigoryev M.A. *Ochistka masel dvigateley vnutrennego sgoraniya* [Purification of internal combustion engine oils]. Moscow: Mashinostroyeniye Publ., 1983, 148 p. (rus)

8. Shevtsov Yu.D. To the question of modelling of elements of systems of greasing engines des as linear electric chains. *St. Petersburg Polytechnical University*

Journal. Computer Science. Telecommunication and Control Systems, 2010, No. 5(108), Pp. 179–182. (rus)

9. Atroshchenko V.A., Shevtsov Yu.D., Vasilenko N.V., Lysenko M.P., Kokorev V.V., Dyachenko R.A.

Sposob otsenki tekhnicheskogo sostoyaniya dvigatelya vnutrennego sgoraniya [The method of assessing the technical condition of the internal combustion engine].

Patent RF No. 2259549, publ. 27.08.2005. (rus)

Received 09.11.2018.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

ШЕВЦОВ Юрий Дмитриевич

SHEVTSOV Yuriy D.

E-mail: YuriShud48@gmail.com

ДУДНИК Людмила Николаевна

DUDNIK Lyudmila N.

E-mail: lududnik@mail.ru

ЧЕРНУХА Юрий Владимирович

CHERNUKHA Yuri V.

E-mail: Chernukha@mail.ru

ФАДЕЕВ Евгений Дмитриевич

FADEEV Evgeniy D.

E-mail: fadeev_ed@mail.ru

НИКИФОРОВ Арсений Михайлович

NIKIFOROV Arseny M.

E-mail: nikiforovwork@mail.ru

DOI: 10.18721/JCSTCS.11406
УДК 004.3

АРХИТЕКТУРА СИСТЕМЫ СБОРА И ОБРАБОТКИ ДАННЫХ ДЛЯ ПОВЫШЕНИЯ ПРОИЗВОДИТЕЛЬНОСТИ ТРУДА РАЗРАБОТЧИКОВ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Д.А. Тимофеев, М.Ю. Маслов

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

Для повышения производительности труда работников необходимо идентифицировать и оптимизировать выполняемые производственные процессы с учетом факторов, влияющих на их выполнение. В области разработки программного обеспечения, как и в других сферах с преобладанием интеллектуальной работы, многие процессы остаются неявными и существенно различаются у разных сотрудников, а результаты и производительность труда существенно зависят от личных качеств и текущего психофизиологического состояния исполнителя. Для получения необходимой информации предложено использовать систему, выполняющую сбор и обработку данных о состоянии человека и выполняемых им действиях, и на их основе формирующую обратную связь, позволяя работнику более эффективно планировать и решать задачи с учетом его состояния и характерных для него рабочих процессов. Описаны архитектура системы и основные технические решения, принятые при ее проектировании.

Ключевые слова: архитектура программной системы, сбор данных, обработка данных, обратная связь, производительность труда.

Ссылка при цитировании: Тимофеев Д.А., Маслов М.Ю. Архитектура системы сбора и обработки данных для повышения производительности труда разработчиков программного обеспечения // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2018. Т. 11. № 4. С. 71–81. DOI: 10.18721/JCSTCS.11406.

ARCHITECTURE OF DATA ACQUISITION AND PROCESSING SYSTEM FOR IMPROVING PRODUCTIVITY OF SOFTWARE DEVELOPERS

D.A. Timofeev, M.Yu. Maslov

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

In order to enhance the workers productivity, it is necessary to identify and optimize the workflows with respect to main influencing factors. In software development industry, as well as in other areas with prevailing intellectual work, these workflows are often implicit and highly variable, and the product quality and worker performance to the large extent depend on the worker current psychophysical state. We propose to regain missing information by using a system to acquire and process the data about the developers' states and activities. To generate the data, portable sensors and computer software plugins are used. Based on the data, the system generates a feedback allowing the users to better plan and complete tasks with respect to their state and personal work processes. In this paper we present the system architecture and describe main technical decisions.

Keywords: software system architecture, data acquisition, data processing, feedback, productivity.

Citation: Timofeev D.A., Maslov M.Yu. Architecture of data acquisition and processing system for improving productivity of software developers. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2018, Vol. 11, No. 4, Pp. 71–81. DOI: 10.18721/JCSTCS.11406.

Введение

Во многих сферах производства основную роль играет интеллектуальная деятельность. Выполняемые специалистом процессы в этих областях обычно мало формализованы, а результаты и эффективность труда существенно больше зависят от личных качеств и текущего психофизиологического состояния работника. Отсутствие стандартизированных технологических процессов производства также приводит к трудностям в обучении новых сотрудников и распространении эффективных практик, снижает точность планирования работ и увеличивает зависимость компании от конкретных исполнителей.

Одной из важных отраслей производства, в которой интеллектуальный труд играет главную роль, является разработка программного обеспечения. Существенная особенность этой области — одновременная реализация многих взаимосвязанных процессов на разных уровнях, от управления проектом до непосредственного создания программного кода. Эти процессы можно разделить на два вида: групповые процессы, в которые вовлечены несколько участников, и индивидуальные процессы, выполняемые одним разработчиком.

Групповые процессы достаточно хорошо исследованы. К ним относятся, в частности, управление проектом и организация коллективной работы над проектом. Каждая методология управления программным проектом предполагает определенный набор этапов разработки системы, набор создаваемых артефактов и порядок взаимодействия разработчиков. Например, при использовании методологии Scrum [1] процесс разработки разделяется на итерации фиксированной длительности (спринты). Для каждого спринта в ходе планирования определяется цель и набор задач, в ходе спринта изменение состояния каждой задачи отмечается в доступном для всех участников виде, а по завершению спринта проводятся обзор выполнения задач и ретроспектива, цель которой — анализ хода выполнения спринта на основе обратной связи от разработчиков.

Индивидуальные процессы можно разделить на два типа. К первому типу относятся процессы, обусловленные используемой методологией управления проектом (например, изменение состояния выполняемых задач в системе управления задачами) или применяемыми инструментальными средствами (взаимодействие с системой контроля версий, тестирование и компиляция проекта). Хотя эти процессы в большинстве случаев формально не описываются, они могут быть смоделированы на основе существующих регламентов, описаний инструментальных средств или результатов наблюдения за действиями разработчиков. Ко второму типу можно отнести скрытые творческие процессы, такие как проектирование структур данных, разработка архитектуры и алгоритмов, написание программного кода, создание тестовых планов, написание документации. Именно эти процессы определяют в итоге свойства разрабатываемого программного обеспечения. При этом анализ итоговых артефактов (кода, данных, документации) и наблюдения за ходом их создания недостаточно для выявления и описания правил, которыми руководствуются разработчики, и выполняемых ими ментальных процессов.

Таким образом, индивидуальные творческие процессы, играя главную роль в создании программного продукта, оказываются одновременно наименее изученными. Несмотря на то, что многие авторы работали над созданием методик эффективного решения творческих задач определенных классов (изобретательство [2], проектирование систем [3], решение математических задач [4]), в области разработки программ на данный момент подобные методики не получили заметного распространения. Основными способами передачи знаний о способах разработки программ является обмен опытом с помощью книг, докладов на профессиональных конференциях, публикаций в социальных сетях или личного общения. При этом обсуждаемые подходы часто не представляются в виде методики, а демонстрируются на конкретных приме-



рах кода или программных проектах (в качестве исключения можно указать [5], где автор предлагает методику целенаправленного поиска алгоритма с учетом свойств задачи и существующих ограничений). Используемые при решении задач подходы существенно различаются в зависимости от используемых технологий, опыта и стиля мышления разработчика [6].

Индивидуальные процессы также играют важную роль при выполнении групповых процессов. К примеру, для организации коммуникации между участниками проекта нужно учитывать такие факторы, как канал связи (личная встреча, телефон, электронная почта, мессенджер) и график занятости каждого из участников. Выбор канала связи предполагает различные методики эффективного управления временем (например, звонок по телефону отвлекает работника от выполняемой задачи, в то время как сообщение по электронной почте может быть отложено до удобного или специально выделенного для этой цели момента времени). Помимо этого, важно текущее состояние работника. Известной проблемой при организации интеллектуальной деятельности является потеря работниками концентрации при прерывании выполняемой ими деятельности. Систематическое исследование этого явления было предпринято М. Чиксентмихайи, который ввел понятие потока [7] — оптимального состояния внутренней мотивации, в котором человек полностью сконцентрирован на выполняемой задаче. В состоянии потока работа выполняется максимально эффективно, однако в случае прерывания деятельности возврат в состояние потока требует существенного времени. Экспериментальные результаты демонстрируют, что состояние потока можно идентифицировать путем анализа электроэнцефалограммы (ЭЭГ) человека [8].

Психофизиологические показатели также могут использоваться для анализа других факторов, связанных с производительностью труда разработчиков программного обеспечения — в частности, для оценки эмоционального состояния [9] (с использо-

ванием таких показателей, как электроэнцефалограмма, температура тела, частота сердечных сокращений, мышечная активность, частота дыхания) или субъективной оценки трудности задачи [10] (в данном случае использовались движения глаз, электрическая активность кожи и ЭЭГ).

Таким образом, индивидуальные процессы оказывают принципиальное влияние на производительность труда как отдельных программистов, так и всего коллектива разработчиков, но при этом остаются относительно малоизученными. При этом выполняемые процессы одновременно проявляются в виде конкретных выполняемых разработчиком действий и оказывают существенное влияние на изменение психофизиологического состояния человека. Анализ этих данных может дать новые сведения о процессах интеллектуальной деятельности человека, а также предоставить новые возможности для повышения эффективности интеллектуального труда в области разработки программного обеспечения за счет оценки текущего состояния разработчиков и формирования обратной связи для предотвращения перехода в нежелательные состояния и поддержки необходимой активности. Для организации сбора и анализа данных разработан аппаратно-программный комплекс, архитектура которого описывается в данной статье.

Функциональность аппаратно-программного комплекса

Разработанный аппаратно-программный комплекс предназначен для выполнения следующих функций.

1. Сбор данных о деятельности пользователя: ввод данных, запуск и завершение процессов, операции в среде разработки программ, действия в браузере.

2. Регистрация электрофизиологических и биологических параметров человека, позволяющих оценить его психофизиологическое состояние. В качестве базовых показателей выбраны электроэнцефалограмма (ЭЭГ), двигательная активность, пульс, температура тела. Для формирования исходных данных используются как

бытовые датчики (например, неинвазивные нейроинтерфейсы Emotiv Insight, MUSE; фитнес-браслет Mio Link), так и специально разработанное в рамках АПК оборудование.

3. Поточковый анализ данных в соответствии с набором моделей и формирование обратной связи с пользователем: подача визуальных сигналов о текущем психофизиологическом состоянии или о заданных событиях, выдача рекомендаций о режиме труда и отдыха, мерах по повышению эффективности работы.

4. Передача данных на сервер более высокого уровня для анализа состояния группы людей и формирования групповых рекомендаций.

5. Сохранение истории событий в центральной базе данных и визуализация истории событий отдельных пользователей и групп.

6. Аутентификация и авторизация пользователей, управление правами доступа и разрешениями на сбор данных каждого конкретного вида, их передачу за пределы локального устройства пользователя (компьютера, смартфона, планшета) и сохранение в центральной базе данных.

В состав комплекса входят датчики, измеряющие значения физиологических показателей, и программная система, обеспечивающая сбор, хранение и обработку данных, поступающих от аппаратных компонентов.

Функционирование системы осуществляется путем сбора и анализа данных о состоянии и действиях пользователя. Каждый пользователь может самостоятельно определить, какие именно данные разрешено собирать, и дать или отозвать разрешение на передачу и сохранение этих данных. Анализ данных осуществляется как для каждого отдельного пользователя на его локальных устройствах и, при наличии разрешения, на удаленных серверах, так и для групп пользователей. В результате анализа пользователи получают обратную связь от системы.

Использование системы может увеличивать производительность труда при вы-

полнении интеллектуальной деятельности двумя путями.

1. Поточковый анализ данных о значениях физиологических показателей пользователя и его действиях, определение его психофизиологического состояния (в частности, сосредоточенность, высокая работоспособность, усталость, стресс) и формирование рекомендаций (например, сделать перерыв, выполнить физические или психологические упражнения, изменить вид деятельности, завершить рабочий день, обратиться к врачу).

2. Накопление данных о деятельности сотрудника в течение долгого времени, моделирование выполняемых им на индивидуальном и групповом уровне процессов, анализ и визуализация этих процессов.

Поточковый анализ данных позволяет определять желательные (удовлетворенность, комфорт, высокая работоспособность) и нежелательные состояния (усталость, стресс, болезнь) для каждого работника и оперативно выдавать рекомендации на основе предопределенных общих и персонализированных моделей. В частности, учитываются когнитивная нагрузка [11] и стрессогенные факторы [12], предполагается также анализ циркадных ритмов [13] и характера выполняемой деятельности [14]. Такой подход позволит своевременно идентифицировать и устранять или учитывать факторы, ведущие к снижению эффективности работы и удовлетворения от нее, а также может быть полезен для раннего обнаружения состояний, в которых может потребоваться медицинское вмешательство. Поточковый анализ предусмотрен не только для отдельных пользователей, но и для групп, что позволяет, например, применять этот подход при проведении собраний или организации обучения: ведущий может оперативно получать информацию о степени концентрации внимания и усталости участников и с учетом этих данных менять скорость подачи информации или планировать расписание.

Анализ процессов позволяет выделять и классифицировать виды деятельности

специалистов, связывать эти модели с измеряемыми показателями эффективности сотрудника и за счет этого идентифицировать более эффективные и менее эффективные процессы с учетом личных особенностей каждого специалиста. Наличие таких моделей позволит уточнять оценки времени для выполнения задач, упрощать обучение новых и менее опытных сотрудников, формировать рекомендации по улучшению производительности труда, а на групповом уровне – настраивать систему коммуникации и оптимизировать параметры процессов, связанных с управлением проектом.

Помимо этого, каждому конкретному специалисту изучение данных об истории собственных состояний и действий может дать информацию для развития и самосовершенствования. В этом смысле особенно важно, чтобы сбор данных не доставлял участникам неудобств, а любая обработка и хранение данных происходили только с их добровольного и осознанного

согласия. Нарушение этих принципов приведет лишь к снижению эффективности работников и нежеланию использовать систему.

Архитектура системы

Компоненты АПК представлены на рис. 1. Все программное обеспечение делится на клиентские компоненты, которые выполняются на компьютере или мобильном устройстве пользователя, и серверные компоненты, которые могут быть развернуты в локальной сети организации или размещены на одном или нескольких серверах в сети Интернет.

На устройствах пользователей выполняются локальные агенты – приложения, обеспечивающие сбор, временное хранение и обработку биометрических и иных данных пользователя. Сбор и первичную обработку данных каждого вида выполняют встраиваемые модули. Каждый из модулей может выступать в роли генератора событий или процессора событий.

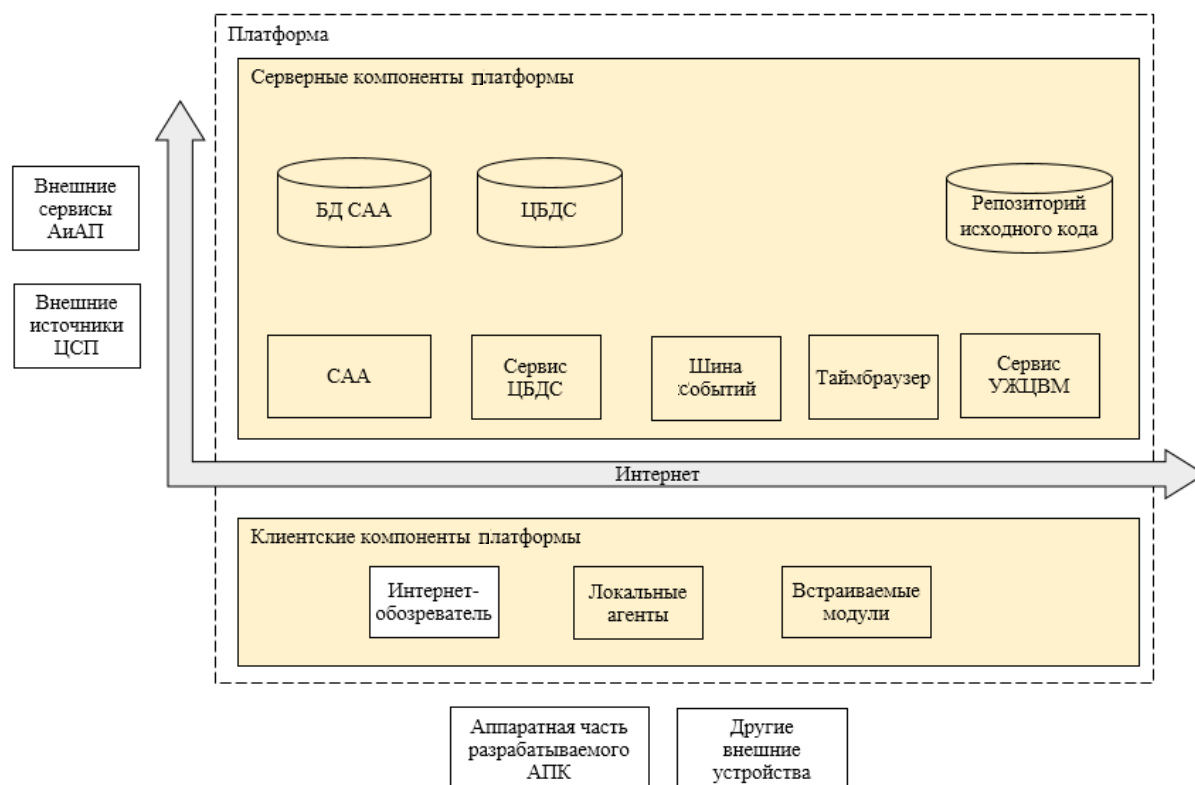


Рис. 1. Компоненты аппаратно-программного комплекса

Fig. 1. Components of hardware and software complex

Генераторы событий порождают данные в ходе взаимодействия с устройствами, операционной системой или приложениями. Каждый элемент данных помечается набором из трех идентификаторов, соответствующих группе, классу и типу события. Например, все события, связанные с психофизиологическими показателями человека, могут относиться к группе bio, их подмножество, соответствующее данным ЭЭГ — к классу eeg, а данные ЭЭГ с электрода AF3 — к типу AF3. Эти идентификаторы используются для выбора подмножества событий, на которые другие модули и локальные агенты осуществляют подписку. Каждый генератор порождает наборы событий и связанных с событиями данных. В простейшем случае событием считается период времени от активации до деактивации модуля. Данные о событиях представляют собой полученные от устройства в этот период значения, снабженные идентификатором соответствующего события и метками времени.

Процессоры событий подписываются на события определенного вида, читают данные о событиях, обрабатывают их и формируют новые события с привязанными к ним результатами обработки. Процессоры событий могут фильтровать данные (например, устраняя выбросы), вычислять статистические характеристики, преобразовывать данные. Важным процессором событий является триггер, формирующий событие при получении удовлетворяющих заданному условию данных.

Каждый встраиваемый модуль может быть загружен в нескольких экземплярах, каждый экземпляр имеет собственную конфигурацию. Набор параметров конфигурации зависит от конкретного модуля: процессоры событий обычно будут иметь параметры группы, класса и типа для фильтрации поступающих на вход данных, а модули для работы с устройствами — MAC-адрес или другой идентификатор устройства.

Наибольшее количество модулей поддерживает локальный агент для ОС Windows. Для его создания использовался

язык программирования C#. Встраиваемые модули представляют собой динамически загружаемые библиотеки и могут быть написаны на любом языке программирования, поддерживающем эту технологию. Большая часть существующих модулей также разработана на языке C#, часть модулей — на языке C++.

Серверная часть включает сервер аутентификации и авторизации (САА) и его базу данных, центральную базу данных событий (ЦБДС) и сервис для работы с ней, шину событий, сервис «таймбраузер», а также сервис управления жизненным циклом встраиваемых модулей (УЖЦВМ) и репозиторий исходного кода модулей.

Сервер аутентификации и авторизации обеспечивает аутентификацию пользователей, авторизацию от имени пользователя по протоколу OAuth2 и хранение пользовательских профилей. Сервер реализован на языке программирования PHP7 с использованием фреймворка Laravel. База данных реализована с использованием СУБД PostgreSQL 9.6.

Центральная база данных событий и шина событий обеспечивают взаимодействие серверной части АПК и локальных агентов. Локальные агенты подключаются к шине событий и передают сообщения о событиях и о связанных с ними данных в формате JSON. Для сокращения количества передаваемых пакетов и нагрузки как на сервере, так и на клиенте, данные о событиях группируются в пакеты настраиваемой длины.

Шина событий передает полученные пакеты сервису центральной базы данных событий, который сохраняет их в базу данных. Для организации групповой работы локальные агенты также могут подписаться на получение данных о событиях из шины событий. Шина событий передает полученные пакеты с соответствующей маской (группа, класс, тип) всем подписавшимся на них агентам. Сервис ЦБДС также предоставляет авторизованным пользователям программный интерфейс для доступа к архивным данным. Алгоритм сбора и накопления данных представлен на рис. 2.

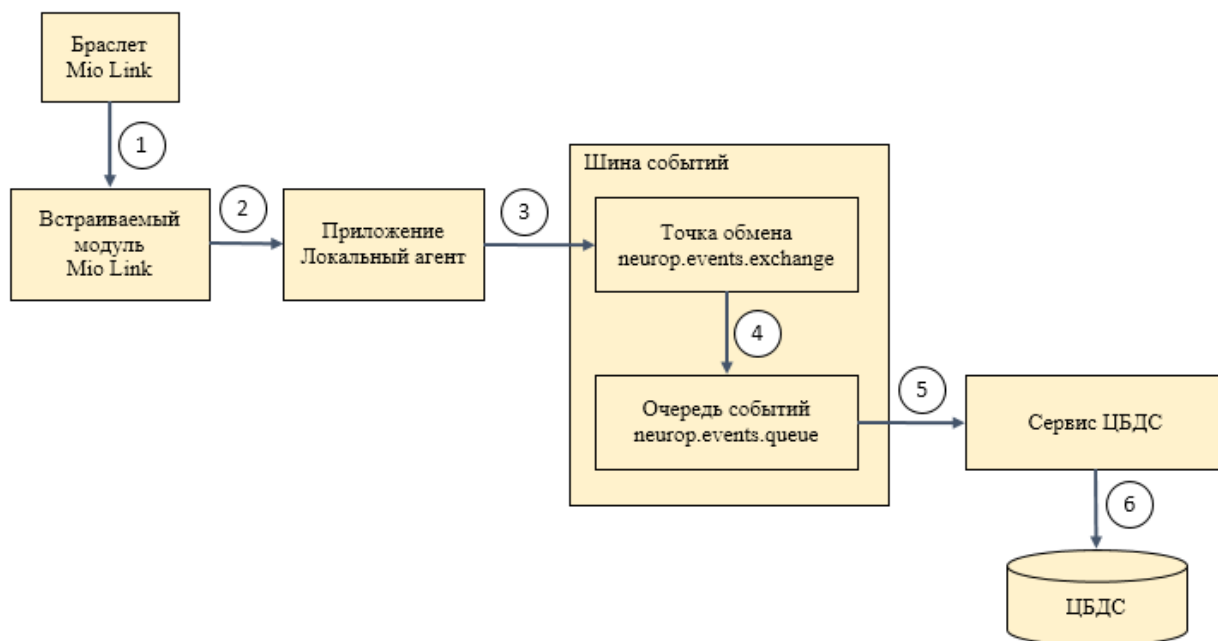


Рис. 2. Порядок обмена собираемыми данными между компонентами системы
 Fig. 2. The procedure for the exchange of collected data between system components

Рисунок 3 иллюстрирует порядок распространения данных между локальными агентами с использованием шины событий. В изображенном примере локальный агент-приемник данных подписывается на события группы *g1*, класса *c1* и типа *t1*.

Для реализации шины событий используется очередь сообщений RabbitMQ. Центральная база данных событий использует СУБД PostgreSQL 9.6, сервис ЦБДС реализован на языке программирования C# с использованием кроссплатформенной среды .NETCore.

Сервис управления жизненным циклом встраиваемых модулей используется для поддержки разработчиков, создающих и поддерживающих встраиваемые модули локальных агентов. Сервис реализует:

- репозиторий исходного кода со встроенными средствами управления версиями;
- витрину модулей;
- панель разработчика;
- подсистему коммуникации разработчиков и пользователей встраиваемых модулей.

Репозиторий исходного кода построен на базе сервера управления репозиториями Gogs и, как и остальные компоненты, ис-

пользует единый сервер аутентификации и авторизации. Сервис управления жизненным циклом встраиваемых модулей обеспечивает управление зависимостями, позволяя формировать сборки, автоматически включающие все модули, необходимые для работы выбранного модуля.

Сервис «таймбраузер» представляет собой инструмент для формирования, просмотра и анализа хронологических последовательностей событий и связанных с ними данных. Таймбраузер представляет собой инструмент для изучения аналитиками выполняемых пользователями процессов, а также для просмотра пользователями истории своих событий.

Альтернативные подходы

Задача сбора и анализа информации о действиях и текущем состоянии человека возникает не только при анализе процессов интеллектуальной профессиональной деятельности, но и во многих смежных областях. Двумя наиболее близкими к представленному проекту являются задачи мониторинга здоровья и протоколирования рабочего времени.

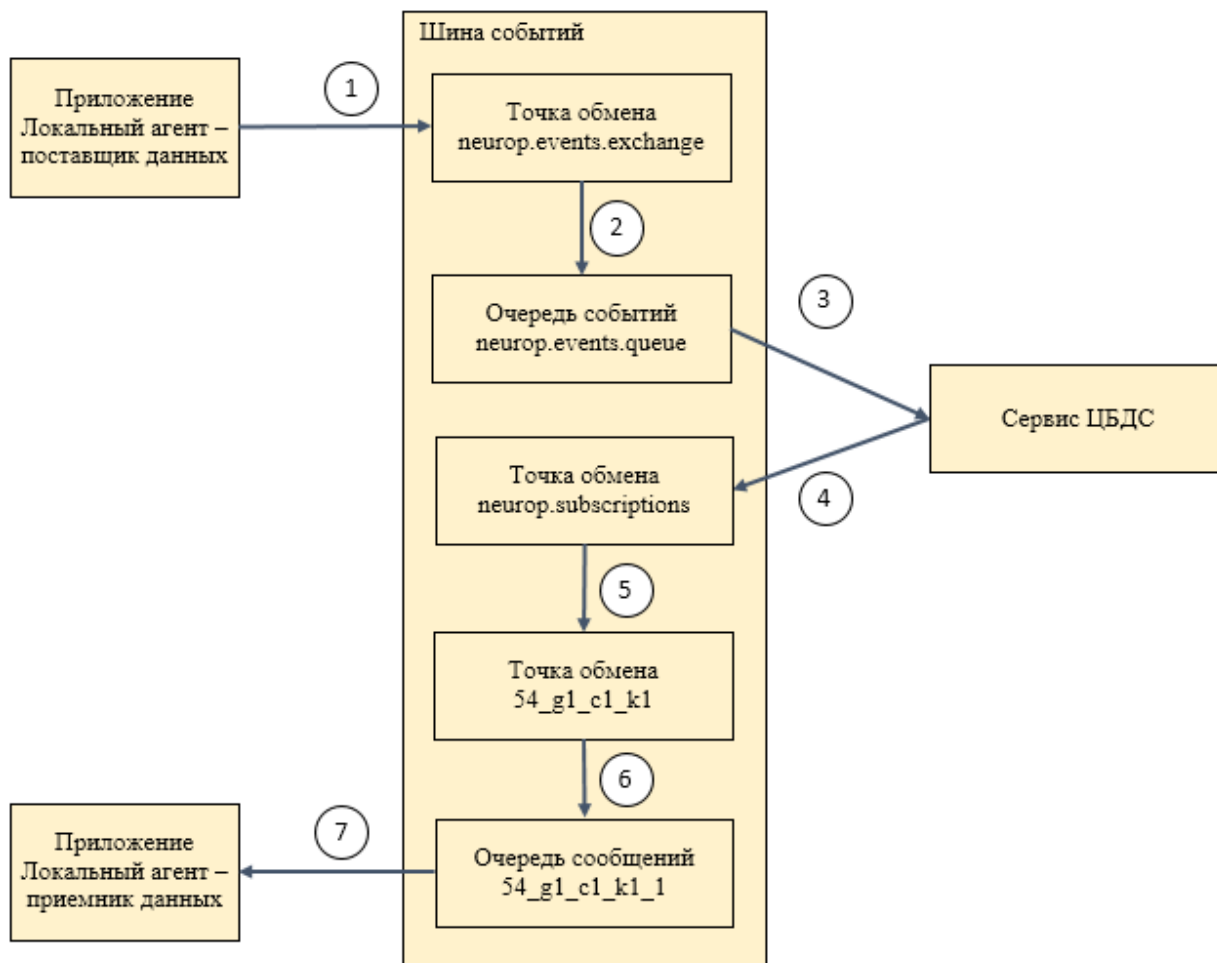


Рис. 3. Порядок распространения данных между локальными агентами

Fig. 3. The order of data distribution between local agents

Системы мониторинга физиологических параметров организма человека успешно используются для оценки и отслеживания динамики показателей здоровья и физического состояния (включая мониторинг показателей во время спортивных тренировок), ранней диагностики заболеваний и автоматического вызова помощи в экстренных случаях. В последние годы активные исследования ведутся в области мониторинга состояния здоровья с учетом контекста [15]. Задачей контекстно-зависимого мониторинга является сопоставление значений физиологических показателей с данными о контексте, в котором снимаются показания, такими как местоположение, время, вид или другие участники выполняемой деятельности. Так

как понятие контекста может быть определено произвольным образом, контекстно-зависимые системы обычно строятся для решения конкретных задач с фиксированным набором учитываемых факторов и их связей с целевыми показателями. Например, при диагностике сердечных заболеваний виды выполняемой человеком деятельности могут быть сведены к нескольким классам, отличающимся степенью физической нагрузки [15].

Протоколирование выполняемых задач и затрат рабочего времени на их выполнение — важная составляющая работы для специалистов, работающих на условиях почасовой оплаты, а также распространенный инструмент для анализа и повышения эффективности использования времени. Как



следствие, существует большое количество прикладных программ и веб-сервисов для протоколирования рабочего времени. Их можно разделить на две основные группы. В первую входят сервисы, в которых сведения о выполняемых задачах вводит сам пользователь. Многие из них предоставляют таймер, который пользователь может запускать в начале выполнения задачи и отключать после ее завершения. Основной целью таких систем является формирование отчетов и счетов на основе собранных данных. Примерами систем этого класса являются сервисы Toggl¹ и Zoho Invoice². Системы второго класса протоколируют выполняемые пользователем компьютера действия и используют собранные данные для автоматической идентификации вида деятельности. Основное применение такого подхода — отслеживание деятельности работников организации, не связанных с выполнением их должностных обязанностей, или для контроля счетов, выставляемых фрилансерами заказчикам; возможно также применение этих сервисов для формирования счетов и оценки собственной производительности труда. В качестве исходных данных могут служить сведения об используемых и запущенных приложениях, открытых вкладках браузеров, применении устройств ввода, периодически формируемые изображения экрана или кадры с веб-камеры. Примерами таких систем являются сервис Desk Time³ и приложение фриланс-биржи Upwork⁴ [16]. Ряд сервисов, таких как Waka Time⁵ или CodeAlike⁶, ориентированы на разработчиков программного обеспечения и предоставляют модули расширения для основных сред разработки программ и веб-браузеров. На основе собираемых данных эти сервисы предоставляют пользователю отчеты о затраченном времени и других показателях (например, о ко-

личестве написанных строк кода или долях времени, затраченных на кодирование и отладку) в каждом из выполняемых проектов, а также реализуют игровые приемы (достижения, соревнования) для повышения производительности труда.

Помимо перечисленных задач, протоколирование и анализ различных аспектов поведения людей используются при обеспечении безопасности (поиск аномалий в системах видеонаблюдения [17], обнаружение и предотвращение мошенничества [18]), лингвистике и когнитивных науках (исследование процессов письма и чтения [19, 20]), эргономике (анализ и проектирование пользовательских интерфейсов), рекламе и торговле (рекомендательные системы, персонализированные предложения) и многих других областях.

Применяемые в описанных областях решения являются специализированными. Они ориентированы на эффективную работу с данными, набор которых определяется конкретной решаемой задачей. В частности, существует четкое разделение между системами, направленными на работу с данными о значениях физиологических показателей, и системами учета рабочего времени. Описанная в данной статье архитектура системы сбора и анализа данных о состоянии и деятельности является открытой и расширяемой. Возможность создания собственных генераторов и процессоров событий, а также отсутствие принципиальных ограничений на вид данных о событиях позволяют использовать представленную систему не только для обеспечения работы конкретного алгоритма формирования обратной связи для повышения эффективности труда, но и в качестве исследовательской платформы для изучения процессов интеллектуальной деятельности человека. В частности, поскольку в качестве основного вида деятельности выбрана разработка программного обеспечения, собираемые с помощью системы данные могут использоваться для анализа процессов создания и модификации исходного кода программ [21].

¹ <https://toggl.com>

² <https://www.zoho.eu>

³ <https://DeskTime.com>

⁴ <https://upwork.com>

⁵ <https://WakaTime.com>

⁶ <https://codealike.com>

Заключение

В данной статье представлена архитектура системы сбора и анализа данных о психофизиологическом состоянии человека и выполняемых им в ходе интеллектуальной деятельности процессах, входящая в состав аппаратно-программного комплекса для регистрации и анализа электрофизиологических и биометрических параметров человека и предоставления биологической и оптической обратной связи. Предложенная архитектура является открытой и расширяемой и может использоваться для формирования массива данных для исследований в области процессов профессиональной деятельности человека, эргономии и других областях,

в которых важен учет контекста в наблюдаемых действиях. Непосредственная прикладная задача, которую решает система, состоит в подготовке данных, позволяющих классифицировать текущее состояние работника и формировать на его основе воздействия, помогающие избежать нежелательных состояний. В настоящее время ведется настройка параметров алгоритмов формирования обратной связи для специалистов, занятых в области разработки программного обеспечения.

Работа подготовлена в ходе реализации проекта в рамках Постановления Правительства РФ от 09.04.2010 № 218 при финансовой поддержке Министерства образования и науки РФ. Договор № 03.G25.31.0247 от 28.04.2017.

СПИСОК ЛИТЕРАТУРЫ

1. Schwaber K., Sutherland J. The Scrum Guide. 2017 // URL: <https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-US.pdf> (Дата обращения: 08.11.2018).
2. Альтшуллер Г.С. Алгоритм изобретения. М.: Московский рабочий, 1969. 272 с.
3. Джонс Дж.К. Методы проектирования. 2-е изд. Пер. с англ. М.: Мир, 1986. 326 с.
4. Пойа Д. Математическое открытие. 2-е изд. Пер. с англ. М.: Наука, 1976. 448 с.
5. Скиена С. Алгоритмы. Руководство по разработке. 2-е изд. Пер. с англ. СПб.: БХВ-Петербург, 2011.
6. Carter C. Sangler. The programmers stone, 1997 // URL: <https://www.datapacrat.com/Opinion/Reciprocity/r0/index.html> (Дата обращения: 10.11.2018).
7. Чиксентмихайи М. Поток: Психология оптимального переживания. Пер. с англ. 7-е изд. М.: Смысл; Альпина нон-фикшн, 2017.
8. Berta R., Bellotti F., De Gloria A., Pranthana D., Schatten C. Electroencephalogram and physiological signal analysis for assessing flow in games // IEEE Transactions on Computational Intelligence and AI in Games 5. 2013. No. 2. Pp. 164–175.
9. Kolakowska A., Landowska A., Szwoch M., Szwoch W., Wrobel M.R. Emotion recognition and its application in software engineering // 6th Internat. Conf. on Human System Interactions. 2013. Pp. 532–539.
10. Fritz T., Begel A., Myller S.C., Yigit-Elliott S., Züger M. Using psycho-physiological measures to assess task difficulty in software development // Proc. of the 36th Internat. Conf. on Software Engineering. 2014. Pp. 402–413.
11. Sweller J. Cognitive load during problem solving: Effects on learning // Cognit. Sci. 1988. No. 12. Pp. 257–285.
12. Chilton M.A., Hardgrave B.C., Armstrong D.J. Person-Job cognitive style fit for software developers: The effect on strain and performance // J. of Management Information Systems. 2014. No. 22:2. Pp.193–226.
13. Blatter K., Cajochen C. Circadian rhythms in cognitive performance: Methodological constraints, protocols, theoretical underpinnings // Physiology and Behaviour. 2006. Pp. 196–208.
14. Sawyer K. The cognitive neuroscience of creativity: A critical review // Creativity Research Journal. 2011. No. 23:2. Pp. 137–154.
15. Mshali H., Lemlouma T., Moloney M., Magoni D. A survey on health monitoring systems for health smart homes // Internat. J. of Industrial Ergonomics. 2018. No. 66. Pp. 26–56.
16. Upwork Help Center. Log Time with the Desktop App, 2018 // URL: <https://support.upwork.com/hc/en-us/sections/202260758-Desktop-Apps> (Дата обращения: 10.11.2018).
17. Ovhal K.B., et al. Analysis of anomaly detection techniques in video surveillance // Internat. Conf. on Intelligent Sustainable Systems. IEEE, 2017. Pp. 596–601.
18. Abdallah A., Maarof M.A., Zainal A. Fraud detection system: A survey // J. of Network and Computer Applications. 2016. Vol. 68. Pp. 90–113.
19. Leijten M., van Waes L. Keystroke logging in writing research: Using inputlog to analyze and visualize writing processes // Written Communication. 2013. No. 30(3). Pp. 358–392.
20. de Smet M.J.R., Leijten M., van Waes L. Exploring the process of reading during writing us-



ing eye tracking and keystroke logging // Written Communication. 2018. No. 35(4). Pp. 411–447.

21. **Timofeev D., Samochadin A.** An unified representation of source code authoring workflows

// Proc. of the 10th Internat. Joint Conf. on Knowledge Discovery, Knowledge Engineering and Knowledge Management. KMIS, 2018. Vol. 3. Pp. 228–232.

Статья поступила в редакцию 13.11.2018.

REFERENCES

1. **Schwaber K., Sutherland J.** The Scrum Guide. 2017. Available: <https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-US.pdf> (Accessed: 08.11.2018).
2. **Altshuller G.S.** *Algorithm of invention*. Moscow: Moskovskiy rabochiy Publ., 1969, 272 p. (rus)
3. **Jones J.C.** *Design Methods*. Moscow: Mir Publ., 1986, 326 p. (rus)
4. **Poya D.** *Mathematical discovery*. Moscow: Nauka Publ., 1976, 448 p. (rus)
5. **Skienya S.** *The Algorithms Design Manual*. St. Petersburg: BHV-Petersburg Publ., 2011. (rus)
6. **Carter A., Sangler C.** The Programmers Stone, 1997. Available: <https://www.datapacrat.com/Opinion/Reciprocity/r0/index.html> (Accessed: 10.11.2018).
7. **Csikszentmihaly M.** *Flow: The Psychology of Optimal Experience*. Moscow: Smysl; Alpina non-fiction Publ., 2017. (rus)
8. **Berta R., Bellotti F., De Gloria A., Pranthan D., Schatten C.** Electroencephalogram and physiological signal analysis for assessing flow in games. *IEEE Transactions on Computational Intelligence and AI in Games* 5, 2013, No. 2, Pp. 164–175.
9. **Kolakowska A., Landowska A., Szwoch M., Szwoch W., Wrobel M.R.** Emotion recognition and its application in software engineering. *2013 6th International Conference on Human System Interactions*, Pp. 532–539.
10. **Fritz T., Begel A., Myller S.C., Yigit-Elliott S., Züger M.** Using psycho-physiological measures to assess task difficulty in software development. *Proceedings of the 36th International Conference on Software Engineering*, 2014, Pp. 402–413.
11. **Sweller J.** Cognitive load during problem solving: Effects on learning. *Cognit. Sci.*, 1988, No. 12, Pp. 257–285.
12. **Chilton M.A., Hardgrave B.C., Armstrong D.J.** Person-Job cognitive style fit for software developers: The effect on strain and performance. *Journal of Management Information Systems*, 2014, No. 22:2, Pp.193–226.
13. **Blatter K., Cajochen C.** Circadian rhythms in cognitive performance: Methodological constraints, protocols, theoretical underpinnings. *Physiology and Behaviour*, 2006, Pp. 196–208.
14. **Sawyer K.** The cognitive neuroscience of creativity: A critical review. *Creativity Research Journal*, 2011, No. 23:2, Pp. 137–154.
15. **Mshali H., Lemlouma T., Moloney M., Magoni D.** A survey on health monitoring systems for health smart homes. *International Journal of Industrial Ergonomics*, 2018, No. 66, Pp. 26–56.
16. Upwork Help Center. Log Time with the Desktop App, 2018. Available: <https://support.upwork.com/hc/en-us/sections/202260758-Desktop-Apps> (Accessed: 10.11.2018).
17. **Ovhal K.B., et al.** Analysis of anomaly detection techniques in video surveillance. *2017 International Conference on Intelligent Sustainable Systems*. IEEE, 2017, Pp. 596–601.
18. **Abdallah A., Maarof M.A., Zainal A.** Fraud detection system: A survey. *Journal of Network and Computer Applications*, 2016, Vol. 68, Pp. 90–113.
19. **Leijten M., van Waes L.** Keystroke logging in writing research: Using inputlog to analyze and visualize writing processes. *Written Communication*, 2013, No. 30(3), Pp. 358–392.
20. **de Smet M.J.R., Leijten M., van Waes L.** Exploring the process of reading during writing using eye tracking and keystroke logging. *Written Communication*, 2018, No. 35(4), Pp. 411–447.
21. **Timofeev D., Samochadin A.** An unified representation of source code authoring workflows. *Proceedings of the 10th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2018)*. KMIS, 2018, Vol. 3, Pp. 228–232.

Received 13.11.2018.

СВЕДЕНИЯ ОБ АВТОРАХ/ THE AUTHORS

ТИМОФЕЕВ Дмитрий Андреевич
TIMOFEEV Dmitrii A.
E-mail: dtim@dcn.icc.spbstu.ru

МАСЛОВ Максим Юрьевич
MASLOV Maxim Yu.
E-mail: maslov@soft-consult.ru

DOI: 10.18721/JCSTCS.11407

УДК 004.75:004.658.3, 004.657, 004.023

ОБЗОР МЕТОДОВ ДИНАМИЧЕСКОГО РАСПРЕДЕЛЕНИЯ ДАННЫХ В РАСПРЕДЕЛЕННЫХ СИСТЕМАХ УПРАВЛЕНИЯ БАЗАМИ ДАННЫХ

С.Г. Попов, В.С. Фридман

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

Приведен обзор методов и связанных с ними алгоритмов распределения данных между узлами системы управления распределенными базами данных. Предложена классификация алгоритмов перераспределения данных и приведены описания алгоритмов для случая установившегося состояния функционирования системы управления базами данных. Рассмотрены подходы, основанные на обучающихся автоматах с памятью, моделях прогнозирования потоков запросов методами анализа временных рядов, эволюционные подходы, базирующиеся на генетических алгоритмах, и связанные с ними алгоритмы распределения. Для описанных алгоритмов выделены критерии оптимальности функционирования подсистемы динамического перераспределения данных системы управления распределенной базой данных. Приведенные подходы могут использоваться для построения подсистем перераспределения данных в системах управления распределенными базами данных в процессе их функционирования.

Ключевые слова: данные, распределенные базы данных, система управления базами данных, распределение данных, оптимизация, алгоритмы.

Ссылка при цитировании: Попов С.Г., Фридман В.С. Обзор методов динамического распределения данных в распределенных системах управления базами данных // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2018. Т. 11. № 4. С. 82–107. DOI: 10.18721/JCSTCS.11407

REVIEW OF METHODS FOR DYNAMIC DISTRIBUTION OF DATA IN DISTRIBUTED DATABASE MANAGEMENT SYSTEMS

S.G. Popov, V.S. Fridman

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

The article provides an overview of the methods and associated data distribution algorithms between nodes of a distributed database management system. The article proposes a classification of data redistribution algorithms and provides algorithms for a database management system functioning in a stable state. The article discusses approaches based on self-configuring finite-state machines with memory, forecasting query flows using time series analysis methods, heuristic and genetic algorithms. Redistribution algorithms are described for each method. The optimality criteria for the functioning of the dynamic data redistribution subsystem of the distributed database are highlighted for the described algorithms. The given approaches can be used to design data redistribution subsystems in control systems of distributed databases in the course of their operation.

Keywords: data, distributed databases, database management system, data distribution, optimization, algorithms.

Citation: Popov S.G., Fridman V.S. Review of methods for dynamic distribution of data in distributed database management systems. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2018, Vol. 11, No. 4, Pp. 82–107. DOI: 10.18721/JCSTCS.11407

Введение

В распределенной системе управления базами данных каждая часть базы должна участвовать в выполнении глобальных запросов, требующих доступа к данным на нескольких других узлах системы. Для получения данных с других узлов используется подсистема связи, время передачи данных по которой существенно влияет на общее время выполнения распределенного запроса [2]. Скорость получения данных из распределенных источников актуальна в таких областях, как геоинформационные системы [22], системы распределенных мультимедиа [15], распределенные системы хранения данных, ориентированные на Web [24]. В этом случае пренебречь временем передачи данных между отдельными частями распределенной базы данных нельзя, и задача оптимального размещения данных становится двухкритериальной: оптимальное размещение данных зависит как от объема данных запроса, так и от скорости каналов связи, соединяющих фрагменты базы.

Рассматривая размещение данных между узлами распределенной базы как динамическую систему, следует поддерживать оптимальность распределения данных между узлами с целью минимизации времени выборки данных из всей базы в целом. При этом глобальным критерием оптимальности функционирования такой системы остается минимизация времени выполнения всех запросов всех абонентов. Непрерывное поддержание распределенной СУБД в оптимальном состоянии невозможно, поскольку непрерывное распределение фрагментов, оказывающее решающее влияние на надежность и производительность [4, 5], является NP-полной задачей [11, 12].

В настоящее время исследователями активно предлагаются подходы к решению задачи дискретной оптимизации размещения данных в системах распределенных баз данных, основанные на методах оптимизации. Основными подходами являются реализации избыточных и неизбыточных методов распределения данных между узлами, ориентированными на технологии переноса

данных [2] или их репликации [6, 20]. Рассматривая методы оптимизации размещения данных без избыточности, основной задачей становится выделение фрагмента в распределенной системе баз данных фрагмента данных для переноса на другой узел и определение этого узла. При этом основной принцип распределения – достижение максимальной локальности приложений и фрагментов. На практике это означает хранение фрагментов ближе к тому узлу, где они чаще используются.

Алгоритмы динамического распределения данных в распределенных базах данных можно классифицировать относительно их внутренних и внешних особенностей реализации, на основе их области применения, эффективности и критериев оптимальности. Выделенные классификационные признаки позволяют разделить алгоритмы по способу определения стоимости операций [17], анализу доступных ресурсов на узле [8, 9], типу применяемой избыточности данных [3, 7, 13], способу распределения данных [2], по методам описания [21] и применению распределенной транзакционной модели [22].

Метод пороговой оценки с ограничением по времени, объему и расстоянию

Пороговый алгоритм с ограничением по времени, объему и расстоянию [1] является результатом набора исследований [2, 8, 9, 11, 13] и обеспечивает множественное обращение к одному фрагменту данных от нескольких узлов, путем перераспределения данных на основе предварительно собранной статистики обращений к данным на каждом узле. Алгоритм выполняется на каждом узле и обеспечивает передачу фрагмента данных тому узлу, доступ из которого осуществляется с максимальной интенсивностью за прошедший временной интервал.

Алгоритм реализуется в две фазы: подготовки и сбора статистических данных и анализа необходимости перемещения фрагмента данных. Используемые в алгоритме обозначения приведены в табл. 1.

Таблица 1

Обозначения в пороговом алгоритме с ограничением по времени, объему и расстоянию

Table 1

Symbols definitions in the algorithm of threshold, time, volume, and distance constraint

Обозначение	Значение
F	Количество фрагментов всей распределенной базы данных
S	Число узлов распределенной базы данных
A	Ограничение доступа на перераспределение фрагментов
B	Ограничение времени на перераспределение фрагментов
A_p^q	p -я попытка доступа к узлу q
n_i^j	Общее количество попыток доступа от узла j к фрагменту i внутри временного интервала вплоть до текущего времени попытки доступа t
V_i^j	Размер данных, передаваемых между фрагментом i и узлом j внутри временного интервала вплоть до текущего времени t
D_i^j	Расстояние между узлами

Фаза подготовки и сбора исходных данных. В этой фазе производится сбор статистических данных для определения необходимости перемещения фрагментов данных между узлами в базе данных.

Предположим, что имеется F фрагментов и S узлов в распределенной системе управления базами данных. Каждый узел имеет как минимум один расположенный на нем фрагмент. Для каждого узла формируется матрица, в которой сохраняется метрическое расстояние между узлами, которым может являться скорость передачи данных между узлами. Каждый узел также хранит значения двух ограничений: порог числа повторных доступов к фрагменту за единицу времени α и ограничение на время перераспределения фрагментов β . Эти параметры позволяют управлять задержкой перед переносом данных на другой узел и могут выбираться статически или динамически.

Процесс подготовки состоит из двух шагов.

Шаг 1. Расположить фрагменты распределенной базы данных в первоначальной расстановке на разных узлах, используя любой статичный метод распределения данных без использования избыточности.

Шаг 2. Написать таблицу логов доступа *Access_Log* для каждого узла. Она имеет следующую структуру: *Access_Log(AFID; ASID; ADateTime; DataVol)*, где *AFID* – ID фрагмента, к которому осуществляется обращение; *ASID* – ID узла, инициирующего это обращение; *ADateTime* – дата и время обращения; *DataVol* – размер данных, переданных от и к фрагменту; каждый узел хранит записи логов для каждого обращения к фрагментам, расположенным на этом узле; каждая запись *Access_Log* обозначается как A_p^q – (p -е обращение на узле q , где $p = 1, 2, \dots, \infty$ и $q = 1, 2, \dots, S$).

Фаза принятия решения о переносе фрагмента. Фаза принятия решения о переносе фрагмента выполняется каждый раз после обращения к фрагменту данных на любом из узлов. Пусть обращение сделано от узла j к фрагменту i , расположенному на узле q во время t , где $i = 1, 2, \dots, F$; $j = 1, 2, \dots, S$; $q = 1, 2, \dots, S$, причем $q = j$ или $q \neq j$. Локальная программа на узле q выполняет четыре шага при каждом обращении к фрагменту этого узла:

Шаг 1. Сделать запись A_p^q в *Access_Log* на узле q .

Шаг 2. Если ID обращающегося узла в лог-записи A_p^q совпадает с ID узла q , то обращение локальное ($q = j$) и, следовательно, делать ничего не нужно.

Шаг 3. Если ID обращающегося узла в $Access_Log$ A_p^q не совпадает с ID узла q , значит обращение удаленное ($q \neq j$), следовательно, необходимо выполнить следующие подпункты.

Шаг 3.1. Вычислить общее количество обращений к фрагменту i от всех узлов, включая локальные обращения, внутри временного интервала β вплоть до текущего времени обращения t . Предполагается, что n_i^s обозначает общее количество обращений от узла s к фрагменту i , расположенному на узле q внутри временного интервала β вплоть до текущего времени t , где $s = 1, 2, \dots, S$.

Шаг 3.2. Вычислить средний размер данных в байтах, передаваемых между фрагментом i и всеми узлами, включая q , откуда были обращения к i внутри временного интервала β вплоть до текущего времени t . Пусть $A_p^q V_i^s$ обозначает размер данных в байтах, передаваемых между фрагментом i , расположенном на узле q , и узлом s в обращении A_p^q внутри временного интервала β вплоть до текущего времени t , где $s = 1, 2, \dots, S$ и $p = 1, 2, \dots, \infty$. Также пусть $V_i S_i$ обозначает средний размер данных, переданных между фрагментом i , расположенном на узле q , и узлом s внутри временного интервала β вплоть до текущего времени t

$$V_i^S t = \left(\sum A_p^q V_i^s \right) / \sum n_i^s.$$

Шаг 3.3. Если общее количество обращений к фрагменту i от узла i внутри временного интервала β вплоть до текущего времени t больше, чем ограничение доступа α , то есть $n_i^j > \alpha$, где $j, s = 1, 2, \dots, S$; $j \neq s$, и средний объем переданных между фрагментом i и узлом j данных в байтах внутри временного β вплоть до текущего времени t больше среднего объема пере-

данных между фрагментом i и всеми узлами, включая q , то есть $V_i^j t > V_i^S t$, где $j, s = 1, 2, \dots, S$, то выполнить следующие подпункты.

Шаг 3.3.1. Если обращающийся узел j не удовлетворяет ограничениям шага 3.3, то ничего не делать.

Шаг 3.3.2. Если только один обращающийся узел j подходит под эти ограничения, то фрагмент i передислоцируется на узел j и удаляется из текущего узла q с попутным обновлением каталогов.

Шаг 3.3.3. Если более чем один обращающийся узел j удовлетворяет ограничениям, обозначенным на шаге 3.3 в ходе одновременного обращения, то необходимо вычислить расстояние между узлом q , где располагается фрагмент i , и узлами, удовлетворяющими ограничениям, обозначенным на шаге 3.3. Выбрать узел с максимальной дистанцией от узла q и перенести фрагмент на этот узел, удалив этот фрагмент из q и обновив каталоги.

Преимущество порогового алгоритма — возможность динамической настройки момента переноса данных между узлами относительно двух параметров: времени и интенсивности запросов. Это достигается корректировкой размера временного окна и значения порогов срабатывания алгоритма относительно интенсивности поступающих внешних запросов. Основной недостаток — сложность выбора пороговых значений при скачкообразных изменениях интенсивности, что может привести к серьезному повышению накладных расходов на передачу данных без видимого приращения эффективности функционирования системы в целом.

Концепция распределенных торгов Mariposa

Подход Mariposa поддерживает функционально полный набор операций с фрагментами данных распределенной базы данных. Фрагментами являются единицы хранения данных, которые «покупаются» и «продаются» узлами. Текущий владелец фрагмента может в любое время разделить его на два фрагмента или наоборот, соединить два фрагмента в один. Идея подхода состоит в

применении механизма деятельности биржи для подготовки и проведения торговой операции: каждый участник заявляет о выполняемых им операциях, делает ставки на запрашиваемые операции, а централизованный брокер определяет, кому и когда выполнять запрос. Для определения параметров ставки используется эвристический алгоритм.

Каждый запрос Q имеет бюджет $B(t)$, который может использоваться для выполнения запроса. Бюджет — это не увеличивающаяся функция во времени, являющаяся оценкой стоимости, которую пользователь дает для ответа на свой запрос в конкретный момент времени t .

Брокер, обрабатывающий запрос Q , получает план запросов, содержащий множество подзапросов $\{Q_1; \dots; Q_n\}$ и $B(t)$. Каждый подзапрос представляет собой ограничение одной переменной на фрагменте F таблицы или соединение двух фрагментов из двух таблиц. Брокер пытается решить каждый подзапрос Q_i , используя либо *протокол дорогих ставок*, либо более дешевый *протокол выставления заказа*. Решением является перечень узлов, которые будут выполнять подзапросы.

Протокол дорогих ставок. Реализация протокола дорогих ставок состоит из двух этапов. На первом этапе брокер отправляет запросы на участие в торгах. Запрос ставки включает в себя часть плана выполнения запроса, на который делается ставка. Участники торгов возвращают заявки, представленные в виде тройки C_i, D_i, E_i , указывающей, что претендент разрешит подзапрос Q_i для стоимости C_i в течение задержки D_i после получения подзапроса, и что эта ставка действительна только до истечения срока действия E_i .

На втором этапе протокола брокер уведомляет победивших участников торгов, что они были выбраны. Брокер также может уведомлять и проигравших. Если уведомления не произошло, то срок годности ставок истекает, и они могут быть удалены участниками торгов. Этот процесс требует обмена большим числом сообщений между узлами.

Протокол выставления заказа. Большинство запросов не требуют трудоемких вы-

числений. Чтобы оправдать такой уровень накладных расходов, они используют более дешевый и простой протокол выставления заказа, отправляющий каждый подзапрос на узел обработки, который несомненно выиграл бы процесс торгов, если бы в них участвовал. Этот узел получает запрос и обрабатывает его, возвращая ответ в форме «счета за услуги». Если узел отказывается от подзапроса, он может либо вернуть его брокеру, либо передать его на следующий по приоритету узел обработки. Если брокер использует более дешевый протокол заказа, существует некоторая опасность не найти решения для запроса в пределах выделенного бюджета. Брокер не всегда знает стоимость, которая будет «взиматься» с выбранного узла обработки, и временные задержки при передаче данных от этого узла. Однако риск компенсируется невысокими вычислительными затратами при реализации этого более быстрого протокола.

Протокол выбора плана. В процессе приема ставок все подзапросы на каждом шаге обрабатываются параллельно, тем не менее следующий шаг не может начаться до тех пор, пока предыдущий не завершен. Поэтому на каждом шаге вместо рассмотрения ставок для каждого отдельного подзапроса рассматриваются сразу множества ставок для подзапросов каждого шага.

При использовании протокола ставок брокеры должны выбирать выигрышную ставку для каждого подзапроса с совокупной стоимостью C и суммарной задержкой D так, чтобы суммарные затраты были меньше или равны затребованным затратам $B(D)$. Существует две проблемы, затрудняющие поиск наилучшего множества ставок: параллелизм подзапросов и комбинаторное пространство поиска. Так как общая задержка не является суммой задержек D_i каждого подзапроса Q_i , поскольку в каждом шаге плана запроса есть параллелизм и с большой вероятностью $D < \sum_i B_i(D)$. Кроме того, количество возмож-

ных множеств ставок растет экспоненциально с количеством шагов в плане запроса.

Предполагаемая задержка для обработки множества подзапросов на каждом шаге



равна наибольшему времени ставки в множестве. Количество различных значений задержек может быть не более, чем общее количество ставок по подзапросам в множестве подзапросов. Для каждого значения задержки оптимальное множество ставок состоит из наиболее дешевых ставок для каждого подзапроса, который может обрабатываться в течение данной задержки. Объединяя множества заявок в шаге и рассматривая их как совокупную ставку, брокер может свести проблему принятия ставки к более простой проблеме выбора одной ставки из числа агрегированных ставок для каждого шага.

При использовании протокола дорогих ставок брокер получает множества из нуля и более ставок для каждого подзапроса. Если для какого-либо подзапроса нет ставки, или не существует множества ставок, удовлетворяющих минимальным клиентским требованиям к цене и производительности из $B(D)$, то брокер должен запрашивать дополнительные ставки, соглашаться выполнить подзапрос самому или уведомить покупателя о том, что запрос не может быть запущен. Возможен случай, когда несколько множеств ставок удовлетворяют минимальным требованиям, поэтому брокер должен выбрать лучшее множество ставок. Чтобы сравнить коллекции ставок, определяется функция разности в множестве ставок — $B(D) - C$.

Протокол распределения ресурсов. Суть проблемы заключается в определении относительных количеств ресурсов времени и затрат, которые должны быть распределены для каждого подзапроса. В [6] для решения этих задач предложен эвристический алгоритм. Возможно, он не оптимален, но показывает хорошие результаты. Предложенный алгоритм является жадным. Он производит пробное распределение, в котором полная задержка является наименьшей из возможных, а затем делает обмены до тех пор, пока не перестанут появляться более выгодные решения. Таким образом, предлагается ряд решений с постоянно увеличивающимися значениями задержки для каждого этапа обработки. На любой итерации алгоритма

предлагаемое решение содержит набор ставок с определенной задержкой для каждого этапа обработки. Для каждого набора ставок с большей задержкой вычисляется градиент затрат — снижение затрат, которое будет иметь место для этапа обработки путем замены коллекции в решении рассматриваемой коллекцией, поделенной на увеличение времени, которое, в свою очередь, было бы результатом замены.

Алгоритм начинается с рассмотрения множества ставок с наименьшей задержкой для каждого этапа обработки и вычисления общей стоимости C и общей задержки D . Вычисляется градиент стоимости для каждой неиспользованной ставки. Затем выполняется шаг обработки, который содержит неиспользованную ставку с максимальным градиентом затрат, B . Если эта ставка заменяет текущую, используемую на этапе обработки, то стоимость станет C с задержкой D . Если результирующая разница больше в D' , чем в D , то необходимо сделать замену ставки. То есть если $B(D') - C' > B(D) - C$, то нужно заменить B на B' , пересчитать все градиенты затрат для этапа обработки, включающие B' , продолжить делать замены до тех пор, пока разность не перестанет увеличиваться.

Используя либо протокол дорогой ставки, либо протокола заказа, брокер должен иметь возможность определить один или несколько узлов обработки каждого подзапроса для поиска участников торгов. В Mariposa это достигается через систему объявлений.

Протокол объявлений о выполняемых действиях. Узлы заявляют о своей готовности выполнять различные услуги, размещая рекламные объявления. Серверы имен сохраняют эти рекламные объявления в таблицу объявлений. Брокеры просматривают таблицу объявлений, чтобы узнать, какие узлы могут выполнять задачи, в которых они нуждаются. В табл. 2 показаны поля таблицы объявлений.

Для обработки запросов на фрагментированных таблицах и поддержки купли, продажи, расщепления и объединения фрагментов алгоритмы подхода Mariposa делятся на три уровня, как показано на рис. 1.

Таблица 2

Обозначения, используемые в полях таблицы объявлений

Table 2

Symbols definitions in the fields of table notification

Поле таблицы	Описание
<i>query_template</i>	Описание предлагаемой услуги в виде запроса с пустыми параметрами
<i>server_id</i>	ID узла, предлагающего услугу
<i>start_time</i>	Время начала исполнения предлагаемой услуги (может быть в будущем)
<i>expiration_time</i>	Время, до которого предложение действительно
<i>price</i>	Цена, взимаемая узлом за услугу
<i>delay</i>	Время, за которое узел планирует выполнить услугу
<i>limit_quantity</i>	Максимальное количество выполнений услуги узлом по заданной цене и времени выполнения
<i>bulk_quantity</i>	Количество заказов, необходимых для реализации предложенной цены и времени выполнения
<i>to_whom</i>	Множество брокеров, которым доступно данное объявление
<i>other_fields</i>	Комментарии и другая информация, специфичная для данного объявления



Рис. 1. Архитектура системы Mariposa
Fig. 1. The architecture of the system Mariposa

Архитектура разделяется на клиентскую программу, инициирующую запросы системе Mariposa, дополняя инструкциями по ценам и ставкам; слой промежуточного программного обеспечения; локальный узел исполнения. Уровень промежуточного программного обеспечения состоит из нескольких модулей подготовки запроса и брокера запросов. Локальный узел состоит из покупателя, менеджера хранилища данных и локального механизма исполнения запросов.

Достоинства предложенного подхода — возможность применения нескольких стратегий оценки заявки на выполнение запроса и технология иерархической агрегации заявок одного уровня в укрупненную заявку, что снижает вычислительную сложность построения плана запроса. К недостаткам следует отнести наличие централизованного брокера, что требует дополнительного времени для передачи заявок и несколько снижает надежность системы в целом.

Концепция одноранговых сетей в кластерной распределенной системе хранения

Концепт PeertoPeer сетей — потенциально наиболее эффективная архитектура для операций с данными. Алгоритм



используется в специфической кластеризованной архитектуре распределенных СУБД, названной FlexiPeer [3]. В этой архитектуре узлы кластеризованы по критериям приоритета локальности, каждый кластер управляется локальным администратором кластера (ЛАК), а вся архитектура администрируется глобальным администратором кластера (ГАК). Процесс кластеризации выполняется на основе количества узлов в регионе, которые заносятся в таблицу местоположения. В этой таблице имеется подробная информация о каждом узле: ID узла, местоположение и регион, по которому выполняется кластеризация узлов. ЛАК соответственного кластера обрабатывает атрибуты: локальный ID кластера (LCA_id), ID узла ($site_id$), регион кластера ($cluster_region$), местоположение узла ($site_location$) и тип данных, хранящихся на этом узле ($site_type$). Глобальные атрибуты всех кластеров (GCA_id , LCA_id и $cluster_region$) обрабатывает ГАК. Кластеризация помогает, если данные по запросу не найдены в конкретном узле: запрос будет перенаправлен к соответствующему ЛАК для дальнейшей обработки. Этот процесс называется *пересылкой неуспешных запросов*. Возрастающее количество таких запросов влияет на время обработки транзакции. Несмотря на то, что очень трудно избежать подобной ситуации, можно создать почву для сокращения числа неуспешных запросов, например, с помощью стратегии перераспределения для узлов одного кластера.

Перераспределение фрагментов осуществляется при помощи венгерского алгоритма оптимизации, где используется матрица стоимостей распределения, в которой полями является количество транзакций, обращающихся к данным конкретного узла. Количество обращений к конкретному узлу в матрице распределения записывается как вес обращения к конкретному фрагменту от конкретного узла. Функция стоимостей на основе венгерского алгоритма оптимизации для

определения перераспределения фрагментов по узлам определяется в соответствии с выражением:

$$\min Z = \sum \sum C_{ij} \times X_{ij}.$$

Применяя венгерский алгоритм оптимизации, можно получить оптимальное распределение фрагментов по узлам.

Достоинство предложенного алгоритма заключается в получении нового решения за полиномиальное время, однако алгоритм предполагает жесткую иерархическую структуру подчиненности кластеров и отсутствие явного критерия перераспределения данных между узлами.

Прогнозирование состояния распределенной базы данных методами временных рядов

Для глобальной оптимизации с прогнозированием состояния распределенной базы данных применяется метод оценки будущего состояния базы данных на основе интегрированной модели авторегрессии и скользящего среднего (Бокса–Дженкинса) ARIMA. Алгоритм обеспечивает прогноз состояния базы данных и на его основе перемещение фрагментов базы данных из одного узла в другой, а также регулирует увеличение и уменьшение числа узлов, на которых хранятся фрагменты баз данных, т. е. осуществляет балансировку не только перераспределением объема данных по существующим узлам, но и изменением числа узлов. Алгоритм состоит из четырех фаз.

1. Выполняется прогнозирование на k шагов вперед для оценки рабочих нагрузок в ближайшие часы.

2. Выполняется корректировка количества узлов, в ходе которой выясняется, что необходимо сделать (добавить дополнительные узлы или удалить лишние).

3. Выбираются фрагменты с низкой стоимостью миграции для необходимых миграций.

4. Выполняется миграция фрагментов для балансировки нагрузок.

```

1: For  $k = 1 : \omega$  Do
2:    $\hat{n}_{t+k} \leftarrow \left[ \sum_{j=1}^m \hat{W}_{F_j,t}(k) / \theta_l \right]$ ;
3: EndFor
4: If  $n < \hat{n}_{t+1}$  Then
5:    $n \leftarrow \hat{n}_{t+1}$ ;
6:   добавить  $\hat{n}_{t+1} - n$  узлов
7: ElseIf  $n > \max_{1 \leq k \leq \omega} (\hat{n}_{t+k})$  Then
8:    $n_d \leftarrow \max_{1 \leq k \leq \omega} (\hat{n}_{t+k})$ ;
9:   While  $n_d < n$  Do
10:     $N_r \leftarrow \{N_i : N_i \in \Omega_N, \nexists N_l \in \Omega_N \text{ такого, что } \hat{W}_{N_i,t}(1) > \hat{W}_{N_l,t}(1)\}$ ;
11:    ForAll  $F_j \in \Omega_F$  таких, что  $a_{rj} = 1$ 
12:       $\Omega_M \leftarrow \Omega_M \cup \{F_j\}$ ;
13:       $a_{rj} \leftarrow 0$ ;
14:    EndFor
15:     $n \leftarrow n - 1$ ;
16:     $\Omega_N \leftarrow \Omega_N - \{N_r\}$ ;
17:  EndWhile
18: EndIf.

```

Листинг 1. Фаза корректировки количества узлов
Listing 1. Node adjustment phase

Алгоритм корректировки числа узлов. Корректировка количества узлов, выполняемая в фазе 2, описана в листинге 1. Сначала оценивается требуемое количество рабочих узлов для следующих ω единиц времени: $\hat{n}_{t+1}, \hat{n}_{t+2}, \dots, \hat{n}_{t+\omega}$ (шаги 1–3). Новые узлы добавляются, если количество существующих рабочих узлов недостаточно для следующего момента времени $n < \hat{n}_{t+1}$ (шаги 4–6). Некоторые узлы удаляются, если есть лишние работающие узлы для всех ω единиц времени $n > \max_{1 \leq k \leq \omega} (\hat{n}_{t+k})$ (шаг 7). В случае добавления узлов учитываются нагрузки в следующие моменты времени для обеспечения достаточного количества рабочих узлов в тот момент времени. Однако для удаления узлов учитываются все ω моменты времени для предотвращения появления ненужных затрат на миграцию данных в течение следующих ω моментов времени.

В случае удаления нескольких узлов алгоритм итеративно выбирает узел для удаления с наименьшей нагрузкой в момент вре-

мени $t + 1$ (шаги 8–16). В [11] указано, что миграция фрагмента занимает от нескольких секунд до нескольких минут. Поскольку миграция любого фрагмента завершается за единицу времени (час), обозначим стоимость миграции фрагмента его нагрузкой в момент времени $t + 1$ и минимизируем общую стоимость миграции выбором узлов с низкой нагрузкой в момент времени $t + 1$ с их последующим удалением.

Алгоритм оценки перегруженности узлов. После фазы корректировки количества узлов алгоритм проверяет, есть ли узлы, которые в следующий момент времени будут перегружены. Если это так, то некоторые фрагменты, принадлежащие этим узлам, выбираются в качестве перемещаемых. Фаза выбора таких фрагментов (фаза 3) изложена в листинге 2. Если узел работает со скоростью выше верхней границы θ_u рабочей нагрузки узла в момент времени $t + 1 - \hat{W}_{N_i}(1) > \theta_u$, алгоритм итеративно выбирает фрагмент для удаления из узла, пока нагрузка не упадет ниже θ_u . По возможности

выбранные фрагменты должны удовлетворять «ограничению 1» модели ARIMA [8] (шаги 3–5) для стабилизации узла после удаления из него фрагментов. Если таких фрагментов нет, то алгоритм использует основное правило выбора фрагментов с малой нагрузкой в момент времени $t + 1$ (шаг 7) для уменьшения стоимости миграции.

Алгоритм миграции фрагментов между узлами. На последнем шаге осуществляется этап миграции фрагментов. Фаза миграции фрагментов (фаза 4) описана в листинге 3. Фаза выполняется в случае, если некоторые фрагменты должны быть перенесены из своих начальных узлов. После сортировки фрагментов в порядке убывания нагрузок в момент времени $t + 1$ (шаг 1), алгоритм итеративно ищет узел назначения для каждого фрагмента. Для того чтобы избежать возникновения ненужных миграционных затрат, выбранные узлы проверяются на перегруженность после получения новых фрагментов, что эквивалентно соблюдению условия $\hat{W}_{N_i}(1) > \theta_u$ (шаг 4). По возможности выбранные фрагменты должны удовлетворять «ограничению 2» модели ARIMA [8] (шаги 5–7), чтобы еще больше уменьшить вероятность ненужных затрат.

Положительной стороной такого подхода является построение прогноза использования данных на основе формальной модели анализа временных рядов и возможность динамического наращивания и исключения числа обрабатывающих данные узлов, что обеспечивает предсказательную способность и динамику алгоритма, отрицательной стороной — централизованный расчет модели, что требует дополнительных накладных расходов на поддержание актуальных данных о состоянии модели.

Эвристический подход к избыточному распределению фрагментов базы данных

Подход состоит в распределении фрагментов баз данных так, чтобы для каждого узла, выполняющего запрос доступа к данным, он занимал малое время, при условии исключения избыточности данных в базе данных в целом. В этом случае для более эффективного решения проблемы распределения необходимо не только определить наличие и количество копий каждого фрагмента на каждом узле, но и найти оптимальное распределение по узлам в соответствии с информацией о запросах, исключив избыточность.

```

1: ForAll  $N_i \in \Omega_N$  Do
2:   While  $\hat{W}_{N_i,t}(1) > \theta_u$  Do
3:      $|\Omega_T = \{F_j : F_j \in \Omega_F, a_{ij} = 1 \text{ так, что } |\hat{G}_{N_i,t} - \hat{G}_{F_j,t}| \leq |\hat{G}_{N_i,t}|\};$ 
4:     If  $|\Omega_T| \neq 0$  Then
5:        $F_r \leftarrow \{F_j : F_j \in \Omega_T, \nexists F_l \in \Omega_T, \text{ такого, что } \hat{W}_{F_j,t}(1) > \hat{W}_{F_l,t}(1);\}$ 
6:     Else
7:        $F_r \leftarrow \{F_j : F_j \in \Omega_F, a_{i,j} = 1, \nexists F_l \in \Omega_F, a_{il} = 1 \text{ такого, что } \hat{W}_{F_j,t}(1) > \hat{W}_{F_l,t}(1);\}$ 
8:     EndIf
9:      $\hat{W}_{N_i,t}(1) \leftarrow \hat{W}_{N_i,t}(1) - \hat{W}_{F_r,t}(1);$ 
10:     $\hat{G}_{N_i,t} \leftarrow \hat{G}_{N_i,t} - \hat{G}_{F_r,t}$ 
11:     $\Omega_M \leftarrow \Omega_M \cup \{F_r\}$ 
12:   EndWhile
13: EndFor.

```

Листинг 2. Фаза отбора фрагментов
Listing 2. Fragment selection phase

```

1: Отсортировать  $F_j \in \Omega_M$  по убыванию  $\hat{W}_{F_j,t}(1)$ 
2: While  $\Omega_M \neq 0$  Do
3:    $|F_r = F_j \in \Omega_M$  с наименьшим  $\hat{W}_{F_j,t}(1)$ 
4:    $|\Omega_T = \{F_j : F_j \in \Omega_F, a_{ij} = 1 \text{ так, что } |\hat{G}_{N_i,t} - \hat{G}_{F_j,t}| \leq |\hat{G}_{N_i,t}|\};$ 
5:    $|\Omega_T = \{N_i : N_i \in \Omega_S \text{ так, что } \hat{G}_{N_i,t} + \hat{G}_{F_r,t} \leq 0\};$ 
6:   If  $\Omega_T \neq 0$  Then
7:      $N_d \leftarrow \{N_i : N_i \in \Omega_T, \nexists N_l \in \Omega_T \text{ такого, что } \hat{W}_{N_i,t}(1) > \hat{W}_{N_l,t}(1)\};$ 
8:   Else
9:      $N_d \leftarrow \{N_i : N_i \in \Omega_S, \nexists N_l \in \Omega_S \text{ такого, что } \hat{W}_{N_i,t}(1) > \hat{W}_{N_l,t}(1)\};$ 
10:  EndIf
11:   $\Omega_M \leftarrow \Omega_M - \{F_r\};$ 
12:   $a_{dr} \leftarrow 1;$ 
13:  Перенести  $F_r$  на  $N_d$ ;
14: EndWhile
15: Удалить все пустые узлы.

```

Листинг 3. Фаза миграции фрагментов
Listing 3. Fragment migration phase

Подход работает на модели, основанной на предположении о том, что имеется полносвязная сеть, состоящая из множества узлов $S = \{S_1, S_2, \dots, S_m\}$, каждый из которых имеет объем C , лимит фрагментов FL и локальную БД, являющуюся частью распределенной базы данных. Ребро между двумя узлами S_i и S_j имеет атрибут CC_{ij} , представлено натуральным числом, обозначающим стоимость передачи единицы данных от S_i к S_j .

Каждый узел генерирует множество запросов $Q = \{Q_1, Q_2, \dots, Q_k\}$, рассматриваемых как наиболее часто выполняемые запросы, например 75 %, обрабатываемых в РБД. Каждый запрос Q_k может выполняться с любого узла с определенной частотой. Частоты выполнения k запросов на m узлах могут быть представлены как матрица $m \times k(QF_{ij})$. Пусть S содержит множество фрагментов $F = \{F_1, F_2, \dots, F_n\}$, представляя собой разделение отношений в ходе фазы фрагментации РСБД.

Исходя из критерия оптимальности — минимальной стоимости записи каждого F_i на узел S_k и стоимости выполнения за-

проса F_i на S_k , и стоимости обновления F_i на всех узлах, где этот фрагмент записан, и стоимости передачи данных. Таким образом, решение проблемы оптимальности в соответствии с используемой моделью перераспределения может быть определено как минимизация стоимости связи в процессе переноса фрагмента F_i на узел S_k .

Описание используемых обозначений приведено в табл. 3.

Модель основана на применении данных о частоте запросов на обновление, связанной с распределением фрагментов по узлам. Для правильного распределения схемы каждый фрагмент должен быть прикреплен к одному или более узлам в РСБД на начальном этапе. Поэтому для заданного множества фрагментов разных размеров и количества сетевых узлов с определенным количеством исполняемых запросов проблема оптимального распределения заключается в нахождении распределения, минимизирующего общую стоимость обновления данных во всей сети без нарушения ограничений, накладываемых узлами.

Таблица 3
Обозначения, используемые в эвристической модели

Table 3

Symbols denotations in the heuristic model

Обозначение	Описание
M	Количество узлов РСБД
S_j	j -й узел
N	Количество фрагментов данных в РСБД
F_i	i -й фрагмент данных
K	Количество запросов в РСБД
RN	Количество связей
Q	Массив запросов
Q_k	k -й запрос
$Q_j \#$	Количество запросов узла j
RF_{ij}	Частота обращения запроса на выборку i узла j
UF_{ij}	Частота обращения запроса на обновление i узла j
QF_{ij}	Частота обращения k -го запроса узла j
C_i	Емкость узла j
CC_{ij}	Стоимость передачи данных между узлами i и j
FL_j	Максимальное количество фрагментов данных на узле j
TC	Общая стоимость

В модели подразумевается, что один и тот же запрос на разных узлах воспринимается как разные запросы с разными частотными характеристиками. Однако для достижения оптимального динамического перераспределения используются частоты запросов всех распределенных фрагментов в целях возможного применения при каждом их перераспределении. Чтобы инициировать начальную фазу распределения фрагментов, которая может содержать дублирующие фрагменты, в предлагаемой методике считается, что фрагменты уже распределены по сетевым узлам на основе матрицы поиска RM и матрицы частот запросов QF, сохра-

няя при этом ограничения узлов. Предлагаемый подход заключается в том, что запрашиваемые фрагменты перемещаются на чаще всего запрашивающий их узел, и поэтому запросы будут обрабатываться без каких-либо затрат на передачу данных.

Используемая методика избыточна и применяется с использованием матрицы частот запросов UM, т. к. запрос на обновление данных включает в себя затраты на передачу данных и стоит больше, чем запрос на выборку данных и обработка частотной матрицы. Матрицы UM и QF будут использоваться в качестве входных данных для построения новой матрицы частот обновления фрагментов (Fragment Update Frequency Matrix – FUEM). FUEM может быть определена значениями запросов на обновление на конкретных узлах для вовлеченных фрагментов.

FUEM используется для формирования кумулятивной частотной таблицы обновлений (Cumulative Update Frequency Table – CUFT). С помощью CUFT и DM получается матрица оплаты фрагментов (Fragment Pay Matrix – FUPM). С помощью FUPM определяется узел с максимальной стоимостью обновления для конкретного фрагмента – такой узел является кандидатом на хранение фрагмента. Если имеются какие-либо нарушения ограничений для узла-кандидата, то выбирается следующий по стоимости. Процедура приоритизации (Fragments Priority, FP procedure) будет работать на всех узлах для гарантирования единственности текущего фрагмента.

Процедура FP показывает количество обращений к фрагменту: как много раз запросы каждого узла QS обратились к предполагаемому фрагменту. Используется, если к одному фрагменту F_i осуществляются запросы более чем с одного узла с одинаковыми значениями стоимости обновления этого фрагмента. Для исключения копий FP рассчитывается для каждого узла и используется для принятия решения о перемещении фрагмента F_i на узел с наивысшим FP. Последовательность шагов для перераспределения данных приведена на рис. 2.

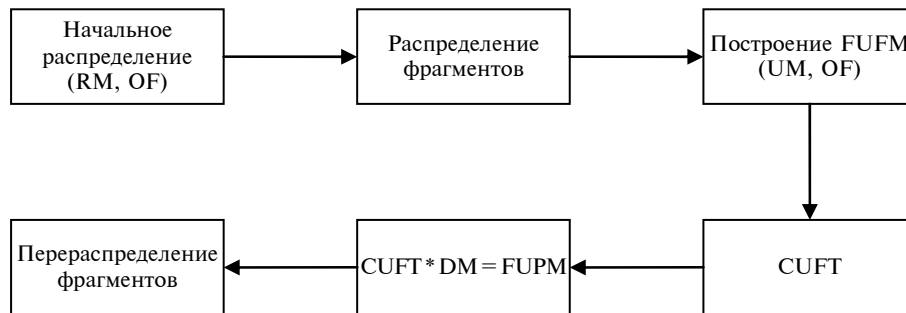


Рис. 2. Эвристический подход для перераспределения данных
Fig. 2. Redistributing data: heuristic approach

$$FP(S_j, F_i) = \sum_{h=1}^k (QF_{hi} \times QS_{hi}), \quad 1 \leq j \leq m;$$

$$QS_{hi} = \sum_{i=1}^n RM_{hi} + \sum_{i=1}^n UM_{hi}, \quad 1 \leq h \leq h.$$

Затраты вызваны запросами на выборку и обновление фрагментов на определенном узле. Для вычисления FUPM берется предыдущее значение FUPM на вход и используются следующие функции стоимостей:

$$CUFT = \sum_{i=1}^n \sum_{j=1}^m (QF_i \times UF(F_j));$$

$$DM_{ij} = \min(CC_{ij}), \quad 1 \leq i, j \leq m;$$

$$FUPM = \sum_{j=1, j \neq j'}^m CUFT(F_j) \times DM_{jj'}, \quad 1 \leq j, j' \leq m;$$

$$\sum_{i=1}^m \max(FUPM_{ij}), \quad 1 \leq i \leq n.$$

Процесс перераспределения может считаться оптимальным при достижении схемы распределения с минимальной стоимостью обновления. Поскольку эта проблема практически невычислима, предлагается эвристический алгоритм (листинг 4).

Предложенный эвристический подход выгодно отличается тем, что обеспечивает сбор появляющихся распределенных фрагментов данных в общий фрагмент, расположенный на одном узле. Это позволяет всегда выполнять операции добавления и

обновления данных локально, не заботясь о сохранении фрагмента в наилучшем узле, что обеспечивает разделение процессов локальной работы системы управления базой данных и средств сетевой синхронизации. Недостаток предложенного метода — сложность реализации распределенной транзакционной модели: откат транзакции потребует перезапуска процедуры перераспределения данных.

Концепция автоматов с памятью, взаимодействующих с внешней средой

Цель применения обучающегося автомата в том, чтобы выбирать оптимальное действие с максимальной вероятностью вознаграждения за счет многократного взаимодействия с внешней средой. Если алгоритм обучения выбран правильно, то итеративно повторяющийся процесс может привести к выбору оптимального действия [13].

Автомат выбирает одно из предлагаемых действий в соответствии с вероятностным вектором, который в любой момент содержит вероятность выбора каждого действия. Выбранное действие активизирует среду, инициируя ее ответ (вознаграждение или штраф), в зависимости от вероятности вознаграждения выбранного действия. Автомат учитывает этот ответ и модифицирует вероятностный вектор с помощью алгоритма обучения. Обучающийся автомат ищет действие с максимальной вероятностью быть вознагражденным и в конечном итоге выбирает это действие чаще, чем другие действия.

- 1: Input:
- 2: Для каждой связи $\{1, \dots, K\}$
- 3: $Q^N = \{Q_1, \dots, Q_n\}$ множество запросов
- 4: $F = \{F_1, \dots, F_n\}$ фрагменты данных в РБД
- 5: $S = \{S_1, \dots, S_m\}$ множество сетевых узлов
- 6: матрицы RM, UM, QF и матрица стоимостей передачи данных
- 7: **Output:**
- 8: Схема перераспределения фрагментов
- 9: **Code:**
- 10: Начальное распределение данных с использованием RM, QF
- 11: Строится таблица $FUPM$ с использованием UM и QF
- 12: Матрица стоимостей расстояний (*Distance Cost Matrix, DM*)=
min(матрица стоимостей передачи данных)
- 13: **ForAll** узел $s \in \{1, \dots, m\}$ **Do** // в $FUPM$, для получения нового $FUPM$
- 14: **ForAll** фрагмент $F_i \in R; 1 \leq i \leq n$
- 15: Вычислить запрос на обновление F_i
- 16: **EndFor**
- 17: **ForAll** узел $s \in \{1, \dots, m\}$ **Do**
- 18: Вычислить $FUPM(F_i)$
- 19: **EndFor**
- 20: Выбрать значение (v) , так что $Pay_v(F_i) = \text{Max}_s^m$ поиск узла с
максимальной стоимостью обновления фрагмента (S_j);
- 21: Перенос F_i на $S_j(v)$;
- 22: **If** $S_j.constraints$ (ограничения на узле S_j) были нарушены **Then**
- 23: Перейти на F_i на S_{j+1} со второй по величине стоимостью
обновления
- 24: **EndIf**
- 25: **If** фрагмент F_i требуется нескольким узлам **Then**
- 26: Выполнить процедуру FP на всех узлах, где требуется F_i
- 27: Перенести фрагмент на S_j с максимальным FP
- 28: **EndIf**
- 29: **EndFor.**

Листинг 4. Эвристический алгоритм перераспределения фрагментов данных
Listing 4. The data redistribution heuristic algorithm

Если РБД состоит из коллекции m узлов $S = \{c_1, c_2, \dots, c_m\}$, где каждый узел i характеризуется его объемом c_i и множество n фрагментов $F = \{s_1, s_2, \dots, s_n\}$, то каждый фрагмент j характеризуется размером s_j . Каждый фрагмент востребован как минимум одним из узлов. Востребованность каждого фрагмента представляется в виде матрицы, где значения на пересечении узлов означают востребованность фрагмента i узлом j , которая обычно обозначается логическими нулем или единицей.

Стоимости передачи данных отражаются в матрице T , где t_{ij} обозначает стоимость до-

ступа узла i к фрагменту, расположенному на узле j . Таким образом, проблема распределения в РБД сводится к задаче поиска оптимального размещения фрагментов на узлах, т. е. поиск размещения $P = \{p_1, p_2, \dots, p_j, \dots, p_n\}$, где равенство $p_j = i$ означает, что фрагмент j расположен на узле i для n фрагментов так, что объем каждого узла не переполнен. Общая стоимость передачи в этом случае

составит $\sum_{i=1}^m \sum_{j=1}^n r_{i,j} \times t_{i,p_j}$, где минимизировано

$$\sum_{j=1}^n r_{i,j} \times s_j \leq c_i \quad \forall i | 1 \leq i \leq m.$$

Ограничивая использование матрицы R и имея нулевую стоимость передачи, проблема распределения данных в РБД сводится к задаче об упаковке в контейнеры, т. е. к NP-полной задаче [9].

В РБД с m узлами и n фрагментами количество решений в области поиска велико. Если использовать обучающийся автомат для решения проблемы распределения фрагментов данных, у автомата будет слишком много возможных действий, что уменьшит скорость нахождения решения в автомате. Для устранения этой проблемы можно применять обучающийся автомат с миграцией объектов [14, 15]. Для решения проблемы используется автомат миграции объектов, обозначаемый как $\langle V, \alpha, \varphi, \beta, F, G \rangle$. В этом автомате $\alpha = \{\alpha_1, \alpha_2, \dots, \alpha_n\}$ представляет собой множество из n возможных действий автомата, причем количество действий равно количеству фрагментов данных. $V = \{V_1, V_2, \dots, V_n\}$ представляет собой множество объектов, которые являются номерами узлов, на которых можно расположить фрагменты данных, а объекты принимают значения 1, 2, ..., m . Объекты перемещаются по различным состояниям автомата и создают новые распределения, $\varphi = \{\varphi_1, \varphi_2, \dots, \varphi_{nN}\}$ — множество состояний, N — глубина памяти автомата. Множество состояний автомата подразделяется на k подмножеств $\{\varphi_1, \varphi_2, \dots, \varphi_N\}, \{\varphi_{N+1}, \varphi_{N+2}, \dots, \varphi_{2N}\}, \dots, \{\varphi_{(k-1)N+1}, \varphi_{(k-1)N+2}, \dots, \varphi_{kN}\}$ так, что объекты классифицируются в терминах их состояний. Если объект u расположен в множестве состояний $\{\varphi_{(j-1)N+1}, \varphi_{(j-1)N+2}, \dots, \varphi_{jN}\}$, то в этом случае фрагмент данных j будет включен в u -й узел. В множестве состояний действия j состояния $\varphi_{(j-1)N+1}$ и φ_{jN} называются соответственно внутренним и граничным состояниями. Объекты, лежащие в $\varphi_{(j-1)N+1}$ и φ_{jN} , называются более и менее определенными объектами. Множество входов автомата $\beta = \{0, 1\}$, где единица и ноль означают отказ и успех

соответственно; $F: \varphi \times \beta \rightarrow \varphi$ обозначает функцию отображения состояний, которая на основании текущего состояния и входов автомата производит следующее состояние, тем самым определяя движение объектов по состояниям автомата.

Функция F различна для различных автоматов. $G: \varphi \rightarrow \alpha$ — выходная функция отображения, решающая, какое действие предпринять в ответ на любое состояние автомата. Если объект u в множестве состояний $\{\varphi_{(j-1)N+1}, \varphi_{(j-1)N+2}, \dots, \varphi_{jN}\}$, выбирается действие j и, следовательно, фрагмент j будет на u -м узле.

Задача распределения реализована в алгоритме, представленном в листинге 5.

Алгоритм начинает работу с создания фрагментов данных со случайным распределением на узлах, с учетом ограничений на емкость узлов для выравнивания нагрузки. Для начала берется несколько узлов, у которых объекты действий автомата находятся в граничном состоянии. Затем выбирается случайный объект i , соответственно, получается либо штраф, либо вознаграждение. Общая стоимость переноса фрагмента данных на заданный узел вычисляется соотношением

$$\text{cost}(\text{data_fragment}) = \sum_{i=1}^m r_{i,\text{data_fragment}} \times t_{i,p,\text{data_fragment}}.$$

Затем пороговое значение вычисляется как средняя общая стоимость передачи фрагментов данных. Если стоимость передачи фрагментов данных меньше или равна пороговому значению, то объект получает вознаграждение, иначе — штраф. Затем эти действия продолжают для другого случайного объекта. Алгоритм продолжает выполняться, пока не выполнится условие остановки. Под условием остановки понимается превышение числа итераций алгоритма выше порога итераций, или достижение оптимального распределения, или локализация всех объектов во внутренних состояниях.

```

1:  $n$  = количество фрагментов в системе;
2:  $m$  = количество узлов в системе;
3: Случайное начальное распределение  $\sigma$  с учетом балансировки
   нагрузки на узлах;
4: Назначить узлы как объекты к граничным состояниям
   соответствующих действий автомата;  $EvalFitness(\sigma)$ ; //  $\sigma$  = начальное
   распределение
5:  $EvalFitness(\sigma)$  // начальное распределение
6:  $iteration = 1$ ;
7: Do
8:   For  $u$  Do //  $1 \leq u \leq n$ ,  $u$  – случайный фрагмент
9:     If  $cost(u) \leq ограничение$  Then
10:       $reward(u)$ 
           //здесь ограничение =  $(\sum_{i=1}^m \sum_{j=1}^n r_{i,j} \times t_{i,p_j}) / n$ 
           //  $cost(data\_fragment) = \sum_{i=1}^m r_{i,data\_fragment} \times t_{i,p_{data\_fragment}}$ 
11:    Else
12:       $penalize(u)$ 
13:    EndIf
14:  EndFor
15:   $Inc(iteration)$ 
16:   $EvalFitness(\sigma)$ 
17: Until ( $iteration = Max\_iterations$  или проход всех объектов в состояниях
   максимальной достоверности).

```

Листинг 5. Процедура распределения данных
Listing 5. The data distribution procedure

Преимуществом автоматного подхода с памятью, основанного на анализе вознаграждения или штрафа является возможность непрерывного анализа текущего состояния распределения фрагментов базы данных и определение наилучшего направления перераспределения за квадратичное время относительно числа узлов, что хорошо работает на базах данных с небольшим числом узлов, в то же время алгоритм является централизованным, что повышает накладные расходы на анализ состояний.

Эволюционный подход к миграции данных для выполнения операций выборки

Эволюционный подход состоит в постепенном перемещении данных, чаще всего участвующих в одном запросе на выборку, на один узел базы данных, в котором он чаще всего используется. Концепция состоит в анализе запросов на выборку

данных с последующей реорганизацией размещения.

В процессе оптимизации размещения рассматриваются две стратегии обработки запроса.

- **MoveSmall**: если бинарная операция для двух отношений, например, объединение, включает в себя два фрагмента данных, расположенных на двух разных узлах, то необходимо доставить меньший фрагмент данных на узел большего фрагмента данных;

- **QuerySite**: отправить все фрагменты данных на узел источника запроса и выполнить запрос.

Цель распределения данных — минимизация общей стоимости передачи данных для обработки всех запросов с использованием одной из перечисленных выше стратегий выполнения. Первая задача распределения данных — максимизация локаль-

ности фрагментов для выполнения запросов. Вторая задача — в нужный момент выбрать стратегию выполнения запросов во время попытки доступа к фрагментам с нескольких узлов и снизить общую стоимость передачи данных для обработки всех запросов.

Рассматривается мультимедийная РБД с m узлами, каждый из которых имеет собственные вычислительные мощности, память и локальную БД. Пусть S_i — название узла i , где $0 \leq i \leq m - 1$. Связь между двумя узлами S_i и $S_{i'}$ (если он существует) имеет связанное с ней натуральное число $c_{ii'}$,

обозначающее стоимость передачи единицы данных из узла S_i на узел $S_{i'}$. Если два узла не связаны напрямую, то стоимость передачи единицы данных будет являться суммой стоимостей связей на выбранном пути от S_i до узла $S_{i'}$. $Q = \{q_0, q_1, \dots, q_{n-1}\}$ — рассматривается как множество наиболее выполняемых запросов — 80 % обработки в РБД. Каждый запрос q_x может выполняться из любого узла S_i с определенной частотой a_{ix} . Частоты выполнения n запросов на m узлах могут быть показаны матрицей $A(m \times n)$. Рассмотрим k фрагментов данных $Q = \{q_0, q_1, \dots, q_{n-1}\}$.

Таблица 4

Обозначения, используемые в эволюционном алгоритме

Table 4

Symbols denotations in the evolutionary algorithm

Обозначение	Описание
m	Количество узлов РСБД
S_i	i -й узел
k	Количество фрагментов данных в РСБД
O_j	j -й фрагмент данных
n	Количество запросов в РСБД
Q	Множество запросов
q_x	x -й запрос
A	Матрица частот доступа
a_{ix}	Частота обращения x -го запроса к узлу i
C	Матрица стоимостей передачи единицы данных
$c_{ii'}$	Стоимость передачи единицы данных от узла i узлу i'
l	Вектор лимитов распределения по узлам
l_i	Лимит распределения для узла i
R	Матрица размеров передаваемых данных запросов
r_{xj}	Размер передаваемых данных фрагмента j запроса x
U	Матрица размеров передаваемых данных узлов
$u_{ij'}$	Размер передаваемых данных фрагмента j узлу j'
D	Матрица зависимостей фрагментов данных
d_{ij}	Размер передаваемых данных от узла фрагмента i узлу фрагмента j
t	Общая стоимость передачи данных

Пусть r_{xj} — размер фрагмента данных O_j , который необходимо передать на узел S_j , откуда был a_{ix} раз за единичный интервал времени инициирован запрос q_x . Соответствующей матрицей является R размера $m \times k$, а матрица U размера $m \times k$ будет содержать в себе u_{ij} , отображающие размеры данных, которые необходимо перенести из узла расположения фрагмента O_j на узел S_i , откуда был инициирован запрос. Таким образом, $u_{ij} = \sum_{x=0}^{n-1} a_{ix} \times r_{xj}$.

В матричном представлении, $U = A \times R$.

В этом случае размер требуемого фрагмента зависит от распределения других фрагментов данных. Пусть $d_{jj'}$ — размер фрагмента данных $O_{j'}$, который необходимо передать на узел, где располагается O_j , для выполнения определенных бинарных операций. Соответствующая матрица размера $k \times k = D$. Всё зависит от обрабатываемого запроса, поэтому для каждого запроса необходима информация о том, как много данных должно быть передано от узла, где располагается один фрагмента данных, на узел с другим фрагментом, с учетом того, что оба фрагмента требуются для выполнения запроса.

Пусть $\delta_{jj'}^x$ — размер данных O_j , необходимых для передачи на узел, где расположен $O_{j'}$, для обработки q_x , соответствующая матрица Δ_x . Количество данных для передачи из узла, на котором располагается O_j , на узел расположения $O_{j'}$ задается следующим образом:

$$d_{jj'} = \sum_{x=0}^{n-1} (\sum_{i=0}^{m-1} a_{ix}) \delta_{jj'}^x.$$

В матричном представлении:

$$D = \sum_{x=0}^{n-1} (\sum_{i=0}^{m-1} a_{ix}) \nabla^x.$$

Пусть $site(O_j)$ обозначает узел, где расположен O_j . Следовательно, общая стоимость передачи t задается как

$$t = \sum_{j=0}^{k-1} \sum_{j'=0}^{k-1} c_{site(O_j), site(O_{j'})} \times d_{jj'} + \sum_{i=0}^{m-1} \sum_{j=0}^{k-1} c_{i, site(O_j)} \times u_{ij},$$

где первое слагаемое обозначает стоимость передачи данных, требующейся для обработки бинарных операций между фрагментами разных узлов, а второе слагаемое — стоимость передачи данных для отправки результатов бинарных операций на узел, инициировавший запрос. Задача распределения данных минимизация t путем изменения функции $site(O_j)$, сопоставляющей фрагмент данных с определенным узлом.

Предлагается четыре эвристических алгоритма в поисковых техниках, хорошо показавших себя в области решения широкого круга задач оптимизации. Первый алгоритм основан на традиционном генетическом алгоритме. Второй — на технологии симулированной эволюции (использование мутаций, скрещиваний и кодирования информации). Третий базируется на технике комбинации нейронных сетей и алгоритма имитации отжига. Четвертый алгоритм основан на технике случайного поиска.

Генетический алгоритм поиска. Генетические методы поиска [16, 17] получены по аналогии с механизмами натуральной генетики, приводящей к выживанию наиболее приспособленных индивидов. ГА манипулируют популяцией потенциальных решений задачи оптимизации.

В предложенном ГА для решения проблемы распределения данных производится кодирование в бинарном представлении назначений каждого фрагмента данных узлам. Эти значения объединяются в одну двоичную строку. Каждая двоичная строка представляет потенциальное решение задачи распределения данных. «Пригодность» (fitness) строки — это стоимость распределения. Механизм селекции реализуется как простая схема пропорционального выбора: строке с пригодностью f выделяется потомство $f / (\bar{f})$, где $\bar{f}$ — среднее значение пригодности популяции. Строке с пригодностью выше среднего значения выделяется более одного потомства, ниже — менее.

Скрещивание (Crossover) — тоже важная операция ГА. Пары строк берутся случайно из популяции и подвергаются скрещиванию. Пусть L — длина строки, тогда алгоритм

случайно выбирает точку скрещивания, принимающую значения в диапазоне от единицы до $L - 1$ с равной вероятностью. Части двух строк за пределами этой точки скрещивания меняются местами, образуя две новые строки. Существует параметр p_c , характеризующий частоту скрещиваний.

После скрещивания строки подвергаются мутации. Под мутацией бита подразумевается его обращение. Так же как и в скрещивании, существует параметр p_m , характеризующий частоту мутаций битов. Биты строки мутируют независимо. Соответственно, мутация бита не подвержена влиянию вероятности мутации других битов.

Сложность распределения с помощью ГА сводится к $O(GN_p(k^2 + km))$, где G – количество поколений, а N_p – размер популяции.

Алгоритм симулированной эволюции. Это вариант традиционного ГА [18, 19]. Имитация отжига, ГА и эволюционные стратегии одинаковы в использовании вероятностных поисковых механизмов, направленных на уменьшение или увеличение целевой ценовой функции. Эти три метода имеют высокую вероятность нахождения оптимального общего решения, несмотря на то, что функция стоимостей мультимедиа имеет несколько локальных оптимальных решений. Однако каждый метод имеет суще-

ственную разницу в режимах работы: имитация отжига вероятно генерирует последовательность состояний, чтобы в конечном итоге сойтись к глобальному оптимуму. Эволюционные стратегии используют мутации как поисковые механизмы и для выбора направления поиска в предполагаемых областях поискового пространства. ГА генерирует последовательность популяции с использованием механизмов селекции и использует скрещивание и мутацию в поисковых механизмах (листинг 6).

Принципиальная разница между генетическим алгоритмом и эволюционной стратегией состоит в том, что первый предполагается на скрещивание, механизм вероятностного и полезного обмена информацией между решениями для нахождения лучшего решения, а последний использует мутации как первичный поисковый механизм (см. листинг 7).

Количество генов в a равно общему лимиту распределения, а количество генов в b равно общему количеству фрагментов.

Для a каждый ген есть один бит. Если бит равен единице, для этой хромосомы разрешено использовать соответствующее пространство распределения. Если бит равен нулю, пространство использовать нельзя. Это уменьшает эффективный лимит распределения для всех узлов.

- 1: Инициализация популяции. Каждый индивид популяции является конкатенацией двоичных представлений случайного начального распределения каждого фрагмента данных;
- 2: Вычислить популяцию;
- 3: `no_of_generation = 0;`
- 4: **While** `no_of_generations < MAX_GENERATION` **Do**
- 5: Выбрать индивидов для следующей популяции;
- 6: Провести скрещивание и мутацию для этих индивидов;
- 7: Вычислить популяцию;
- 8: `no_of_generations ++;`
- 9: **EndWhile**
- 10: Определить конечное распределение путем выбора наиболее подходящего (приспособившегося) индивида. Если конечное распределение невозможно, то необходимо рассмотреть каждый перенаселенный узел и перенести фрагменты на другие узлы с минимальным приростом цены.

Листинг 6. Генетический алгоритм распределения данных
Listing 6. The data distribution genetic algorithm

- 1: Построить первую хромосому на основе входных данных и заставить хромосому сгенерировать начальную популяцию;
- 2: Использовать эвристики отображения для генерации решения по каждой хромосоме;
- 3: Вычислить полученные решения;
- 4: $no_of_generations = 0$;
- 5: **While** $no_of_generations < MAX_GENERATION$ **Do**
- 6: Выбрать хромосомы для следующей популяции;
- 7: Выполнить скрещивание и мутации для этих наборов хромосом;
- 8: Использовать эвристики отображения для генерации решения по каждой хромосоме;
- 9: Вычислить полученные решения;
- 10: $no_of_generations++$;
- 11: **EndWhile**
- 12: Вывод лучшего найденного решения.

Листинг 7. Алгоритм симулированной эволюции для распределения данных
 Listing 7. The stimulated evolution algorithm of data distribution

Для b каждый ген — это целое число, означающее приоритет рассматриваемого фрагмента: чем больше значение, тем выше приоритет.

Используя матрицу стоимости сетевой передачи C частоты доступа b размеров передаваемых данных, мы можем вычислить матрицу стоимостей распределения фрагментов данных по узлам U' , задаваемую как $U' = C \times U$. То есть каждый элемент U' , обозначаемый u_{ij}' , представляет собой стоимость распределения по узлам, возникшую при размещении фрагмента j на узле i .

В алгоритме симулируемой эволюции первая хромосома начальной популяции строится на основании информации u_{ij}' из таблицы. Для каждого фрагмента j вычисляется $x_j = \sum_{i=1}^m u_{ij}'$.

Идея в минимизации стоимости распределения путем задания высокого приоритета фрагменту с большей j . Большое значение j означает, что этот фрагмент затратит больше времени на передачу, т. е. будет больше стоить. Таким образом, просто назначается j в качестве генов для каждой позиции фрагментов в части b первой хромосомы в начальной популяции.

Все гены в части a задаются равными единице для первой хромосомы в начальной популяции, т. к. это предел распределения исходной задачи. Для оставшихся хромосом в начальной популяции гены в a выбраны случайно (ноль или единица). Гены в b являются искажениями соответствующих генов в b у первой хромосомы.

Эвристика отображения состоит в том, что для каждой хромосомы ищется решение, перенося фрагмент j с наивысшим приоритетом на узел i , таким образом, чтобы u_{ij}' был наименьшим для всех u_{ij}' . $1 < x < m$. Если лимит распределения, включенный в гены части a хромосомы этого узла, превышен, т. е. узел переполнен, то этот фрагмент переносится на узел со следующим наименьшим значением u_{ij}' для всех u_{ij}' , $1 < y < m$, $y \neq x$. Фрагмент гарантированно будет распределен на какой-нибудь узел, т. к. было проверено, что общий лимит распределения больше или равен общему количеству фрагментов для каждого поколения хромосомы.

Для каждой хромосомы функция стоимости — это общая стоимость передачи после распределения всех фрагментов на некоторые узлы с использованием эвристики отображения. Вычисление этой стоимости передачи учитывает зависимости между

фрагментами и размеры фрагментов, необходимые для каждого узла. Значение пригодности для хромосомы вычисляется как

$$f(i) = \frac{(\text{MaxCost} - \text{Cost}(i))^{\tau}}{\sum_{i=1}^{N_p} (\text{MaxCost} - \text{Cost}(i))^{\tau}},$$

где N_p — размер популяции; MaxCost — максимальная стоимость среди хромосом в популяции; τ — коэффициент сходимости, использующийся для контролирования скорости сходимости. Сложность алгоритма составляет $O(GP(k^2 + km))$.

Алгоритм отжига среднего поля. Техника отжига среднего поля (Mean Field Annealing — MFA) [3, 5] комбинирует свойство коллективного вычисления нейронной сети Хопфилда (Hopfield Neural Network — HNN) с техникой имитации отжига. MFA первоначально была создана для решения задачи коммивояжера как альтернатива HNN, недостаточно масштабируемой для крупных задач. Доказано, что MFA является общим подходом [3, 4], который может быть применен к различным задачам комбинаторики.

Алгоритм MFA [3] получен из аналогии с моделью спинов Изинга, использующейся для оценки состояния системы частиц или спинов в тепловом равновесии. В этой модели энергия системы с S спинами имеет вид:

$$H(s) = \frac{1}{2} \sum_{x=1}^S \sum_{y \neq x} \beta_{xy} s_x s_y + \sum_{x=1}^S h_x s_x,$$

где β_{xy} представляет собой уровень взаимодействия между спином x и y ; $s_x \in \{1, 0\}$ — значение спина x . Предполагается, что $\beta_{xy} = \beta_{yx}$ и $\beta_{xx} = 0$ для $1 \leq x, y \leq S$. В тепловом равновесии среднее значение $\langle s_x \rangle$ спина x может быть вычислено с помощью распределения Больцмана:

$$\langle s_x \rangle = \frac{1}{1 + e^{-\Phi_x/T}},$$

где $\Phi_x = \langle H(s) \rangle|_{s_x=0} - \langle H(s) \rangle|_{s_x=1}$ представляет собой среднее поле, действующее на спин x , где средняя энергия системы $\langle H(s) \rangle$:

$$\langle H(s) \rangle = \sum_{x=1}^S \sum_{y \neq x} \beta_{xy} \langle s_x s_y \rangle + \sum_{x=1}^S h_x \langle s_x \rangle.$$

Сложность вычисления Φ_x с использованием этого уравнения экспоненциальная. Однако для большого числа спинов приближение среднего поля может использоваться для вычисления средней энергии:

$$\langle H(s) \rangle = \frac{1}{2} \sum_{x=1}^S \sum_{y \neq x} \beta_{xy} \langle s_x \rangle \langle s_y \rangle + \sum_{x=1}^S h_x \langle s_x \rangle.$$

Следовательно, $\langle H(s) \rangle$ линейна в $\langle s_x \rangle$, среднее поле Φ_x может быть вычислено через

$$\begin{aligned} \Phi_x &= \langle H(s) \rangle|_{s_x=0} - \langle H(s) \rangle|_{s_x=1} = \\ &= \frac{\partial \langle H(s) \rangle}{\partial \langle s_x \rangle} = - \left(\sum_{y \neq x} \beta_{xy} \langle s_y \rangle + h_x \right). \end{aligned}$$

Таким образом, сложность вычисления Φ_x сводится к $O(S)$.

При каждой температуре, начиная с начальных средних спинов, вычисляется среднее поле x , действующее на случайно выбранный спин. После этого обновляется среднее значение спина. Этот процесс повторяется для случайной последовательности спинов, пока система не стабилизируется на текущей температуре (см. листинг 8).

Проблема распределения данных формулируется через MFA следующим способом (см. листинг 9). Вместо линейного массива спинов используется $x \times m$ матрица спинов s_{ij} для кодирования распределения фрагментов данных по узлам (x фрагментов данных по m узлам). Единица в записи означает, что фрагмент расположен на соответствующем узле. Правильное распределение — такое, когда в каждой строке матрицы спинов есть только одна единица. Каждая переменная спина непрерывна в диапазоне $[0, 1]$. Значения спинов сходятся к единице или нулю в фиксированной точке. Таким образом, функция энергии в интерпретации стоимости передачи данных для задачи распределения данных может быть формализована следующим образом:

$$E(s) = \sum_{i=0}^{k-1} \sum_{i'=0}^{k-1} \sum_{j=0}^{m-1} \sum_{j'=0}^{m-1} c_{ij'} d_{i'j} s_{ij'} s_{i'j} + \sum_{j=0}^{m-1} \sum_{i=0}^{k-1} u_{ji} s_{ji}.$$

- 1: Получить начальную температуру T_0 , задать $T = T_0$;
- 2: Инициализировать средние значения спина $\langle s \rangle = [\langle s_1 \rangle, \dots, \langle s_s \rangle]$;
- 3: **While** T в диапазоне охлаждения **Do**
- 4: **While** система не стабилизирована на текущей температуре **Do**
- 5: Выбрать случайный спин x ;
- 6: Вычислить x ;
- 7: Обновить $\langle s_x \rangle$;
- 8: **EndWhile**
- 9: Обновить T в соответствии с расписанием охлаждения;
- 10: **EndWhile.**

Листинг 8. Общий алгоритм MFA

Listing 8. General MFA algorithm

- 1: Получить начальную температуру T_0 , задать $T = T_0$;
- 2: Инициализировать средние значения спина $s = [s_{00}, s_{01}, \dots, s_{k-1, m-1}]$,
каждый s_{ij} инициализирован как случайное число между 0 и 1;
- 3: **While** температура T в диапазоне охлаждения **Do**
- 4: **While** E уменьшается **Do**
- 5: Выбрать случайный фрагмент данных i ;
- 6: Вычислить среднее поле спинов $(\Phi_{ij}, \forall j)$ в строке i ;
- 7: Вычислить сумму $\sum_{j'=0}^{m-1} e^{\Phi_{ij}/T}$;
- 8: Вычислить новые значения спина в строке i ;
- 9: Вычислить изменения энергии исходя из вышеуказанных вычислений;
- 10: **EndWhile**
- 11: Обновить температуру T в соответствии с расписанием охлаждения;
- 12: **EndWhile**
- 13: Определить окончательное распределение, помещая каждый фрагмент данных на узел с максимальным значением спина.

Листинг 9. Алгоритм MFA распределения данных

Listing 9. The MFA algorithm of data distribution

Используя приближение среднего поля, выражение для Φ_{ij} , влияющего на спин s_{ij} , становится:

$$\Phi_{ij} = \frac{\partial E(s)}{\partial s_{ij}} = - \sum_{i' \neq i} \sum_{j' \neq j} c_{ij'} d_{i'j'} s_{i'j'} - u_{ji}.$$

Сумма спинов каждой строки должна быть равна единице, т. е. каждый фрагмент данных должен быть распределен исключительно на один узел. Этого можно достичь, нормализуя каждый спин s_{ij} :

$$s_{ij} = \frac{e^{\Phi_{ij}/T}}{\sum_{j'=0}^{m-1} e^{\Phi_{ij'}/T}}.$$

Если окончательное распределение невозможно, то необходимо рассмотреть

каждый перегруженный узел, чтобы перенести фрагменты данных на другие узлы с наименьшим приростом стоимости. Общая сложность алгоритма MFA $O(BK(k^2 + km))$.

Алгоритм случайного поиска. Используется не сложная, но эффективная оптимизационная техника, называемая поиском случайной окрестности (см листинг 10). Основная идея этого алгоритма – сгенерировать начальное решение среднего качества. Затем, с учетом предопределенной окрестности, алгоритм вероятностно выбирает и тестирует соседнее решение в области поиска на предмет того, лучше оно или нет. Если лучше, то алгоритм принимает его и начинает поиск в новых окрестностях, если нет, выбирается другое решение.

```
1: Используется отдельная кластеризация [19] для поиска начального
   распределения Initial_Alloc;
2: Best_Alloc = Initial_Alloc;
3: New_Alloc = Best_Alloc; iteration = 0;
4: Do
5:   searchstep = 0; counter = 0;
6:   Do
7:     Выбрать случайно два узла из New_Alloc;
8:     Выбрать случайно два фрагмента данных из каждого узла;
9:     Поменять их местами;
10:    If стоимость уменьшилась Then
11:      принять изменения, counter = 0;
12:    Else
13:      отменить их, counter ++;
14:    EndIf
15:  Until ++ searchstep > MAX_STEP || counter > MARGIN;
16:  If cost(New_Alloc) < cost(Best_Alloc) Then
17:    Best_Alloc = New_Alloc;
18:  EndIf
19:  Случайно обменять два фрагмента данных из двух отдельных
   случайных узлов из New_Alloc; /* Вероятностный прыжок */
20: Until iteration > MAX_ITERATION.
```

Listing 10. Алгоритм случайного поиска (Random Search (RS))

Listing 10. The random search algorithm

Алгоритм останавливается после определенного количества шагов или после фиксированного количества шагов, в ходе которых решение не улучшалось. Качество решения этой техники в значительной степени зависит от выбора и построения окрестностей.

Заключение

В данной обзорной статье рассмотрены подходы к решению задачи динамического оптимального размещения фрагментов данных распределенной базы данных, основанной на задаче глобальной минимизации времени выполнения запросов всех абонентов на всех узлах системы. Предложенные решения исключают полный перебор вариантов принятием дополнительных допущений, к которым относятся: оптимизация размещения только в установившихся режимах функционирования системы относительно частотного распределения запросов, ранжирование запросов по частоте и отсеивание редко используемых, ранжирование запросов по времени исполнения и оптимизация самых медленных, выделение наиболее ча-

сто используемых и длительно выполняемых запросов.

Рассмотренные подходы реализуют различные технологические приемы к решению задачи оптимизации относительно двух критериев: времени доступа к данным и объема хранимых данных на каждом узле. К ним относятся задача минимизации суммарного времени доступа ко всем фрагментам, как в подходе с пороговой оценкой, концепция локальной оценки и минимизации путем выполнения распределенных «торгов» по технологии Mariposa, формирование иерархии кластеров для реализации распределения данных между кластерами в венгерском алгоритме, применение методов прогнозирования будущей нагрузки методами авторегрессии и скользящего среднего для распределения на основе прогнозирования, применение эвристики для оценки стоимости пути на полносвязном графе с выделением путей максимальной стоимости, автоматные подходы, состоящие в изменении состояния автомата для получения наиболее выгодного состояния системы, эволюционные алгоритмы, реализующие эффективные модификации генетических алгоритмов.

Таблица 5

Сравнительная характеристика подходов и алгоритмов к перераспределению данных
в распределенных базах данных

Table 5

The comparison of approaches and algorithms for the redistribution of data in distributed databases

Подход	Ретроспектива	Прогноз	Анализи- руемые операции	Наличие центра- лизованного брокера	Алгоритм оценки перераспреде- ления
Пороговой оценки	Да, плавающим окном по времени	Нет	Выборка, добавление	Нет	Оригинальный
Распределен- ных торгов	Да, локальная на основе данных узла	Нет	Выборка добавление	Да, выборы по- бедителя торгов	Оригинальный
Одноранговых сетей	Нет	Нет	Выборка	Да, требует иерар- хии брокетов	Венгерский
Прогнозирова- ния состояния	Нет	Да, ARIMA	Выборка	Да, обеспечивает изменение числа узлов	Оригинальный
Избыточное распределение	Нет	Нет	Выборка	Да	Оригинальный
Автоматный подход	Да, на основе ав- томатов с памятью	Нет	Выборка, добавле- ние	Да	Оригинальный
Эволюцион- ный подход	Нет	Нет	Выборка	Да	Генетические ал- горитмы: поиска, симулированной эволюции, сред- него поля, слу- чайного поиска

Результаты анализа подходов и применяе-
мых в них алгоритмов приведены в табл. 5.

Предложенный обзор алгоритмов может
использоваться для реализации эффективных
методов распределения данных в системах
управления распределенными базами данных.

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

СПИСОК ЛИТЕРАТУРЫ

1. Kumar R., Gupta N. An extended approach to non-replicated dynamic fragment allocation in distributed database systems // IEEE Internat. Conf. on Issues and Challenges in Intelligent Computing Techniques. 2014. Pp. 861–865.
2. Nilarun Mukharjee Non-replicated dynamic fragment allocation in distributed database systems // CCSIT 20 II, Part I. Berlin Heidelberg: Springer-Verlag, 2011. CCIS 131. Pp. 560–569.

3. Ceri S., Pelagatti G. Distributed databases: principles systems. McGraw-Hill Internat. Editions, 1985.
4. Horea-Adrian Grebla, Anca Gog Redesign based optimization for distributed databases // Studia univ. Babe. bolyai, informatica. 2005. Vol. 1. No. 1.
5. Agrawal S., Narasayya V., Yang B. Integrating vertical and horizontal partitioning into automated physical database design // Proc. 2004 ACM

SIGMOD Internat. Conf. Management of Data. 2004. Pp. 359–370.

6. **Chu W.W.** Optimal file allocation in multiple computer systems // *IEEE Transaction on Computers*. 1969. Vol. C-18. No. IO.

7. **Ozsu M., Valduriez P.** Principles of distributed database systems. 2nd ed. Prentice Hall, 1999.

8. **Raju Kumar, Neena Gupta** Non-redundant dynamic data allocation in distributed database systems // Special Issue of Internat. J. of Computer Applications, on Issues and Challenges in Networking, Intelligence and Computing Technologies. 2012. Pp. 6–10.

9. **Singh A., Kahlon K.S.** Non-replicated dynamic data allocation in distributed database systems // *Internat. J. of Computer Science and Network Security*. 2009. Vol. 9. No. 9.

10. **Kumar R., Gupta N.** Dynamic data allocation in distributed database systems: A systematic survey // *Internat. Review on Computers and Softwares*. 2013. Vol. 8. No. 2. Pp. 660–667.

11. **Brunstroml A., Leutenegger S.T., Simhal R.** Experimental evaluation of dynamic data allocation strategies in a distributed database with changing workload // *ACM Trans. Database Systems*. 1995.

12. **Eswaran K.P.** Placement of records in a file and file allocation in a computer network // *Proc. IFIP Congr. North-Holland*, 1974.

13. **Ulus T., Uysal M.** Heuristic approach to dynamic data allocation in distributed database systems // *Pakistan J. of Information and Technology* 2003. No. 2(3). Pp. 231–239.

14. **Sarathy R., Shetty B., Sen A.** A constrained nonlinear 0-1 program for data allocation // *European J. Operational Research*. 1997. Vol. 102. Pp. 626–647.

15. **Hababeh I.O., Ramachandran M., Bowring N.** A high-performance computing method for data

allocation in distributed database systems // *J. Supercomput.* Springer, 2007. No. 39. Pp. 3–18.

16. **Chiu G., Raghavendra C.** A model for optimal database allocation in distributed computing systems // *Proc. IEEE INFOCOM 1990*. Vol. 3. Pp. 827–833.

17. **Ram S., Narasimhan S.** Database allocation in a distributed environment: Incorporating a concurrency control mechanism and queuing costs // *Management Science*. 1994. Vol. 40. No. 8. Pp. 969–983.

18. **Huang Y., Chen J.** Fragment allocation in distributed database design // *J. Information Science and Eng.* 2001. Vol. 17. Pp. 491–506.

19. **Tamhankar A., Ram S.** Database fragmentation and allocation: A integrated methodology and case study // *IEEE Trans. Systems, Man and Cybernetics*. 1998. Part A. Vol. 28. No. 3.

20. **Wolfson O., Jajodia S., Huang Y.** An adaptive data replication algorithm // *ACM Trans. Database Systems*. 1997. Vol. 22. No. 2. Pp. 255–314.

21. **Zou L., Peng P.** A survey of distributed RDF data management // *Jisuanji Yanjiu Yu Fazhan/Computer Research and Development*. 2017. No. 54(6). Pp. 1213–1224.

22. **Eldawy A., Mokbel M.F.** The era of big spatial data // 31st IEEE Internat. Conf. on Data Engineering Workshops. Seoul, 2015. Pp. 42–49.

23. **Papastavrou Stavros, Chrysanthi Panos, Samaras George, Pitoura Evangelia.** An evaluation of the JAVA-BASED approaches to web database access // *Internat. J. of Cooperative Information Systems*. 2000. No. 10.

24. **Jun-Lin Lin, Dunham M.H.** A survey of distributed database check pointing // *Distrib. Parallel Databases*. 1997. Vol. 5. No. 3. Pp. 289–319.

Статья поступила в редакцию 27.11.2018.

REFERENCES

1. **Kumar R., Gupta N.** An extended approach to non-replicated dynamic fragment allocation in distributed database systems. *IEEE International Conference on Issues and Challenges in Intelligent Computing Techniques*, 2014, Pp. 861–865.

2. **Nilarun Mukharjee** Non-replicated dynamic fragment allocation in distributed database systems. *CCSIT 20 II, Part I*, Berlin Heidelberg: Springer-Verlag, 2011, CCIS 131, Pp. 560–569.

3. **Ceri S., Pelagatti G.** *Distributed databases: principles systems*. McGraw-Hill International Editions, 1985.

4. **Horea-Adrian Grebla, Anca Gog** Redesign based optimization for distributed databases.

Studia univ. Babe_bolyai, informatica. 2005, Vol. 1, No. 1.

5. **Agrawal S., Narasayya V., Yang B.** Integrating vertical and horizontal partitioning into automated physical database design. *Proc. 2004 ACM SIGMOD International Conf. Management of Data*, 2004, Pp. 359–370.

6. **Chu W.W.** Optimal file allocation in multiple computer systems. *IEEE Transaction on Computers*, 1969, Vol. C-18, No. IO.

7. **Ozsu M., Valduriez P.** *Principles of distributed database systems*. 2nd ed. Prentice Hall, 1999.

8. **Raju Kumar, Neena Gupta** Non-redundant dynamic data allocation in distributed database sys-



tems. *Special Issue of International Journal of Computer Applications, on Issues and Challenges in Networking, Intelligence and Computing Technologies*, Nov. 2012, Pp. 6–10.

9. **Singh A., Kahlon K.S.** Non-replicated dynamic data allocation in distributed database systems. *International Journal of Computer Science and Network Security*, 2009, Vol. 9, No. 9.

10. **Kumar R., Gupta N.** Dynamic data allocation in distributed database systems: A systematic survey. *International Review on Computers and Softwares*, 2013, Vol. 8, No. 2, Pp. 660–667.

11. **Brunstroml A., Leutenegger S.T., Simhal R.** Experimental evaluation of dynamic data allocation strategies in a distributed database with changing workload. *ACM Trans. Database Systems*, 1995.

12. **Eswaran K.P.** Placement of records in a file and file allocation in a computer network. *Proc. IFIP Congr. North-Holland*, 1974.

13. **Ulus T., Uysal M.** Heuristic approach to dynamic data allocation in distributed database systems. *Pakistan Journal of Information and Technology*, 2003, No. 2(3), Pp. 231–239.

14. **Sarathy R., Shetty B., Sen A.** A constrained nonlinear 0-1 program for data allocation. *European J. Operational Research*, 1997, Vol. 102, Pp. 626–647.

15. **Hababeh I.O., Ramachandran M., Bowring N.** A high-performance computing method for data allocation in distributed database systems. *J. Supercomput.* Springer, 2007, No. 39. Pp. 3–18.

16. **Chiu G., Raghavendra C.** A model for optimal database allocation in distributed computing

systems. *Proc. IEEE INFOCOM 1990*, Vol. 3, Pp. 827–833.

17. **Ram S., Narasimhan S.** Database allocation in a distributed environment: Incorporating a concurrency control mechanism and queuing costs. *Management Science*, 1994, Vol. 40, No. 8, Pp. 969–983.

18. **Huang Y., Chen J.** Fragment allocation in distributed database design. *J. Information Science and Eng.*, 2001, Vol. 17, Pp. 491–506.

19. **Tamhankar A., Ram S.** Database fragmentation and allocation: A integrated methodology and case study. *IEEE Trans. Systems, Man and Cybernetics*, 1998, Part A, Vol. 28, No. 3.

20. **Wolfson O., Jajodia S., Huang Y.** An adaptive data replication algorithm. *ACM Trans. Database Systems*, 1997, Vol. 22, No. 2, Pp. 255–314.

21. **Zou L., Peng P.** A survey of distributed RDF data management. *Jisuanji Yanjiu Yu Fazhan/Computer Research and Development*, 2017, No. 54(6), Pp. 1213–1224.

22. **Eldawy A., Mokbel M.F.** The era of big spatial data. *31st IEEE International Conference on Data Engineering Workshops*, Seoul, 2015, Pp. 42–49.

23. **Papastavrou Stavros, Chrysanthis Panos, Samaras George, Pitoura Evaggelia.** An evaluation of the JAVA-BASED approaches to web database access. *International Journal of Cooperative Information Systems*, 2000, No. 10.

24. **Jun-Lin Lin, Dunham M.H.** A survey of distributed database checkpointing. *Distrib. Parallel Databases*, 1997, Vol. 5, No. 3, Pp. 289–319.

Received 27.11.2018.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

ПОПОВ Сергей Геннадьевич

POPOV Sergey G.

E-mail: popovserge@spbstu.ru

ФРИДМАН Виктор Сергеевич

FRIDMAN Viktor S.

E-mail: essllesse@gmail.

DOI: 10.18721/JCSTCS.11408
УДК 004

РАЗРАБОТКА АППАРАТНО-РЕКОНФИГУРИРУЕМОГО ВЫЧИСЛИТЕЛЯ С ПОДДЕРЖКОЙ СТАНДАРТА OPENCL

А.П. Антонов, А.С. Филиппов, И.О. Киселев

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

Наиболее важными параметрами современных суперкомпьютерных систем становятся отношение производительности к энергопотреблению и эффективная производительность (отношение реальной производительности к пиковой производительности). Аппаратно-реконфигурируемые вычислители, обеспечивая на аппаратном уровне возможность перестраивать структуру вычислителя под решаемую задачу, способны повысить эффективность суперкомпьютерных систем в соответствии с указанными параметрами. Аппаратно-реконфигурируемые вычислители предполагают использование современной элементной базы, средств высокоуровневого синтеза и соответствующих средств разработки. Существующие на рынке аппаратно-реконфигурируемые вычислители позволяют строить высокопроизводительные гетерогенные суперкомпьютерные системы. В то же время для подготовки специалистов требуется массовое использование подобных платформ в учебных лабораториях, что определяет необходимость разработки бюджетного варианта аппаратно-реконфигурируемого вычислителя, имеющего полную поддержку со стороны современных средств проектирования. В статье показана необходимость создания подобной платформы, приведены параметры и особенности разработанного на базе программируемой логики фирмы Xilinx вычислителя, имеющего полную совместимость со стандартом OpenCL и интегрированного в среду разработки SDAccel.

Ключевые слова: реконфигурируемые вычислительные системы, высокопроизводительные вычисления, ПЛИС, OpenCL, высокоуровневый синтез аппаратуры, образовательные технологии.

Ссылка при цитировании: Антонов А.П., Филиппов А.С., Киселев И.О. Разработка аппаратно реконфигурируемого вычислителя с поддержкой стандарта OpenCL // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2018. Т. 11. № 4. С. 108–118. DOI: 10.18721/JCSTCS.11408.

DESIGN OF RECONFIGURABLE COMPUTER SUPPORTING OPENCL STANDARD

A.P. Antonov, A.S. Filippov, I.O. Kiselev

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

The most significant parameters of modern supercomputing systems are performance to power consumption rate and effectiveness (real performance to peak performance rate). Hardware-reconfigurable computers can provide real-time hardware adaptation according

to the specific task, improving the given parameters. Reconfigurable computing uses cutting-edge hardware and high level synthesis tools for parallel computations. For this reason, skilled engineers are required for managing such systems. The existing reconfigurable high-performance computing modules allow building heterogeneous supercomputer systems. However, they are too expensive and unsuitable for education purposes. In this paper, we have confirmed the need for cheap reconfigurable computers that could be used as educational facilities for students and described the significant features of the designed OpenCL-compatible reconfigurable platform.

Keywords: reconfigurable computers, high-performance computing, FPGA, OpenCL, high-level synthesis, educational technologies.

Citation: Antonov A.P., Filippov A.S., Kiselev I.O. Design of reconfigurable computer supporting OpenCL standard. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2018, Vol. 11, No. 4, Pp. 108–118. DOI: 10.18721/JCSTCS.

Введение. В настоящее время мир высокопроизводительных вычислений базируется на гибридных вычислительных системах, в которых кроме массивов процессоров общего назначения различной структурной организации имеются специализированные вычислительные узлы SIMD (Single Instruction Multiple Data), реализованные на базе технологии CUDA. Однако энергоэффективность и эффективная производительность таких систем чувствительна к структурной организации алгоритмов решаемых задач: ее соответствию архитектуре суперкомпьютерной системы [1].

Современным подходом к увеличению энергоэффективности и эффективной производительности суперкомпьютерных систем является реализация гетерогенных систем, предполагающих использование аппаратно-реконфигурируемых вычислителей, аппаратная структура которых в режиме реального времени подстраивается под структурную организацию алгоритмов решаемых задач [2]. Основа для реализации аппаратно-реконфигурируемых вычислителей – современные СБИС Программируемой Логике (FPGA), выпускаемые компаниями Intel [3] и Xilinx [4].

Задачи создания аппаратуры мультипроцессорных систем и CUDA-ускорителей решены: подобные изделия выпускаются серийно. Аппаратно-реконфигурируемые вычислители – новый и активно развивающийся класс вычислительных устройств. Разработка аппаратуры и интеграция подобного вычислителя в гетерогенную вычисли-

тельную систему является нетривиальной задачей, т. к. требует использования современных FPGA и специального программного обеспечения. Кроме того, для создания, внедрения и эксплуатации подобных систем требуются подготовленные инженеры. Из этого следует, что для развития отрасли высокопроизводительных вычислений актуальны задачи создания реконфигурируемых вычислительных комплексов, учебных лабораторий и соответствующих курсов обучения.

Стандарт OpenCL. Для создания специализированных вычислительных устройств чаще всего используют FPGA. При этом для того, чтобы реализовать требуемый алгоритм на базе FPGA и создать конфигурационный файл требовалось использовать языки описания аппаратуры и вручную переходить от описания алгоритма на языках высокого уровня к языку описания аппаратуры. Такой переход требует от инженера знания всех особенностей аппаратной платформы, языка высокого уровня (например C++ или MATLAB) и предметной области реализуемой задачи.

По этой причине, несмотря на удачные примеры применения FPGA в области суперкомпьютерных вычислений, широкого распространения реконфигурируемые вычислители не получили [6]. Для упрощения процесса проектирования и формализации процесса перехода от алгоритмической симуляции к аппаратной реализации появились средства высокоуровневого синтеза аппаратуры, т. е. преобразование текста программы, написанного на языках высокого уровня, в

язык описания аппаратуры. В настоящее время подобные средства предоставляются производителями FPGA: Xilinx и Intel.

Попытки создания средств высокоуровневого синтеза, т. е. программного обеспечения, которое в автоматизированном режиме преобразует алгоритм, описанный на языках высокого уровня абстракции, в описание на языках описания аппаратуры, которое было бы функционально тождественным исходному алгоритму, предпринимались неоднократно, некоторые из них были весьма успешными [7]. В России также ведутся разработки собственных методик высокоуровневого синтеза. В [8] описан специализированный язык высокого уровня для разработки реконфигурируемых вычислительных систем – COLAMO.

Долгое время подобные проекты не поддерживались производителями FPGA. Однако ситуация изменилась с появлением стандарта OpenCL – кроссплатформенным расширением языка C++, позволяющим писать одноплатформенный код для выполнения параллельных вычислений для различного типа платформ (CPU, GPU, FPGA). OpenCL активно используется при разработке программного обеспечения для суперкомпьютеров и систем обработки больших данных и позволяет, не внося значительных изменений в описание алгоритма, быстро заменить один тип вычислителя на другой [9, 10].

В [11] показано, что использование методов высокоуровневого синтеза позволяет сократить время разработки, тестирования и интеграции платформы на основе FPGA в гетерогенную вычислительную систему до шести раз.

Результаты предыдущих работ. Ранее совместно с ООО «СТЦ» была спроектирована вычислительная реконфигурируемая платформа для высокопроизводительных вычислений и достигнуты соглашения об интеграции ее в СКЦ «Политехнический» [2]. Данная платформа выполнена в формате PCIe-x16 карты расширения, интегрирована в средства проектирования, предоставляемые производителем FPGA, и поддерживается в рамках стандарта OpenCL [5]. Каждый мо-

дуль содержит 4 FPGA Kintex UltraScale (XCKU115-2FLVA1517) фирмы Xilinx (каждая из которых имеет порядка 660 тыс. логических ячеек и 5,5 тыс. блоков умножения-сложения), а также по 4 Гб DDR3 памяти

Для эффективного использования доступных вычислительных ресурсов и определения целесообразности применения аппаратно-реконфигурируемой платформы вместо традиционных массивов процессоров общего назначения или CUDA-ускорителей для решения отдельно взятой задачи необходимы теоретические знания в области архитектуры реконфигурируемых вычислительных устройств и опыт практического использования средств разработки программного обеспечения. Для этого требуется создание учебно-лабораторного комплекса, в котором можно проводить подготовку и переобучение специалистов.

Себестоимость единицы разработанной платформы оценивается в 35 тыс. долл. США. Установка подобного оборудования является целесообразной для суперкомпьютерных центров и иного рода высокопроизводительных кластеров. Для задач обучения студентов, а также переподготовки специалистов подобные вычислительные мощности являются избыточными, а стоимость – слишком высокой для установки даже одной платформы в учебной лаборатории.

В связи с этим возникла потребность в создании аппаратно-реконфигурируемой платформы, представляющей собой полнофункциональное с точки зрения инструментария разработчика устройство, при этом имеющее достаточно низкую стоимость для использования в рамках учебного процесса.

Постановка задачи. В связи с отсутствием на рынке недорогих учебных платформ для реализации проектов, основанных на стандарте OpenCL, потребовалось исследовать возможность реализации реконфигурируемого вычислителя с поддержкой OpenCL на устройствах бюджетных линеек FPGA Xilinx, а в случае успеха – разработать и интегрировать в среду разработки SDAccel компании Xilinx бюджетный аппаратно-реконфигурируемый вычислитель с под-

держкой стандарта OpenCL, предназначенный для массового использования в учебном процессе.

В процессе разработки были установлены следующие требования:

- формат — карта расширения PCIe;
- логический ресурс FPGA — достаточный для реализации интерфейсов взаимодействия с ПК и внешней памятью типа DDR3, а также для реализации вычислительных ядер;
- поддержка ПО — платформа должна быть интегрирована в среду разработки, предоставляемую производителем FPGA, и поддерживаться средствами высокоуровневого синтеза, в частности, быть совместимой со стандартом OpenCL [5];
- стоимость производства не должна превышать 500 долл. США.

Для реализации поставленной задачи требуется:

- определить нижнюю границу количества доступных аппаратных ресурсов FPGA (таким образом формируется критерий выбора подходящей элементной базы: минимальная стоимость при достаточных ресурсах);
- определить необходимые для работы устройства периферийные элементы на плате;

- сравнить основные параметры.

Реализацию динамически-реконфигурируемых вычислителей обеспечивает технология частичной реконфигурации, позволяющая разделить конфигурационный файл на несколько независимых частей, каждая из которых отвечает за свою область на кристалле.

Данная технология применяется для того, чтобы часть логических ресурсов FPGA, реализующая PCIe-контроллер, оставалась неизменной, т. к. восстановление связи с отключившимся во время конфигурации устройством возможно восстановить только при перезагрузке системы; часть логических ресурсов FPGA, реализующая алгоритм, решающий требуемую задачу, реконфигурировалась в процессе решения прикладных задач; реализовать передачу конфигурации через PCIe-интерфейс, как наиболее быстрый способ.

Анализ минимальных требований к FPGA. Основываясь на опыте разработанной ранее платформы, были получены результаты по количеству занимаемых статической частью ресурсов. Они представлены в табл. 1. Помимо этого потребовалась поддержка аппаратных контроллеров PCIe и DDR3 памяти.

Таблица 1

Расход ресурсов FPGA для статической области для микросхемы XCKU115

Table 1

FPGA resources utilization by static area of XCKU115 IC

Тип ресурса	Использовано	Доступно	Использовано, %
LUT	84900	663360	13
LUTRAM	11733	293760	4
FF	120420	1326720	8
BRAM	177	2160	8
DSP	3	5520	0
IO	128	624	21
GT	8	48	17
BUFG	20	1248	2
MMCM	3	24	13
PLL	4	48	8
PCIe	1	6	17

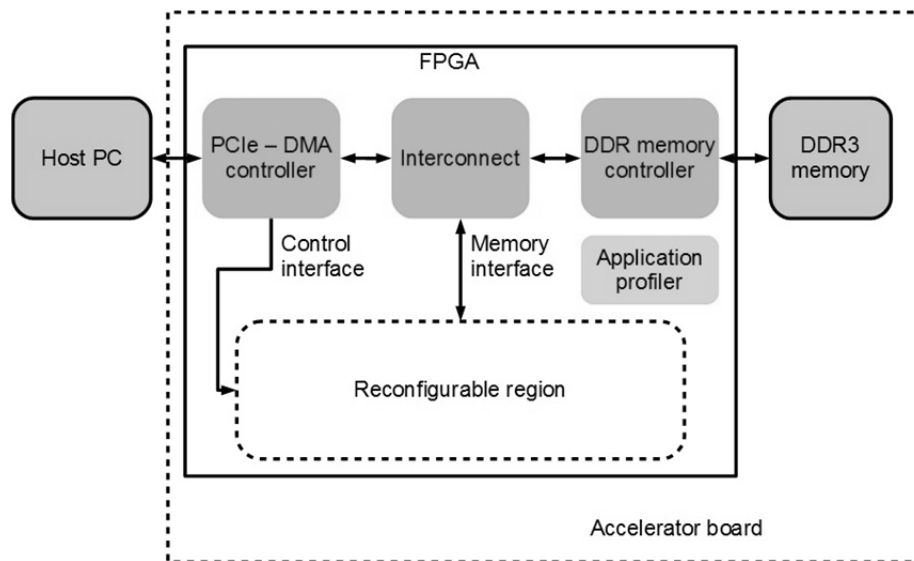


Рис. 1. Структура аппаратного проекта для FPGA
Fig. 1. FPGA hardware project structure

Для создания бюджетной платформы предполагается использовать FPGA семейства Artix 7, имеющие достаточное количество логических ресурсов и относительно низкую стоимость. Для определения наименьшего необходимого количества аппаратных ресурсов, необходимых для реализации инфраструктуры OpenCL, проанализированы требования к отдельным его компонентам.

На рис. 1 представлена структура аппаратного проекта для FPGA, которую необходимо реализовать для возможности интеграции платформы в среду SDAccel [4].

Согласно технической документации, предоставляемой фирмой-производителем [4], произведена оценка ожидаемого количества ресурсов, занимаемого каждым из блоков. Для простоты приведено лишь количество логических ячеек (LUT) и триггеров (FF):

PCIe-DMA controller: 17066, FF – 17370;

DDR3 controller: LUT – 14016, FF – 9019;

Application profiler: LUT – 13413, FF – 14667;

Interconnect: LUT – 4320, FF – 1340.

Итоговая оценка: LUT – 48815, FF – 42396.

Следует учесть, что при максимальной плотности расположения элементов логики возникает нехватка локальных соединительных трасс, в связи с чем трассировщик при компиляции использует трассы, лежащие вне области локализации логических ресурсов.

Поэтому требуется ввести коэффициент запаса 1,5–2, чтобы имелась возможность работы системы на более высоких частотах.

Исходя из представленных данных, сделаны следующие выводы: для реализации аппаратной поддержки OpenCL и интеграции платформы в среду SDAccel необходимо использовать FPGA, имеющую:

более 80 тыс. логических ячеек (LUT);

более 80 тыс. триггеров (FF);

модули встроенной RAM памяти;

DSP-модули, реализующие операции типа multiply-accumulate (MAC);

аппаратный контроллер DDR3/4 памяти;

аппаратный контроллер PCIe;

умножители частоты тактовых сигналов;

блоки, обеспечивающие возможность частичной реконфигурации в процессе работы FPGA.

Обзор существующих решений. В табл. 2 приведены перечень и основные характеристики представленных на рынке платформ на базе FPGA фирмы Xilinx, конструктивно реализованных в формате PCIe, имеющих достаточное количество аппаратных ресурсов для реализации интерфейсов и вычислительных ядер OpenCL [4]. Несмотря на то, что использование нескольких FPGA позволяет реализовать полный потенциал шины PCIe [12], данный подход достаточно сложен в реализации, и все платформы, поставляемые фирмой Xilinx, имеют по одной FPGA на плате.

Некоторые из существующих аппаратных ускорителей

Таблица 2

Table 2

Some of existing hardware accelerator boards

Платформа	Семейство/Тип	LUT	DSP	BRAM, МБ	DDR, Гб	Стоимость, долл. США
KCU105	Kintex Ultrascale XCKU040	242 k	1920	21,1	2	2995
AC701	Artix7 XC7A200T	134 k	740	13	1	1295
Alveo U200	Virtex Ultrascale+XCVU9P	1182 k	6840	345,9	16	8995
Вычислитель СТЦ	Kintex Ultrascale XCKU115	2652 k	22080	303,6	16	35000

Из таблицы видно, что существующие на рынке аппаратно-реконфигурируемые вычислители имеют большое количество доступных логических ресурсов и памяти для реализации вычислительных ядер. Это позволяет использовать данные платформы для построения высокопроизводительных гетерогенных вычислительных систем. Но из-за высокой стоимости они не подходят для массового использования в учебных лабораториях, обеспечивающих подготовку соответствующих специалистов.

Следует заметить, что для платформы AC701, в отличие от всех остальных, производитель не реализовал поддержку стандарта OpenCL.

Разработка пакета поддержки платформы. Согласно ранее сформулированным требованиям для реализации устройства была выбрана микросхема типа XC7A200T, имеющая аппаратные контроллеры интерфейсов PCIe и DDR3, а также более 130 к логических ячеек [4].

Для интеграции разработанного аппаратно-реконфигурируемого вычислителя в среду разработки SDAccel компании Xilinx был создан пакет поддержки платформы, состоящий из аппаратной конфигурации для FPGA и программного обеспечения, поддерживающего взаимодействие хоста и вычислителя.

В табл. 3 представлено количество ресурсов микросхемы, занятых под аппаратный проект для FPGA, обеспечивающий взаимодействие хоста и вычислительного ядра, созданного в соответствии со стандартом OpenCL. Из данной таблицы видно, что количество использованных ресурсов имеет незначительные расхождения с теоретическими оценками.

Расход ресурсов FPGA для статической области микросхемы XC7A200T

Таблица 3

Table 3

FPGA resources utilization by static area of XC7A200T IC

Тип ресурса	Использовано	Доступно	Использовано, %
LUT	49847	133800	37
LUTRAM	8488	46200	18
FF	59788	267600	22
BRAM	77	365	21
DSP	0	740	0
IO	52	285	18
GT	1	4	25
BUFG	13	32	40
MMCM	3	10	30
PLL	2	10	20
PCIe	1	1	100

На FPGA выделена область, доступная для реконфигурации: где будет формироваться аппаратная структура, реализующая функциональность решаемой задачи. В связи с тем, что для создания реконфигурируемой области FPGA необходимо использовать прямоугольные зоны, наиболее подходящим вариантом выбрали разбиение кристалла, показанное на рис. 2, где половина ресурсов FPGA выделена для реконфигурации, а вторая — для ресурсов, обеспечивающих взаимодействие вычислителя с хостом и глобальной памятью.

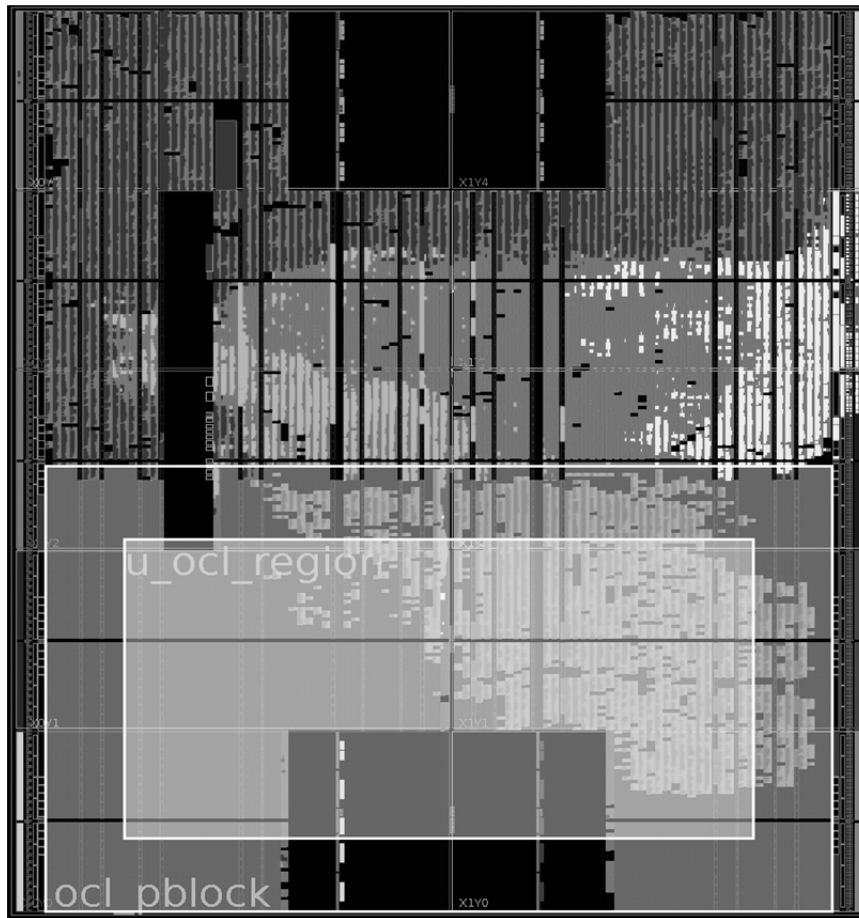


Рис. 2. Результаты трассировки кристалла FPGA

Fig. 2. FPGA routing results

На рисунке подсвеченный белым прямоугольник — область, доступная для реконфигурации и создания аппаратной структуры, реализующей функциональность решаемой задачи (Reconfigurable region). Остальные элементы в верхней половине микросхемы: PCIe-DMA controller; DDR memory controller; application profiler; interconnect.

Для интеграции разработанного аппаратно-реконфигурируемого вычислителя в среду разработки SDAccel необходимо сформировать:

файл конфигурации FPGA;

RTL-схему, описывающую статическую область;

драйвер взаимодействия хоста и вычислителя.

После формирования требуемой структуры папок и добавления платформы в среду разработки стало возможным использовать ее наряду с предустановленными платформами.

Результаты интеграции представлены на рис. 3. На рисунке показано меню выбора аппаратной платформы для проекта и информация о вычислителях. После формирования всех необходимых файлов и подключения их к среде SDAccel разработанный вычислитель появился в списке (под названием ARTIXCL).

Проектирование аппаратной платформы.

Основываясь на реализованном пакете поддержки платформы и общим требованиям к реализации устройств на базе FPGA, составлены требования к набору компонентов, устанавливаемых на плату, структура которой приведена на рис. 4, где изображены:

XC7A200T — FPGA компании Xilinx;

PCIe gen 2.0x4 — разъем PCIe для реализации протокола PCIe2.0x4;

DDR3 512 MB — DDR3 память объемом 512 МБ для реализации глобальной памяти — памяти, к которой имеют доступ как ядро, так и хост-компьютер;

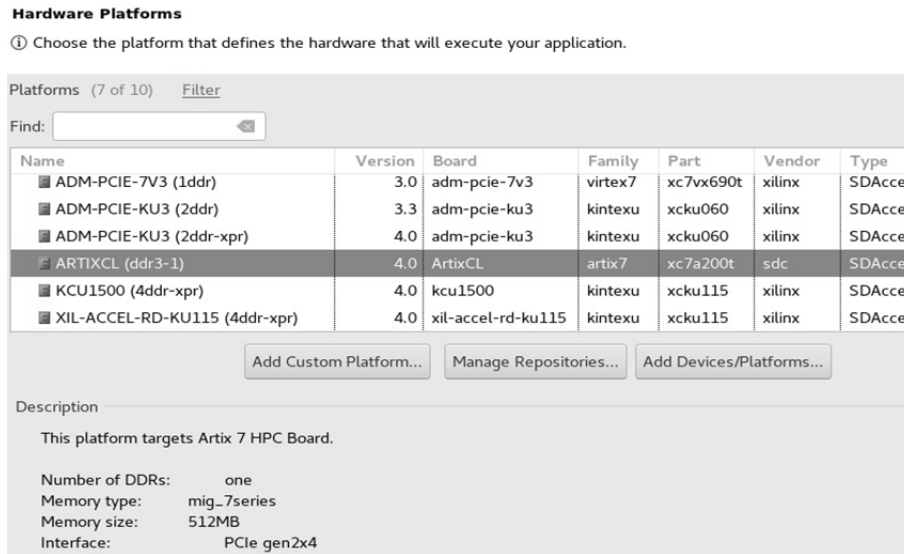


Рис. 3. Список платформ, доступных для выбора в среде SDAccel
Fig. 3. Available platforms list in SDAccel IDE

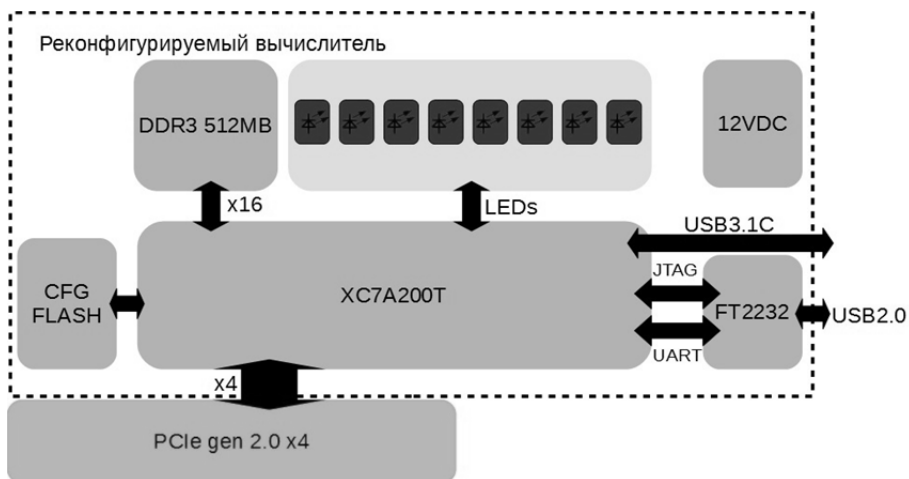


Рис. 4. Структура платы разработанного аппаратно-реконфигурируемого вычислителя
Fig. 4. The designed hardware-reconfigurable accelerator structure

CFG FLASH — конфигурационная память для энергонезависимого хранения конфигурационных данных FPGA;

устройства ввода/вывода для обеспечения отладки: LEDs — 8 светодиодов; FT2232 — преобразователь USB-UART/USB-JTAG; USB3.1C — разъем типа USB3.1 type C для подключения внешних панелей управления; 12VDC — блок питания, на вход которого поступает 12 В.

По приведенному в табл. 4 списку можно дать приблизительную оценку сто-

имости каждого из представленных на схеме компонентов [13]. Затраты на проектирование, тестирование и прототипирование покрыты за счет полученного на разработку гранта от фонда М. Потанина и предоставленных компанией Xilinx лицензий, компонентов и отладочных комплектов.

По данным, приведенным в технической документации на выбранные компоненты, дана оценка максимальной мощности, потребляемой разработанным устройством: 8 Вт.

Таблица 4
Стоимость компонентов разработанного вычислителя

Table 4
Hardware accelerator components' price

Наименование	Стоимость, долл. США
Печатная плата (8 слоев)	15
FPGA (XC7A200T)	193
CFG FLASH (MT25QU256)	3,8
USB-UART (FT2232)	6,99
DDR3L (MT41K256M16TW-107)	16,2
LED x8 (VLMO1500)	4,24
Блок питания (LTM4622x2)	32,7
Итого	271,93

Для каждой платформы из табл. 2 составлена оценка максимальной потребляемой мощности, а также отношения производительности к мощности и производительности к стоимости. Результаты представлены на рис. 5 и 6.

Производительность оценивалась по количеству операций умножения с накоплением (MAC/Multiply-accumulate), выполняемых DSP-модулями. Потребляемая мощность для платы Alveo U200 взята из технической документации, для всех остальных — по результатам экспериментальных измерений.

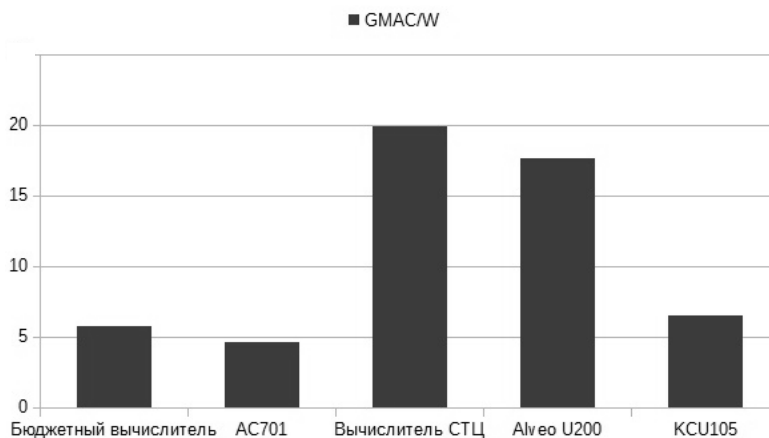


Рис. 5. Отношение производительности к потребляемой мощности для вычислительных платформ
Fig. 5. Performance to power consumption rate for listed hardware platforms

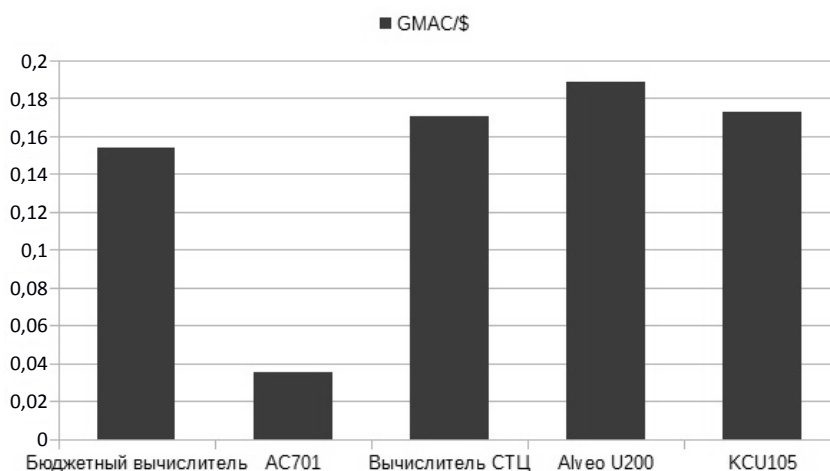


Рис. 6. Отношение производительности к стоимости для вычислительных платформ
Fig. 6. Performance to price rate for listed hardware platforms

По представленным данным можно сделать вывод, что разработанная платформа превосходит платформу AC701, построенную на базе аналогичной FPGA по искомым критериям. Это связано с большей ценой и потребляемой мощностью, которые обусловлены наличием в платформе AC701 дополнительных периферийных устройств, не являющихся необходимыми для реализации OpenCL вычислителя, а также использованием в ней модульной SO-DIMM памяти. Кроме того, из графиков на рис. 5 и 6 видно, что разработанная платформа близка по своим характеристикам к платформе KCU105, построенной на базе FPGA следующего поколения.

Заключение. На основе полученного ранее опыта разработки аппаратно-реконфигурируемых устройств и теоретических расчетов составлены требования к реализации платформы для нового учебно-лабораторного комплекса, обеспечивающего подготовку специалистов для работы с высокопроизводительными аппаратно-реконфигурируемыми системами.

Разработана бюджетная версия аппаратно-реконфигурируемого вычислителя с поддержкой стандарта OpenCL. Вычислитель интегрирован в средства разработки компании Xilinx и обеспечивает поддержку

стандартной процедуры создания приложений для гетерогенных суперкомпьютерных систем.

В ходе сравнительного анализа существующих на рынке реконфигурируемых вычислителей показано, что разработанный бюджетный вычислитель не уступает своим более дорогим аналогам по критериям как отношения производительности на единицу стоимости, так и отношения производительности на единицу мощности.

На базе разработанного вычислителя планируется, в рамках проекта, поддержанного фондом М. Потанина, создание курса для программ магистратуры института ИКНТ (Санкт-Петербургский политехнический университет Петра Великого), обеспечивающего подготовку специалистов, имеющих знания, умения и навыки в технологии разработки и внедрения реконфигурируемых аппаратных ускорителей, адаптируемых в реальном времени вычислительного процесса под решаемую задачу.

Предлагаемый курс базируется на курсах, читаемых в рамках магистерской программы «Проектирование компьютерных систем», обеспечивающих знания в области аппаратных средств, архитектур современных вычислительных систем, системного и прикладного ПО, соответствующих инструментальных средств.

СПИСОК ЛИТЕРАТУРЫ

1. **Каляев И.А., Левин И.И., Семерников Е.А., Шмойлов В.И.** Развитие отечественных многокристалльных реконфигурируемых вычислительных систем: от воздушного к жидкостному охлаждению // Труды СПИИРАН. 2017. Вып. 1. С. 27–39.
2. **Заборовский В.С., Антонов А.П., Киселев И.О.** Специализированные реконфигурируемые вычислители в сетевых суперкомпьютерных системах // Системы высокой доступности. 2018. Т. 14. № 3. С. 57–62.
3. Intel FPGA Devices // URL: <https://www.intel.com/content/www/us/en/fpga/devices.html> (Дата обращения: 04.09.2018).
4. Xilinx Inc. // URL: <https://www.xilinx.com/> (Дата обращения: 04.09.2018).
5. OpenCL specification 1.2. // URL: <https://www.khronos.org/registry/OpenCL/specs/opencl-1.2.pdf> (Дата обращения: 04.09.2018).
6. **Awad M.** FPGA supercomputing platforms: A survey // 19th Internat. Conf. on Field Programmable Logic and Applications. Praga, 2009. Pp. 564–568.
7. **Meeus W., Van Beeck K., Goedemé T., Meel J., Stroobandt D.** An overview of today's high-level synthesis tools // Design Automation for Embedded Systems. 2012. No. 16(3). Pp. 31–51.
8. **Дордопуло А.И., Левин И.И., Каляев И.А., Гудков В.А., Гуленок А.А.** Программирование вычислительных систем гибридного типа на основе метода редукции производительности // Параллельные вычислительные технологии: труды междунар. науч. конф. 2016. С. 131–140.
9. **Shuhao Z., Bingsheng H., Wei Z.** Melia a MapReduce framework on OpenCL based FPGA // IEEE Transactions on Parallel and Distributed Systems. 2015. Pp. 27–39.
10. **Kobayashi R.** OpenCL-ready high speed FPGA network for reconfigurable high perfor-

mance computing // Proc. of the Internat. Conf. on High Performance Computing in Asia-Pacific Region. 2018. Pp. 192–201.

11. Hill K., Craciun S., George A., Lam H. Comparative analysis of OpenCL vs. HDL with image-processing kernels on Stratix-V FPGA // 26th Internat. Conf. on Application-Specific Sys-

tems, Architectures and Processors. IEEE, 2015. Pp. 189–193.

12. Gao S. Characterization of OpenCL on a scalable FPGA architecture // 2014 Internat. Conf. on Reconfigurable Computing and FPGAs. 2014. 8 p.

13. Digi-key Electronics // URL: <https://www.digikey.com/> (Дата обращения: 27.10.2018).

Статья поступила в редакцию 10.10.2018.

REFERENCES

1. Kalyayev I.A., Levin I.I., Semernikov Ye.A., Shmoylov V.I. Evolution of domestic multichip reconfigurable computer systems: from air to liquid cooling. *Trudy SPIIRAN*, 2017, Vol. 1, Pp. 27–39. (rus)

2. Zaborovskiy V.S., Antonov A.P., Kiselev I.O. The reconfigurable computational modules in network-centric supercomputer systems. *Sistemy vysokoy dostupnosti*, 2018, Vol. 14, No. 3, Pp. 57–62 (Accessed: 04.09.2018). (rus)

3. Intel FPGA Devices. Available: <https://www.intel.com/content/www/us/en/fpga/devices.html> (Accessed: 04.09.2018).

4. Xilinx Inc. Available: <https://www.xilinx.com/> (Accessed: 04.09.2018).

5. OpenCL specification 1.2. Available: <https://www.khronos.org/registry/OpenCL/specs/opencl-1.2.pdf> (Accessed: 04.09.2018).

6. Awad M. FPGA supercomputing platforms: A survey. *19th International Conference on Field Programmable Logic and Applications*, Praga, 2009, Pp. 564–568.

7. Mees W., Van Beeck K., Goedemé T., Meel J., Stroobandt D. An overview of today's high-level synthesis tools. *Design Automation for Embedded Systems*, 2012, No. 16(3), Pp. 31–51.

8. Dordopulo A.I., Levin I.I., Kalyayev I.A., Gudkov V.A., Gulenok A.A. Programming of hybrid computer systems based on the performance reduction method. *Parallelnyye vychislitelnyye tekhnologii: trudy mezhdunarodnoy nauchnoy konferentsii*, 2016, Pp. 131–140. (rus)

9. Shuhao Z., Bingsheng H., Wei Z. Melia a MapReduce framework on OpenCL based FPGA. *IEEE Transactions on Parallel and Distributed Systems*, 2015, Pp. 27–39

10. Kobayashi R. OpenCL-ready high speed FPGA network for reconfigurable high performance computing. *Proceedings of the International Conference on High Performance Computing in Asia-Pacific Region*, 2018, Pp. 192–201.

11. Hill K., Craciun S., George A., Lam H. Comparative analysis of OpenCL vs. HDL with image-processing kernels on Stratix-V FPGA. *2015 IEEE 26th International Conference on Application-Specific Systems, Architectures and Processors*, Pp. 189–193.

12. Gao S. Characterization of OpenCL on a scalable FPGA architecture. *2014 International Conference on Reconfigurable Computing and FPGAs*, 2014, 8 p.

13. Digi-key Electronics. Available: <https://www.digikey.com/> (Accessed: 27.10.2018).

Received 10.10.2018.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

АНТОНОВ Александр Петрович

ANTONOV Alexander P.

E-mail: antonov@eda-lab.ftk.spbstu.ru

ФИЛИПPOB Алексей Семенович

FILIPPOV Alexey S.

E-mail: alexey.s.filippov@gmail.com

КИСЕЛЕВ Иван Олегович

KISELEV Ivan O.

E-mail: kio.93@mail.ru

DOI: 10.18721/JCSTCS.11409
УДК 681.51

УПРАВЛЕНИЕ ИМПУЛЬСНЫМ СГЛАЖИВАНИЕМ ФРИКЦИОННЫХ АВТОКОЛЕБАНИЙ ПРИ КОНТАКТНОМ ВЗАИМОДЕЙСТВИИ РОБОТА С ОБРАБАТЫВАЕМОЙ ПОВЕРХНОСТЬЮ

Т.А. Байдина, С.Ф. Бурдаков, О.Б. Шагниеф

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

Рассмотрена задача адаптивного подавления фрикционных автоколебаний системы «робот-инструмент-обрабатываемая поверхность», возникающих при контактном взаимодействии инструмента с поверхностью на ползучих скоростях движения инструмента вдоль поверхности с заданным прижатием к ней в условиях нелинейного трения в области контакта. Подавление фрикционных автоколебаний осуществляется с помощью внешнего импульсного воздействия на систему. При моделировании процессов в области контактного взаимодействия используется модель трения со Штрибек-эффектом, а также учитываются инерционность и податливость элементов системы, что позволило получить процессы, близкие к наблюдаемым на практике. Показано, что в этих условиях на ползучих скоростях движения инструмента относительно поверхности возникают фрикционные автоколебания с кратковременными остановками инструмента. Импульсное воздействие на систему сглаживает процессы, однако после снятия импульсов автоколебания возобновляются. Предложен адаптивный вариант импульсного сглаживания, при котором импульсы подаются на систему по мере возникновения автоколебаний. Небольшое увеличение предписанной скорости движения инструмента после снятия импульсов исключает условия возникновения автоколебаний. Алгоритм реализован с помощью логического переключающего устройства, эффективность работы которого подтверждена моделированием в программном комплексе Matlab.

Ключевые слова: система «робот-инструмент-обрабатываемая поверхность», контактное взаимодействие, нелинейное трение, Штрибек-эффект, ползучая скорость, фрикционные автоколебания, силовое ощущение, адаптивное импульсное сглаживание.

Ссылка при цитировании: Байдина Т.А., Бурдаков С.Ф., Шагниеф О.Б. Управление импульсным сглаживанием фрикционных автоколебаний при контактном взаимодействии робота с обрабатываемой поверхностью // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2018. Т. 11. № 4. С. 119–129. DOI: 10.18721/JCSTCS.11409

CONTROL OF FRICTION SELF-OSCILLATIONS PULSE SMOOTHING IN CONTACT INTERACTION OF A ROBOT WITH THE WORKING SURFACE

T.A. Baidina, S.F. Burdakov, O.B. Shagniev

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

We have considered the problem of adaptive suppression of friction self-oscillations in a robot-tool-working surface system. These oscillations appear under contact interaction of the tool with the surface in case of nonlinear friction in the contact area and creeping speed of the tool motion along the surface. Frictional self-oscillation suppression is carried out using an external pulse load on the system. The friction model with Stribeck's effect in the contact area, inertia and flexibility of the system elements are taken into account in modeling, making it possible to obtain processes close to those observed in practice. It is shown that friction self-oscillations with short-term tool stops arise under these conditions at slow speed of the tool motion along the surface. The pulse load on the system smoothes the processes but self-oscillations appear again after unloading of the pulses. In this paper, we propose an adaptive version of pulsed smoothing. Pulses are fed to the system when auto-oscillations occur. A small increase of the prescribed speed of the tool motion after pulse unloading eliminates the conditions in which self-oscillations occur. The proposed algorithm is implemented using the logical block whose efficiency is confirmed by modeling in Matlab.

Keywords: robot-tool-working surface system, contact interaction, nonlinear friction, Stribeck's effect, creeping speed, friction self-oscillations, force sensing, adaptive pulse smoothing.

Citation: Baidina T.A., Burdakov S.F., Shagniev O.B. Control of friction self-oscillations pulse smoothing in contact interaction of a robot with the working surface. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2018, Vol. 11, No. 4, Pp. 119–129. DOI: 10.18721/JCSTCS.11409

Введение

В настоящее время одной из важнейших задач машиностроения остается создание современных робототехнических комплексов, позволяющих автоматизировать процессы механообработки, сборки и т. д. [1, 2]. Особенность этих процессов связана с контактным взаимодействием инструмента, установленного на роботе, с обрабатываемой поверхностью [3, 4]. Такое взаимодействие, как правило, имеет нелинейный характер, а система «робот-инструмент-обрабатываемая поверхность» обладает инерцией и податливостью [5]. В совокупности эти факторы могут приводить к возникновению нежелательных режимов типа автоколебаний и даже

к неустойчивости [6, 7]. Эффективно управлять робототехническими системами в подобных случаях можно только с помощью силомоментного очувствления робота, когда обратные связи формируются с учетом сил и моментов взаимодействия в области контакта инструмента и обрабатываемой поверхности [8–10 (2 ролика)].

В данной статье рассматривается задача управления фрикционными автоколебаниями, возникающими на ползучих скоростях движения инструмента вдоль поверхности при заданном прижатии к ней в условиях нелинейного трения с эффектом Штрибека в области контакта инструмента с обрабатываемой поверхностью. Ранее в

[11] было установлено, что эффективное подавление фрикционных автоколебаний возможно с помощью импульсного сглаживания нелинейного трения. В настоящей работе показано, как с помощью дополнительного контура адаптации осуществлять импульсное сглаживание в режиме online по мере возникновения нежелательных режимов.

Постановка задачи

Пусть робот обеспечивает движение по трем взаимно перпендикулярным степеням подвижности. На роботе в упругом подвесе закреплен рабочий инструмент. Упругий подвес обеспечивает силовое очувствление робота по трем осям. Обрабатываемая поверхность имеет гладкий профиль. При таких предположениях динамическая взаимосвязь степеней подвижности робота в основном проявляется через силы трения в области контактного взаимодействия инструмента с обрабатываемой поверхностью. Расчетная схема системы «робот-инструмент-обрабатываемая поверхность» представлена на рис. 1. Учитывается движение робота по горизонтальной и вертикальной осям.

На расчетной схеме введены следующие обозначения: m – приведенная масса руки робота; m_s – приведенная масса инструмента; c_{sx} , c_{sy} – жесткости упругого подвеса инструмента; b_x , b_y , b_{sx} – коэффициенты вязкого трения; F_x , F_y – силы, развиваемые приводами робота; F_f – сила трения в области контакта инструмента с поверхностью; R_x , R_y – горизонтальная и нормальная реакции упругого подвеса инструмента; $y_{sf}(x)$ – уравнение обрабатываемой поверхности.

Рассматривается движение робота с заданной скоростью $v_x^d = \dot{x}^d$ вдоль обрабатываемой поверхности $y_{sf}(x)$ при заданном прижатии F_y^d к ней. Считается, что переходные процессы, связанные с выходом робота на контакт с обрабатываемой поверхностью, завершены, а дальнейшее движение робота происходит без нарушения контакта. Известно [12], что на практике в рассматриваемых условиях при очень малой (ползучей) скорости $v_{sx} = \dot{x}_s$ возможно прерывистое движение инструмента типа фрикционных автоколебаний с кратковременными остановками (залипаниями).

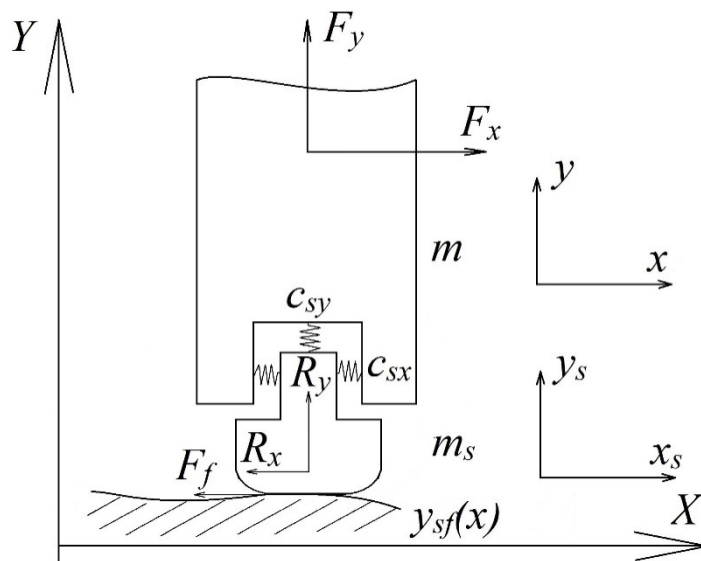


Рис. 1. Расчетная схема системы «робот-инструмент-обрабатываемая поверхность»

Fig. 1. The design scheme of the system «robot-tool-machined surface»

В работе [11] показано, что эффективным способом борьбы с залипаниями инструмента является импульсное сглаживание нелинейного взаимодействия в области контакта. Оно может быть осуществлено с помощью дополнительного внешнего импульсного воздействия на робота. Недостатком этого способа является то, что при снятии импульсного воздействия автоколебания вновь возобновляются. Таким образом, задача состоит в том, чтобы сделать процесс импульсного сглаживания адаптивным, т. е. включать внешнее импульсное воздействие на робота лишь в момент возникновения автоколебаний и снимать его при их отсутствии. Подавать импульсное воздействие постоянно нецелесообразно как по энергозатратам, так и из-за повышения общего уровня вибраций в системе «робот-инструмент-обрабатываемая поверхность».

Импульсное сглаживание фрикционных автоколебаний

Движение системы «робот-инструмент-обрабатываемая поверхность» в режиме контакта с поверхностью описывается следующей системой дифференциальных уравнений (см. рис. 1):

$$\begin{aligned} m\ddot{y} + b_y\dot{y} + c_{sy}(y - y_s) &= F_y - mg, \quad y_s = y_{sf}(x_s), \\ m\ddot{x} + b_x\dot{x} + c_{sx}(x - x_s) &= F_x, \\ m_s\ddot{x}_s + b_{sx}\dot{x}_s + c_{sx}(x_s - x) &= -F_f. \end{aligned} \quad (1)$$

Первое уравнение в системе (1) описывает движение робота вдоль вертикальной оси без нарушения контакта инструмента с обрабатываемой поверхностью. Второе и третье уравнения – движение вдоль горизонтальной оси.

Анализ конструктивных вариантов использования инструмента с силовым чувствлением [9, 10], а также характерного рабочего частотного диапазона для подобных систем позволил при проведении моделирования выбрать следующие значения параметров системы «робот-инструмент-обрабатываемая поверхность»: $m = 2$ кг;

$b_x = b_y = 25$ Н·с/м; $c_{sy} = 4000$ Н/м; $m_s = 0,5$ кг; $b_{sx} = 5$ Н·с/м; $c_{sx} = 3000$ Н/м.

Для принятых значений параметров корни характеристического полинома для уравнения, описывающего движение: вдоль вертикальной оси $p_{1,2} = -6,2 \pm 44,3i$, вдоль горизонтальной оси $p_1 = 0$; $p_2 = -20,7$; $p_{3,4} = -20,9 \pm 82,5i$.

Введем модель трения, учитывающую вязкое трение, сухое (кулоново) трение и трение с учетом Штрибек-эффекта (рис. 2):

$$F_f = \left(\alpha_0 + \alpha_1 \exp \left(- \left(\frac{v_{sx}}{v_{x0}} \right)^2 \right) \right) \text{sign}(v_{sx}) + \alpha_2 v_{sx},$$

где $\alpha_0 + \alpha_1$ – уровень максимальной силы трения покоя; $\alpha_0 = \mu R_y$ – величина силы трения при движении, зависящая от вертикальной реакции поверхности R_y и коэффициента трения μ ; α_2 – коэффициент вязкого трения; v_{x0} – параметр Штрибек-эффекта (крутизна падения силы трения в начале движения).

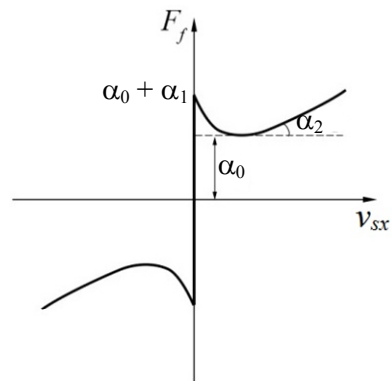


Рис. 2. Характеристика трения

Fig. 2. Friction characteristic

При появлении контакта инструмента с поверхностью на инструмент со стороны поверхности начинает действовать сила реакции $R_y = c_{sy}(y - y_s)$. После достижения требуемого прижатия F_y^d робот начинает перемещаться вдоль оси X , при этом поддерживается необходимое значение силы прижатия F_y^d . Требуемое прижатие обеспечивается при помощи обратной связи по

силе R_y . Позиционно-силовой закон управления по вертикальной координате имеет вид:

$$\begin{aligned} F_y &= \left(k_p + k_i \frac{1}{p} + k_d \frac{N}{1 + Np^{-1}} \right) (\tilde{y}^d - y), \\ \tilde{y}^d &= \left(\theta_p + \theta_i \frac{1}{p} + \theta_d \frac{N}{1 + Np^{-1}} \right) (F_y^d - R_y). \end{aligned} \quad (2)$$

Настройка ПИД-регуляторов (2) осуществлялась с использованием программного комплекса Matlab по показателям качества переходных процессов. При настройке учитывалась ширина диапазона рабочих частот, которая не превышала 30–40 Гц. Предписанное значение силы прижатия составляло $F_y^d = 40$ Н. Для выбранных параметров системы получены следующие настройки: $k_p = 25,3 \cdot 10^3$ Н/м; $k_i = 11,2 \cdot 10^4$ Н/мс; $k_d = 12,7 \cdot 10^2$ Н/мс; $N = 350$; $\theta_p = 0$; $\theta_i = 12,4 \cdot 10^{-4}$ м/сН; $\theta_d = 0$; $N = 100$.

Предписанное движение инструмента вдоль горизонтальной оси $\dot{x}^d$ реализуется с помощью обратной связи по скорости руки робота $\dot{x}$

$$F_x = \left(k_{px} + k_{ix} \frac{1}{p} + k_{dx} \frac{N_x}{1 + N_x p^{-1}} \right) (\dot{x}^d - \dot{x}). \quad (3)$$

Настройка ПИД-регулятора (3) имеет вид $k_{px} = 4,2$ Н/мс; $k_{ix} = 319,7$ Н/мс²; $k_{dx} = 0$ Н/мс; $N_x = 100$.

При моделировании использовалась модель трения, учитывающая эффект Штрибека при $\alpha_0 + \alpha_1 = 5$ Н, $v_{x0} = 0,01$ м/с, $\mu = 0,1$ (см. рис. 2). Моделирование показало [11], что при движении вдоль поверхности с ползучей скоростью порядка $\dot{x}^d = 0,01$ м/с в системе возникают фрикционные автоколебания (рис. 3). Для рассматриваемого примера частота и амплитуда автоколебаний равны соответственно $\omega_A = 76$ рад/с (12 Гц) и $A = 0,012$ м/с.

Моделировалась ситуация, когда в течение первой секунды робот выходил на контакт с поверхностью до достижения силы прижатия $F_y^d = 40$ Н. После чего начиналась отработка предписанного движения вдоль горизонтальной оси со скоростью $v_x^d = 0,01$ м/с.

Характер процессов, приведенных на рис. 3, хорошо согласуется с процессами, наблюдаемыми на практике [12]. Модель трения с учетом Штрибек-эффекта оказалась вполне адекватной для описания кратковременных остановок инструмента при движении на ползучей скорости вдоль обрабатываемой поверхности в условиях прижатия к ней.

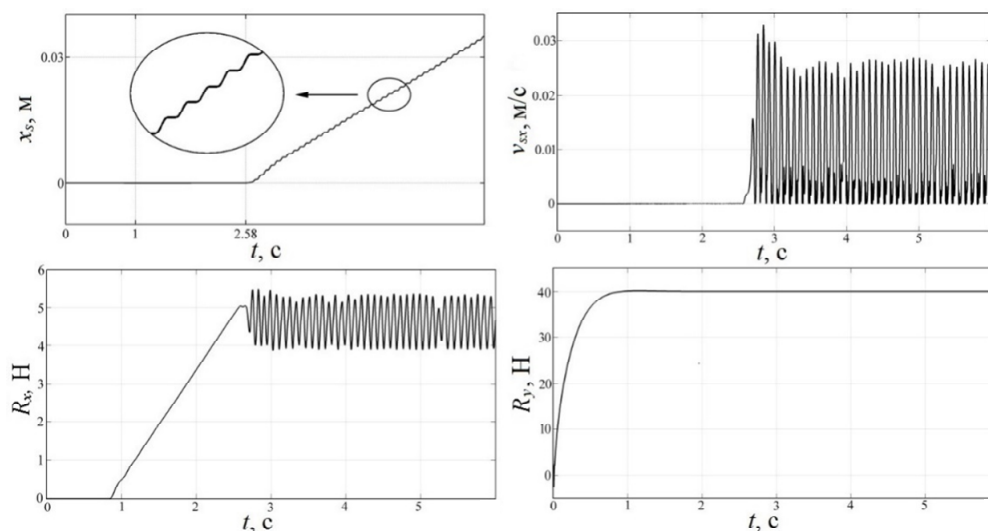


Рис. 3. Процессы возникновения фрикционных автоколебаний на ползучей скорости

Fig. 3. Friction self-oscillation processes at creeping speed

Для подавления фрикционных автоколебаний в контур горизонтального движения дополнительно к $\dot{x}^d$ вводилась составляющая v_{ex}

$$v_{ex}(t) = \begin{cases} B \sin(\omega_B t), & v_{ex} > 0 \\ 0, & v_{ex} \leq 0 \end{cases}, \quad (4)$$

амплитуду B и частоту $\omega_B > \omega_A$ которой можно было варьировать. Эффективное подавление фрикционных автоколебаний получено при $B = 0,025$ м/с, $\omega_B = 300$ рад/с.

На рис. 4 приведены процессы подавления фрикционных автоколебаний. Дополнительное воздействие подавалось в систему в момент начала движения робота вдоль горизонтальной оси. Видно, что в этом случае на

ползучей скорости автоколебания не возникают, и после небольшого переходного процесса скорость инструмента устанавливается на некотором уровне $v_{sx} = 0,018$ м/с, превышающем $\dot{x}^d$. Это превышение обусловлено ненулевым средним значением импульсного воздействия (4). Поэтому при варьировании амплитуды B следует ориентироваться на минимальное эффективное значение.

На рис. 5 приведены графики процессов при снятии внешнего высокочастотного импульсного воздействия.

Подача импульсов осуществлялась в течение 0,5 с. Затем импульсное воздействие снималось. Видно, что после снятия импульсного воздействия автоколебания возобновляются.

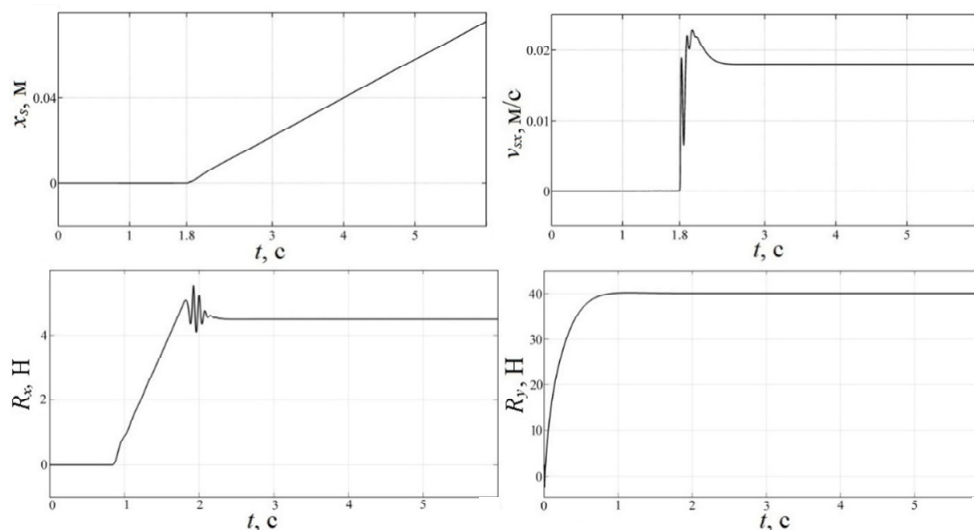


Рис. 4. Процессы подавления фрикционных автоколебаний с помощью импульсного воздействия

Fig. 4. The processes of suppression of friction self-oscillations using pulsed exposure

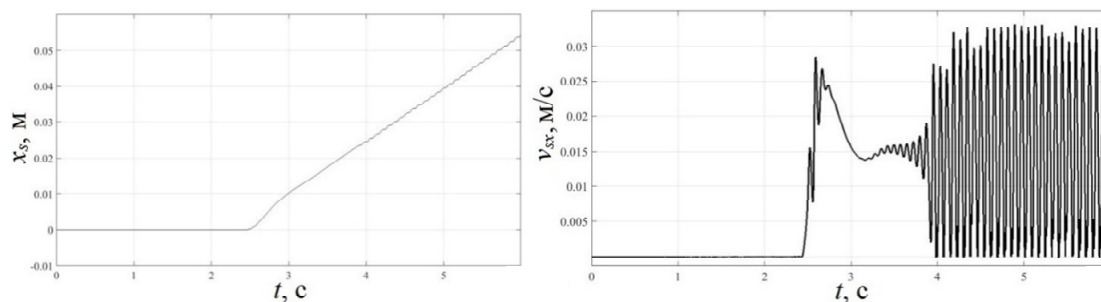


Рис. 5. Процессы повторного возникновения фрикционных автоколебаний при снятии импульсного воздействия

Fig. 5. The processes of re-emergence of friction self-oscillations when removing the pulse effect

Адаптивное управление импульсным сглаживанием

В предыдущем разделе показана принципиальная возможность подавления фрикционных автоколебаний с помощью высокочастотного внешнего импульсного воздействия. Также установлено, что при снятии импульсного воздействия в системе вновь возникают автоколебания (см. рис. 5). В целях снижения общей вибрационной нагрузки на робота необходимо включать импульсное воздействие только тогда, когда возникают автоколебания. А при их отсутствии импульсное воздействие должно сниматься. Иными словами, необходимо сделать систему управления адаптивной. В данной работе адаптация осуществлялась с помощью логического блока (рис. 6).

Для адаптации законов управления необходимо определить сигналы, по которым можно производить включение и выключение дополнительного внешнего импульсного воздействия. Для формирования переключающего устройства проанализируем процессы в системе «робот-инструмент-обрабатываемая поверхность» непосредственно перед возникновением автоколебаний и перед установлением равномерного движения при импульсном сглаживании.

При появлении в системе автоколебаний переходные процессы по горизонтальной силе R_x и горизонтальной скорости v_{sx}

будут иметь ярко выраженный колебательный характер с частотой, равной частоте автоколебаний (см. рис. 3). В этой ситуации появление автоколебаний в системе можно зафиксировать с помощью фильтра высоких частот (ФВЧ), пропуская через него сигнал R_x с соответствующего силового сенсора. При таком преобразовании первый импульс сигнала R_x^* на выходе фильтра будет свидетельствовать о том, что в системе возникают автоколебания, и в момент времени появления данного импульса необходимо включать высокочастотное импульсное воздействие. На рис. 7 приведены графики соответствующих процессов R_x и R_x^* после пропускания через ФВЧ.

При моделировании оказалось, что включение внешнего импульсного воздействия должно произойти в момент времени $t_{on} = 2,3$ с. Результатом его действия на систему должно стать равномерное движение инструмента вдоль горизонтальной оси без кратковременных остановок. Поэтому в качестве сигнала выключения может использоваться информация с датчика скорости по горизонтальной координате $v_x = \dot{x}$. При достижении равномерного движения по горизонтальной координате импульсное воздействие следует выключать. На рис. 8 приведены графики процессов v_x и v_x^* после пропускания через ФВЧ соответственно.

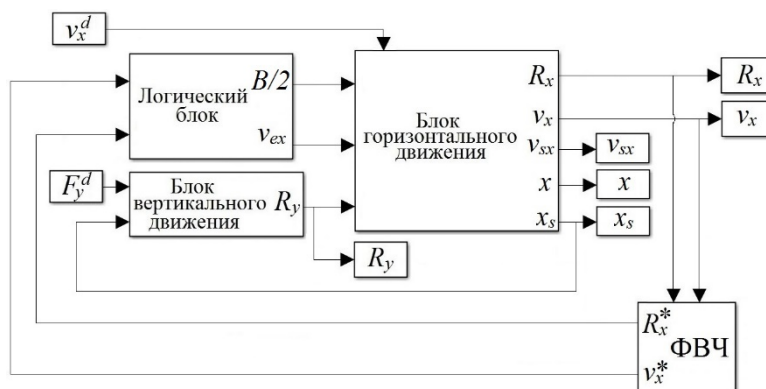


Рис. 6. Структурная схема системы управления

Fig. 6. Control system block diagram

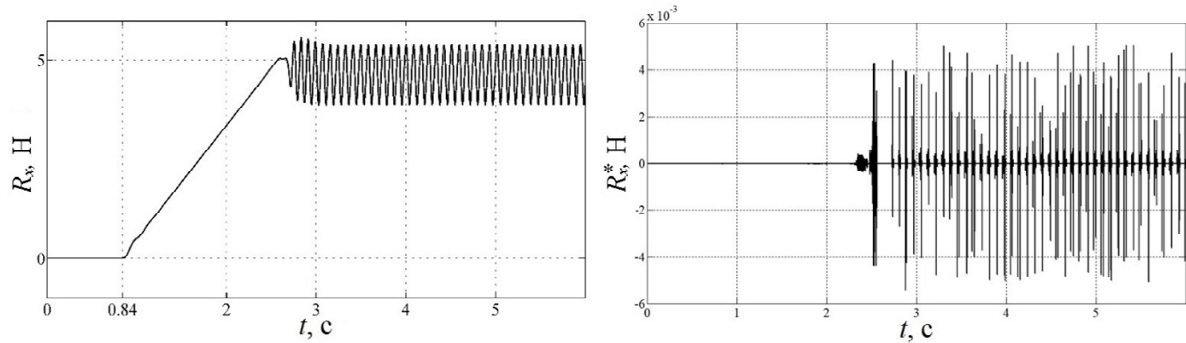


Рис. 7. Процессы в системе R_x до и R_x^* после пропускания через ФВЧ

Fig. 7. Processes in the system R_x and R_x^* after passing through the High-pass Filters

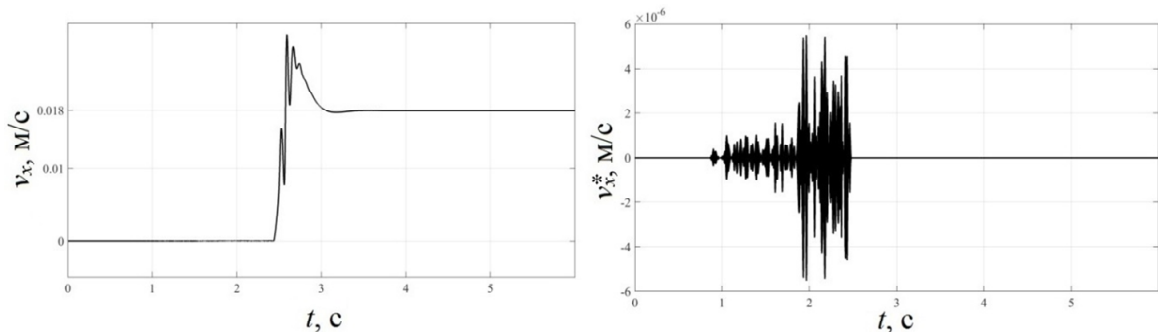


Рис. 8. Процессы в системе v_x до и v_x^* после пропускания через ФВЧ

Fig. 8. Processes in the system v_x and v_x^* after passing through the High-pass Filters

Сигналом к выключению внешнего импульсного воздействия будет служить последний импульс в процессе v_x^* . Момент времени выключения берется с некоторым запасом Δt . При моделировании момент времени выключения внешнего импульсного воздействия составлял $t_{off} = 2,6 + 0,2$ с. Таким образом, подача импульсов внешнего воздействия осуществлялась только в течение 0,5 с.

Для предотвращения повторного возникновения автоколебаний в момент выключения импульсного воздействия следует увеличить задание по скорости v_x^d на величину $B/2$. Небольшое увеличение предписанной скорости v_x^d исключает условия возникновения фрикционных автоколебаний, т. к. при увеличении предписанной скорости система как бы «проскакивает» падающий участок на характеристике трения (см. рис. 2).

В итоге получаем следующую логическую схему. Включение внешнего импульсного воздействия происходит по первому импульсу сигнала R_x^* , пропущенного через

ФВЧ, выключение по сигналу v_x^* — при достижении равномерного движения вдоль горизонтальной оси. В момент снятия импульсного воздействия происходит небольшое увеличение задания на горизонтальную скорость до значения $\tilde{v}_x^d = v_x^d + B/2$ (рис. 9), при котором пропадают условия для возникновения автоколебаний. При таких переключениях автоколебания эффективно подавляются, а время импульсного воздействия на систему составляет менее секунды. Такой метод позволяет оперативно сгладить неблагоприятную динамику системы при условии, что увеличение задания на горизонтальную скорость не приводит к нарушению технологии. В противном случае необходимо изменение технологических требований.

Моделирование полного цикла движения показало эффективность предложенного подхода. Переходные процессы в системе «робот-инструмент-обрабатываемая поверхность» с дополнительным логическим блоком приведены на рис. 10.

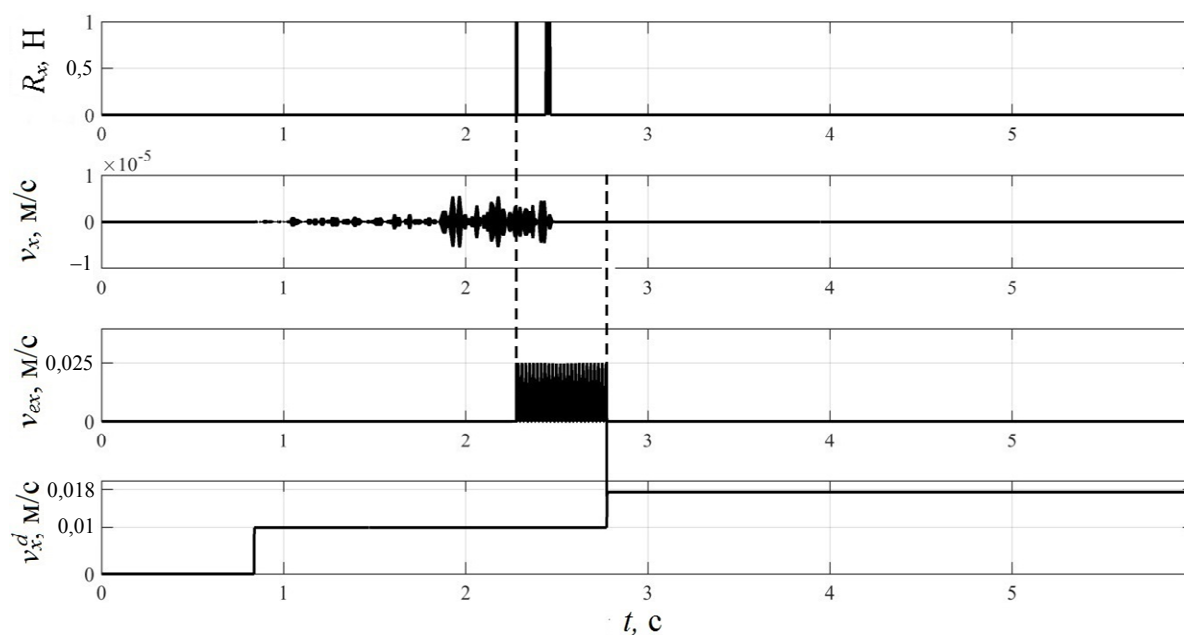


Рис. 9. Сигналы включения/выключения импульсного воздействия

Fig. 9. Pulse on/off signals

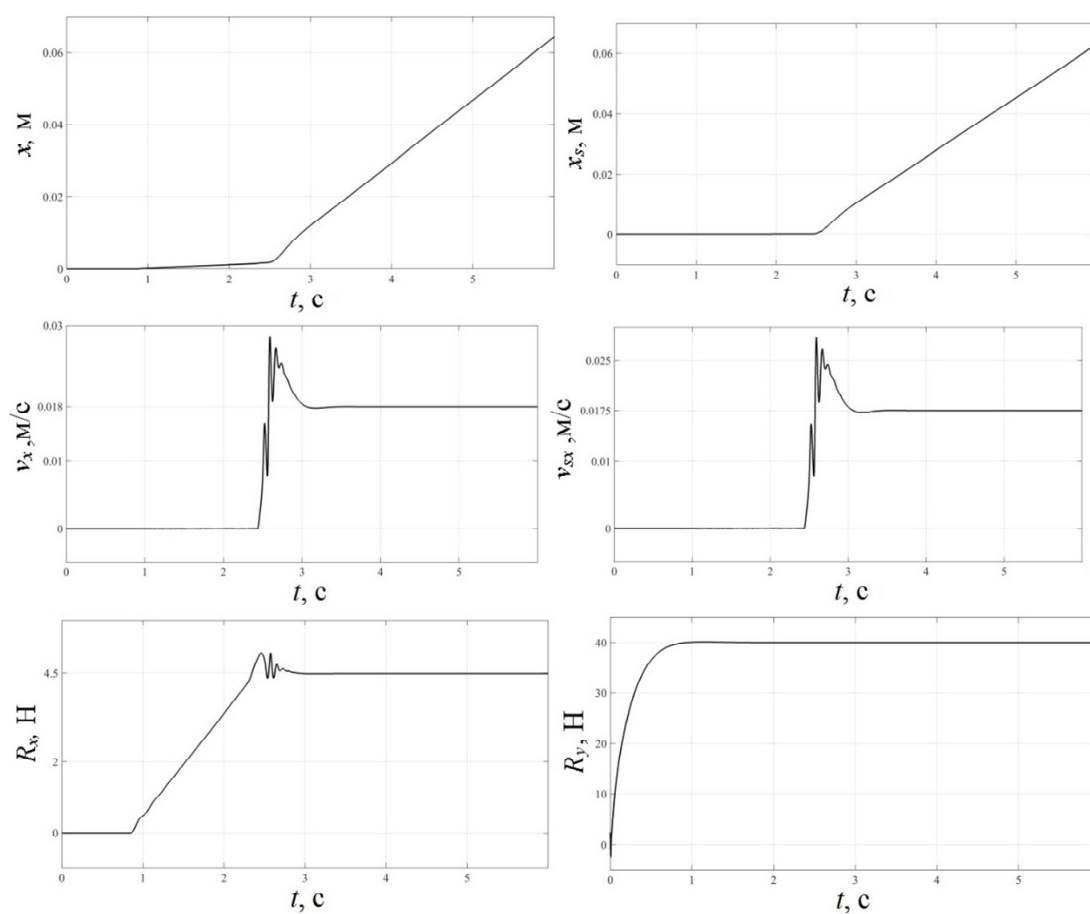


Рис. 10. Процессы в системе в условиях адаптивного импульсного сглаживания фрикционных автоколебаний

Fig. 10. Processes in the system under conditions of adaptive impulse smoothing of friction self-oscillations

Заключение

Задачи с контактным взаимодействием роботов с другими объектами относятся к актуальным и наиболее сложным в робототехнике, т. к. алгоритмы управления приходится создавать с учетом сил и моментов, возникающих в области контакта. Чтобы иметь соответствующую информацию, такой робот должен обладать силомоментным очувствлением. Наличие силомоментного очувствления позволяет вводить в систему управления элементы искусственного интеллекта, такие как обучение, нечеткая логика, нейросети и т. д. Искусственный интеллект в задачах с контактным взаимодействием необходим в силу сложности физико-механических процессов, протекающих в области контакта.

Исследования многих авторов [5, 13, 14] показывают, что в большинстве случаев адекватные модели взаимодействия должны иметь нестационарный, нелинейный, динамический характер. Рассмотренная в данной статье модель трения с

падающей характеристикой при малых относительных скоростях при движении робота вдоль обрабатываемой поверхности в условиях инерционности и податливости элементов робота раскрывает только одну из возможных причин появления автоколебаний. Большой класс задач по моделированию физико-механических процессов в области контактного взаимодействия охватывает проблемы, возникающие при механообработке, такие как недопустимо высокий уровень вибраций, первичные и вторичные (регенеративные) автоколебания при точении и фрезеровании, неустойчивость процессов.

Таким образом, главные направления дальнейших исследований в данной области должны быть связаны с физическими экспериментами и совершенствованием моделей физико-механических процессов при контактных взаимодействиях, а также с повышением уровня интеллектуализации алгоритмов управления роботами, ориентированных на расширение класса решаемых задач.

СПИСОК ЛИТЕРАТУРЫ

1. Егоров И.Н. Позиционно-силовое управление робототехническими и мехатронными устройствами. Владимир: Изд-во Владимир. гос. ун-та, 2010. 192 с.
2. Гориневский Д.М., Формальский А.М., Шнейдер А.Ю. Управление манипуляционными системами на основе информации об усилиях. М.: Физматлит, 1994. 368 с.
3. Altintas Y. Metal cutting mechanics, machine tool vibrations, and CNC design. 2nd ed. Cambridge University press, 2012. 382 p.
4. Попов В.Л. Механика контактного взаимодействия и физика трения. От нанотрибологии до динамики землетрясений. М.: Физматлит, 2013. 352 с.
5. Бурдаков С.Ф., Шагниев О.Б. Модели механики в задаче управления силовым взаимодействием робота с поверхностью неопределенного профиля // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2015. № 4. С. 68–79. DOI: 10.5862/JCSTCS.224.7
6. Гайворонский Е.Г., Кеглин Б.Г. Особенности фрикционных автоколебаний в континуальной системе // Трение и смазка в машинах и механизмах. 2010. № 11. С. 3–9.
7. Крагельский И.В., Гитис Н.В. Фрикционные автоколебания. М.: Наука, 1987. 181 с.
8. Рассказчиков Н.Г. Управление промышленными роботами для абразивной зачистки с использованием силомоментной адаптации // Современные проблемы науки и образования. 2014. № 6 // URL: <http://science-education.ru/ru/article/view?id=16772>
9. URL: <https://www.youtube.com/watch?v=qIFGmlO9Uek>
10. URL: https://www.youtube.com/watch?v=1JdH_ytpzpk
11. Байдина Т.А., Бурдаков С.Ф. Управление автоколебаниями при контактном взаимодействии робота с обрабатываемой поверхностью в условиях нелинейного трения // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2018. Т. 11. № 1. С. 65–74. DOI: 10.18721/JCSTCS.11106
12. Каширин А.И. Исследование вибраций при резании. М.: Изд-во АН СССР, 1944. 262 с.

13. Tobias S.A., Fishwick W. The chatter of lathe tools under orthogonal cutting conditions // Transactions of ASME. 1958. Vol. 80. Iss. 1. Pp. 1079–1088.

14. Tlustý J., Poláček M. The stability of the machine tools against self-excited vibrations in machining // International Research in Production Engineering. 1963. Vol. 1. Iss. 1. Pp. 465–474.

Статья поступила в редакцию 20.11.2018.

REFERENCES

1. Yegorov I.N. *Pozitsionno-silovoye upravleniye robototekhnicheskimi i mekhatronnymi ustroystvami* [Positional-power control of robotic and mechatronic devices]. Vladimir: State University Publ., 2010, 192 p. (rus)

2. Gorinevskiy D.M., Formalskiy A.M., Shneyder A.Yu. *Upravleniye manipulyatsionnymi sistemami na osnove informatsii ob usiliyakh* [Manage handling systems based on effort information]. Moscow: Fizmatlit Publ., 1994, 368 p. (rus)

3. Altintas Y. *Metal cutting mechanics, machine tool vibrations, and CNC design*. 2nd ed. Cambridge University press, 2012, 382 p.

4. Popov V.L. *Contact mechanics and friction physics. From nanotribology to earthquake dynamics*. Moscow: Fizmatlit Publ., 2013, 352 p. (rus)

5. Burdakov S.F., Shagneyev O.B. Mechanics models in the control problem of the force interaction between a robot and a free-formed surface. *St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunication and Control Systems*, 2015, No. 4, Pp. 68–79. DOI: 10.5862/JCSTCS.224.7 (rus)

6. Gayvoronskiy Ye.G., Keglin B.G. The features of frictional self-oscillations in a continuous system. *Friction & Lubrication in Machines and Mechanisms*, 2010, No. 11, Pp. 3–9. (rus)

7. Kragelskiy I.V., Gitis N.V. *Friktsionnyye avtokolebaniya* [Friction auto-oscillations]. Moscow: Nauka Publ., 1987, 181 p. (rus)

8. Rasskazchikov N.G. Upravleniye promyshlennymi robotami dlya abrazivnoy zachistki s ispolzovaniyem silomomentnoy adaptatsii [Control of industrial robots for abrasive cleaning using force-moment adaptation]. *Sovremennyye problem nauki i obrazovaniya* [Modern problems of science and education], 2014, No. 6. Available: <http://science-education.ru/ru/article/view?id=16772> (rus)

9. Available: <https://www.youtube.com/watch?v=qIFGmlO9Uek>

10. Available: https://www.youtube.com/watch?v=IJdH_ytpzpk

11. Baydina T.A., Burdakov S.F. Control of self-oscillations in contact interaction of a robot with a treated surface under nonlinear friction. *St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunication and Control Systems*, 2018, Vol. 11, No. 1, Pp. 65–74. DOI: 10.18721/JCSTCS.11106

12. Kashirin A.I. *Issledovaniye vibratsiy pri rezanii* [Investigation of vibrations during cutting]. Moscow: AN USSR Publ., 1944, 262 p. (rus)

13. Tobias S.A., Fishwick W. The chatter of lathe tools under orthogonal cutting conditions. *Transactions of ASME*, 1958, Vol. 80, Iss. 1, Pp. 1079–1088.

14. Tlustý J., Poláček M. The stability of the machine tools against self-excited vibrations in machining. *International Research in Production Engineering*, 1963, Vol. 1, Iss. 1, Pp. 465–474.

Received 20.11.2018.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

БАЙДИНА Татьяна Александровна

BAIDINA Tatyana A.

E-mail: baydinatanya2401@gmail.com

БУРДАКОВ Сергей Федорович

BURDAKOV Sergey F.

E-mail: burdakov.s@yandex.ru

ШАГНИЕВ Олег Булатович

SHAGNIEV Oleg B.

E-mail: shagnoleg@yandex.ru

DOI: 10.18721/JCSTCS.11410

УДК 004.8.032.26,681.513.8

ИССЛЕДОВАНИЕ И АНАЛИЗ ДИНАМИКИ ИМПУЛЬСНЫХ РЕКУРРЕНТНЫХ НЕЙРОННЫХ СЕТЕЙ В КОНТЕКСТЕ ЗАДАЧИ РАСПОЗНАВАНИЯ ОБРАЗОВ

К.В. Никитин

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

Проведен анализ существующих моделей рекуррентных нейронных сетей (РНС) и подходов к их обучению. В рамках новой парадигмы — резервуарных вычислений рассмотрена модель на основе импульсной РНС в качестве резервуара. Приведены описание и анализ этой модели, построена иерархия и проведена классификация ее основных параметров. Выделен класс решаемых задач, введены показатели качества резервуара. Выполнено экспериментальное исследование влияния параметров на динамику РНС. Результаты этого исследования использованы в разрабатываемой методике синтеза импульсных РНС. Приведен пример успешного применения данной методики при решении задачи классификации динамических образов.

Ключевые слова: машина неустойчивых состояний, резервуарные вычисления, распознавание динамических образов, импульсная нейронная сеть, нейрон интегрирования и возбуждения, синаптическая пластичность.

Ссылка при цитировании: Никитин К.В. Исследование и анализ динамики импульсных рекуррентных нейронных сетей в контексте задачи распознавания образов // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2018. Т. 11. № 4. С. 130–150. DOI: 10.18721/JCSTCS.11410

ANALYSIS OF SPIKING RECURRENT NEURAL NETWORKS DYNAMICS IN CONTEXT OF PATTERN RECOGNITION PROBLEM

K.V. Nikitin

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

The article presents the analysis of current models and learning algorithms of recurrent neural networks (RNN). The model of spiking RNN is considered within the new paradigm of reservoir computing (RC). This model was first introduced in 2006 by Maass and is called the liquid state machine (LSM). The main idea of RC is to construct a random recurrent topology and train only a single linear readout layer. A spiking neural network consists of biologically realistic models of spiking neurons and chemical synapses. Analysis of mathematical model of RNN continues with building a hierarchy of its main parameters



and their classification by different groups. The main problem solved with the LSM is dynamic pattern recognition. Several measures of reservoir quality are introduced for solving this problem. After that, an experimental study has been carried out to assess the influence of all parameters of model on the dynamics, behavior and properties of the RNN. Results of this study are used to build the procedure of synthesis of pulsed RNN for the problem of pattern recognition. The final part of the article demonstrates using the proposed procedure for solving a simple problem of dynamic pattern classification. It is shown that it can simplify synthesis and help to improve the quality of pattern recognition.

Keywords: liquid state machine, reservoir computing, dynamic pattern recognition, spiking neural network, integrate and fire neuron, spike time dependent plasticity.

Citation: Nikitin K.V. Analysis of spiking recurrent neural networks dynamics in context of pattern recognition problem. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2018, Vol. 11, No. 4, Pp. 130–150. DOI: 10.18721/JCSTCS.11410

Введение. Анализ существующих подходов к обучению РНС

Задача распознавания динамических или изменяющихся во времени образов в настоящий момент весьма актуальна и имеет большое количество различных приложений [1, 2]. Одним из наиболее перспективных средств для ее решения являются рекуррентные нейронные сети (РНС), характерная особенность которых — наличие обратных связей (ОС) [3, 4]. За счет ОС входная информация может бесконечно распространяться в динамике нейронных сетей (НС), что при соответствующем подборе параметров в результате процедуры обучения гарантирует универсальные вычислительные возможности РНС в рамках решения рассматриваемого класса задач [5, 6].

Исследования по РНС начались в 70-х — 80-х гг. XX века [7, 8], и на данный момент предложено большое количество вариантов как самих РНС [9, 10], так и подходов к их обучению [5, 11–13]. В наиболее упрощенном варианте все модели РНС можно разделить на два типа: с т. н. управляемой динамикой или аттракторные [9] и РНС общего вида [5].

Суть аттракторных НС [9] сводится к тому, что они функционируют в определенных динамических режимах и спустя некоторое время сходятся к состоянию равновесия — аттрактору. Простейший вариант аттрактора — точка в многомерном про-

странстве, которой сопоставляется некоторый образ, записываемый в НС [8]. При подаче на вход НС нового образа она соотносит его с одним из аттракторов и таким образом решает задачу ассоциативной памяти. Аттракторы могут иметь и более сложную форму: колебательную (предельный цикл, тор) [14] или хаотическую (странный аттрактор). РНС с такими аттракторами называют, соответственно, *осцилляторными* [15] и *хаотическими* [16]. Их характерной особенностью является эффект синхронизации отдельных групп нейронов, используемый часто при решении задач кластеризации [17] и сегментации [18].

РНС общего вида могут иметь произвольную динамику, поскольку основной принцип их построения заключается в том, чтобы сеть выдавала правильную выходную последовательность сигналов в ответ на входную. В этом смысле данные сети похожи на обычные нейронные сети прямого распространения (НСПР), поскольку на основе обучающей выборки они модифицируют свои весовые коэффициенты так, чтобы конечная функция стоимости была минимальной. В настоящее время разработано большое число алгоритмов обучения РНС [12]. Отдельно от всех остальных находится алгоритм обучения машины Больцмана [19]. Основную группу составляют градиентные алгоритмы [12, 13]:

- обычное обратное распространение ошибки (НС Элмана, НС Джордана, НС с контекстом);

- обратное распространение во времени (Back propagation through time – BPTT);
- рекуррентное обучение в реальном времени (Real-time recurrent learning – RTRL);
- быстрое прямое распространение (Fast forward propagation – FFP);
- использование функций Грина;
- рекуррентное обучение Атия–Парлоса.

Помимо градиентных также получили распространение более усовершенствованные алгоритмы обучения [12]:

- усеченный алгоритм распространения во времени;
- сфокусированное обратное распространение;
- использование фильтров Калмана;
- модель длительной краткосрочной памяти (Long short term memory – LSTM) [10, 20];
- аппроксимация алгоритма Левенберга–Марквардта.

В ряде случаев помогает простое угадывание весов. Для определенных задач также разработаны специальные алгоритмы, такие, например, как адаптивная фрагментация последовательностей [12].

Если охарактеризовать представленные выше алгоритмы, то следует сказать об их высокой сложности, больших вычислительных и временных затратах, а также о проблемах, связанных со сходимостью обучения, и в частности, с обучением долгосрочным зависимостям [11]. В связи с этим параллельно разрабатывались и исследовались альтернативные подходы, которые со временем объединились в новое направление – резервуарные вычисления (РВ) [21–24].

Суть подхода на основе РВ заключается в том, что специальным образом синтезируется специальная РНС – резервуар, которая преобразует исходное пространство временных функций в пространство большей размерности таких же функций, т. е. РНС действует как некоторый нелинейный оператор [25]. При выполнении определенных требований РНС способна разде-

лять различные входные последовательности, что позволяет по состоянию РНС через некоторое время после подачи входного образа определить, какой образ был подан на вход. Предложено большое количество различных моделей РВ [22, 26–29]:

- сеть эхо-состояний (Echo state network – ESN);
- машина неустойчивых состояний (Liquid state machine – LSM) – РНС построена на импульсных нейронах;
- эволюино (Evolino) – LSTM НС и эволюционное обучение;
- обратное распространение и декорреляция (Backpropagation-Decorrelation) – подход к обучению, обобщающий многие алгоритмы РВ и основанный на APRL;
- временные рекуррентные сети (Temporal recurrent networks) – моделируются кортикальные сети мозга с уклоном в вычислительную нейронауку, нейробиологию;

• другие типы резервуаров с более сложными моделями элементов, возможно с аппаратной реализацией (электрические цепи, оптические, квантовые, физические компьютеры), из «живых» клеток (сеть генной регуляции из бактерий).

Модель МНС является примером эффективного синтеза биологически ориентированных моделей и методов машинного обучения для решения задач распознавания динамических образов [30–34]. Теоретически подтверждаются универсальные вычислительные возможности МНС при аппроксимации нелинейных систем с затухающей памятью [35]. При этом остается актуальной задача синтеза резервуара под конкретную задачу и тип данных [23, 36–40].

Далее приведено описание предварительных этапов перед построением методики синтеза резервуара импульсной РНС в составе МНС. Проводится анализ всех составляющих модели, строится иерархия параметров, описываются решаемые задачи и используемые критерии, после чего выполняются исследование и анализ влияния параметров на функционирование.

Описание модели машины неустойчивых состояний

МНС структурно состоит из двух составляющих [41]: резервуара в форме импульсной РНС и блока считывателей. РНС используется как резервуар с богатой динамикой и возможностями по различению входных сигналов. При этом РНС генерируется случайным образом в начале работы без дополнительного этапа обучения. При подаче на вход РНС временной последовательности она через какое-то время переходит в новое состояние. Выход РНС соединен с т. н. считывателями — устройствами, выделяющими из сигнала РНС полезную информацию, решая требуемую задачу. Общая схема МНС показана на рис. 1.

Импульсная РНС имеет случайную трехмерную структуру. Вероятность связи между двумя нейронами задается следующим законом [3]:

$$P(a,b) = \min[1, C(a,b) \cdot \exp(- (D(a,b)/\lambda)^2)],$$

где символами a и b обозначаются два нейрона; $D(a, b)$ — евклидово расстояние между этими нейронами; λ — параметр, управляющий средним радиусом связи между нейронами; $C(a, b)$ характеризует плотность связей и зависит от того, какой тип имеют нейроны: возбуждающий или подавляющий.

Рассмотрим далее более подробно каждую из составляющих МНС.

Импульсные НС [42, 43] состоят из импульсных нейронов, связанных химическими синапсами. На вход таких НС может поступать информация как в импульсной форме через химические синапсы, так и в аналоговой — через электрические синапсы. На вход каждого нейрона может также подаваться шум для того, чтобы НС находилась в слегка возбужденном состоянии.

Как правило, используются модели *импульсных нейронов*, описываемые дифференциальными уравнениями (ДУ) 1-2 порядка [44]. Наиболее простая и понятная модель интегрирования и возбуждения с утечкой (Leaky integrate and fire — LIF) описывается с помощью ДУ и условия сброса. ДУ записывается как [42]

$$C \frac{dv}{dt} = \frac{v}{R} + \frac{v_{rest}}{R} + i_{ext},$$

где v_{rest} — это потенциал покоя нейрона; R — сопротивление резистора; i_{ext} — внешний входной сигнал.

Вторая часть модели отвечает за возбуждение: как только потенциал v достигает порогового значения θ , происходит генерация импульса, потенциал сбрасывается до значения v_{reset} , а сам нейрон переводится в состояние рефракторности на период T_{refrac} , в течение которого он не воспринимает никакие внешние сигналы.

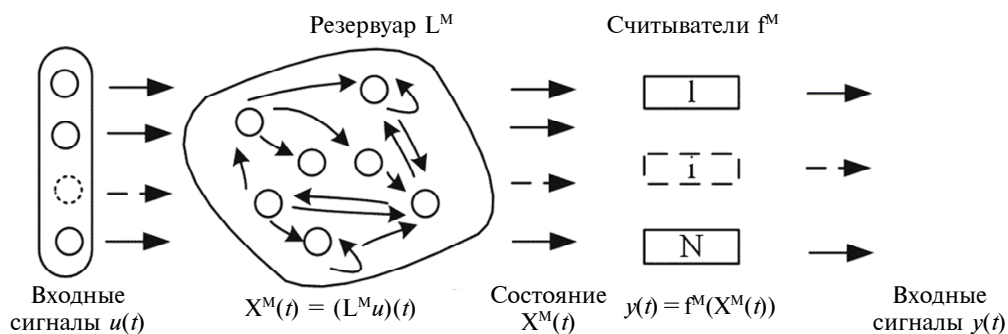


Рис. 1. Схема машины неустойчивых состояний

Fig. 1. Machine diagram of unstable states

Модель нейрона *Ижикевича* [45] является аппроксимацией более сложных моделей типа Ходжкина–Хаксли и др. Она также состоит из двух частей – ДУ и уравнения, описывающего условие возбуждения и сброс нейрона. Оригинальная модель записывается в форме уравнений состояния

$$\begin{cases} \frac{dv}{dt} = 0,04v^2 + 5v + 140 - u + i_{ext}, \\ \frac{du}{dt} = a(bv - u) \end{cases}$$

и условием сброса нейрона при достижении порогового значения

$$\text{если } v \geq 30, \text{ то } v = c, \quad u = u + d,$$

где v – потенциал нейрона, мВ; u – характеризует восстановление мембраны; i_{ext} – входной ток; t – время, мс; a, b, c, d – параметры модели.

Топология и размер импульсной РНС могут быть произвольными. Ограничения определяются только временем моделирования. Импульсные НС из более сложных нейронов, например, Ходжкина–Хаксли, имеют в среднем 10–100 нейронов, из нейронов Ижикевича – 1000 нейронов, из нейронов интегрирования и возбуждения – более 1000 нейронов [46].

В импульсных НС большая часть синапсов являются химическими [46]. Химические синапсы с точки зрения моделирования имеют две особенности:

активизация только в ответ на импульсы (ток, протекающий через нейрон-получатель в момент импульса, моделируется затухающей функцией);

ненулевая задержка при передаче импульса.

При моделировании химических синапсов постсинаптический ток i_{post} , вызванный импульсами, можно смоделировать с помощью убывающих экспоненциальных функций [42]:

$$i_{post}(t) = w_{chem} \sum_i 1(t - t_i - \tau_{delay}) \exp\left(-\frac{t - t_i - \tau_{delay}}{\tau_{syn}}\right),$$

$$\frac{dv}{dt} = f(v, par) + ki_{post}(t),$$

где w_{chem} – сила связи синапса; t_i – время i импульса; τ_{delay} – задержка; $1(t - t_i - \tau_{delay})$ – ступенчатая функция Хевисайда, переходящая из нуля в единицу в момент времени $t_i + \tau_{delay}$; τ_{syn} – постоянная времени, характеризующая скорость убывания экспоненты.

В ряде случаев могут использоваться более сложные модели синапсов. Динамические синапсы [47] учитывают предысторию изменения сигналов на их входах и таким образом способны моделировать эффекты насыщения и восстановления. Синапсы с пластичностью [46] способны адаптировать свою пластичность по правилу Хэбба, тем самым усиливая передачу информации вдоль направлений с наибольшей активностью.

При подаче и съеме информации с импульсной РНС требуется применение специальных методов кодирования/декодирования из обычной непрерывной в импульсную и наоборот [42]. Наиболее распространенные методы [42] основаны на частотном подходе, кодировании по принципу да/нет, популяционном или пространственно-временном кодировании, а также на основе синхронизации [48].

Считыватели – устройства, предназначенные для анализа динамики резервуара и решения конкретных прикладных задач.

Иерархия параметров модели МНС

Перед выполнением экспериментального исследования влияния параметров импульсной РНС на ее свойства необходимо вначале определиться с набором подстраиваемых параметров и диапазоном их изменения. Параметры всей системы можно разделить на три большие группы: параметры входных данных (задачи); параметры резервуара – импульсной РНС; параметры считывателей.

Рассмотрим вкратце каждую из этих групп.

- Входные данные представлены входными образами из обучающих и тестовых примеров. Они характеризуются размерностью, длительностью, формой представления, форматом подачи на вход, алгоритмом нормали-



зации (во временном диапазоне), алгоритмами преобразования данных из одной формы в другую, скоростью и сложностью.

- Импульсная РНС (резервуар) задается структурой, типом элементов и типом связей.

Структура определяется количеством нейронов $n_{nm} = n_x n_y n_z$, средним радиусом связности λ , процентом связей от входов к РНС p_{vh} , плотностью связей $C(a, b)$, силой (весами) внутренних $W_{int}(a, b)$ и внешних $W_{vh}(a)$ связей, соотношением числа подавляющих/возбуждающих нейронов p_{inh} и уровнем шума p_{noise} .

В качестве *элементов* могут использоваться простые модели типа нейрона интегрирования возбуждения (IaF) и Ижикевича, а также детальные модели Ходжкина–Хаксли, Морриса–Лекара, Фитцху–Нагумо. Общими параметрами этих моделей являются сила шума i_{noise} и начальные условия v_0 . Нейроны IaF могут быть с утечкой или без; их основные параметры – это постоянная времени τ , период рефракторности T_{refrac} , потенциалы покоя v_{rest} , сброса v_{reset} , порог θ . Нейрон Ижикевича задается четверкой параметров a, b, c, d . В детальных моделях все параметры, как правило, фиксированы.

Связи между нейронами реализуются посредством электрических и химических синапсов. Их основными параметрами являются сила $W_{int}(a, b)$, $W_{vh}(a)$ и задержка τ_{delay} . У электрических синапсов задержка отсутствует ($\tau_{delay} = 0$). У химических синапсов отдельно задается постоянная времени τ_{syn} затухания синаптического тока. Динамические синапсы со свойством кратковременной памяти задаются тройкой параметров U, D, F . Для синапсов с пластичностью необходимо задавать максимальную по модулю силу синапса w_{max} , минимальное приращение силы синапса Δw_{min} и постоянную времени забывания приращения силы синапса ΔT_{forget} .

При использовании синапсов с пластичностью происходит обучение без учителя, для которого также задается несколько параметров: период обучения T_{learn} , функция обучения $f_w(\Delta t)$ и ее параметры.

В качестве функции чаще всего используются три типа: экспоненциальная (параметры $X_{0LTP}, X_{0LTD}, T_{LTP}, T_{LTD}$), пороговая (параметры $A_{LTP}, A_{LTD}, \Delta T_{max}, \Delta T_{min}$) и гауссоида (параметры $C, \beta, F_{max}, B, \Delta T_{max}, \Delta T_{min}$).

- Считыватели в зависимости от решаемой конечной задачи могут быть представлены по-разному. По выполняемой функции все считыватели могут выполнять задачи предобработки выходного сигнала РНС (сокращение размерности, преобразование информации из импульсной формы в непрерывную, нормализация), анализ выходного сигнала РНС (определение типа динамики, вычисление различных показателей) или решение конечной задачи (классификация, регрессия, кластеризация).

На основании рассмотренной выше группы параметров можно построить обобщенную иерархию параметров системы (рис. 2). На ней параметры сгруппированы по принципу принадлежности к той или иной группе по аналогии с тем, как это было рассмотрено выше. Стрелками показана зависимость параметров и групп параметров друг от друга.

Из рисунка видно, что для синтеза резервуара-импульсной РНС требуется подобрать значения около 30 различных параметров на различных уровнях иерархии. Для этого целесообразно часть наименее значимых или зависимых параметров зафиксировать, а для наиболее значимых параметров задать диапазоны поиска наилучших значений этих параметров.

Анализ иерархии позволил выделить стратегии к поиску параметров. В частности, часть параметров взаимосвязана друг с другом, поэтому их изменение следует выполнять согласованно. Также все параметры были классифицированы с точки зрения их значимости и влияния на общую динамику, в результате чего они были разделены на несколько групп: важные параметры, которые необходимо варьировать в широких диапазонах, менее важные параметры, которые можно варьировать в небольших диапазонах, и фиксированные параметры. Результаты приведены в табл. 1.

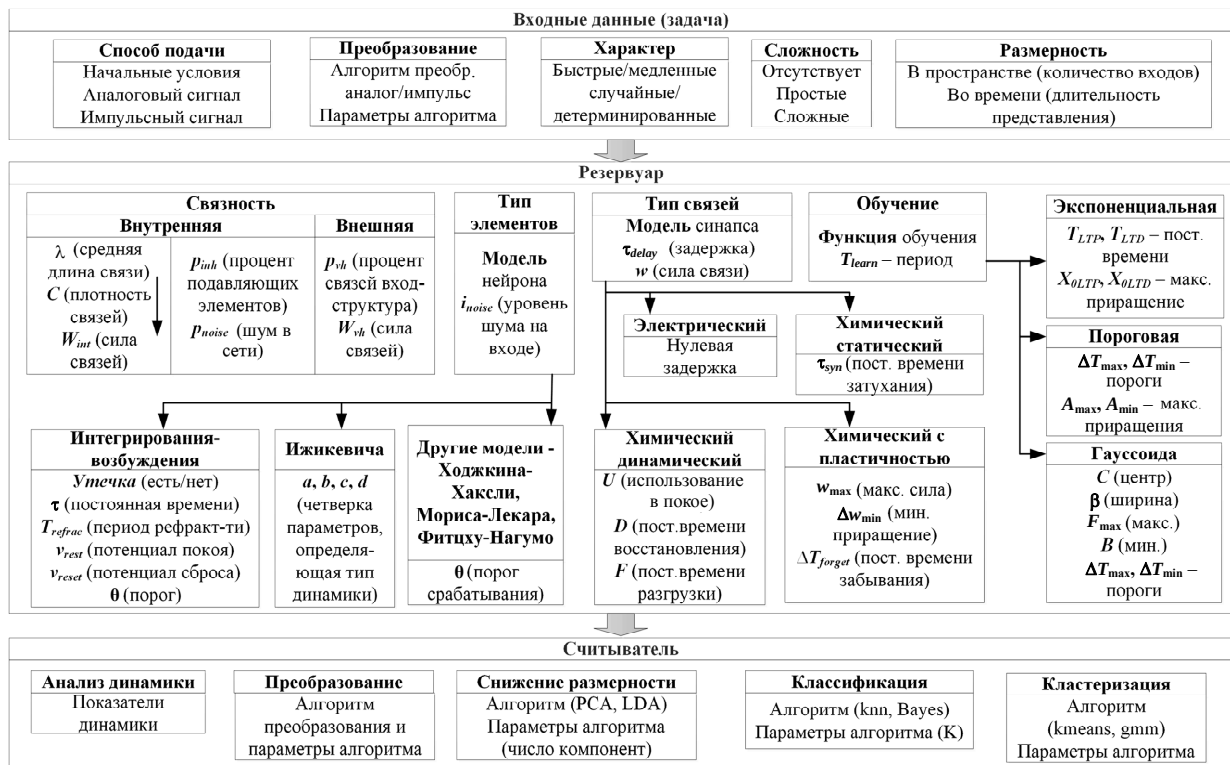


Рис. 2. Иерархия параметров МНС

Fig. 2. MNN Parameter Hierarchy

Таблица 1

Классификация основных параметров по группам

Table 1

Classification of the main parameters by groups

1 группа (варьируемые)	$\lambda \leftarrow C \leftarrow W_{int}, P_{inh}$ (внутренняя связность) P_{vh}, W_{vh} (внешняя связность) P_{noise}, i_{noise} (шум) $t_{readout}$ (считыватель)
2 группа (настраиваемые)	n_{nm} (число нейронов) τ, T_{refrac} (IaF нейрон) a, b, c, d (нейрон Ижикевича, модели RS, FS, CH) τ_{syn} (синапсы), T_{learn} (обучение)
3 группа (фиксируемые)	n_{vh} (число входов) $v_{resets}, v_{rest}, \theta$ (IaF) τ_{delay} (задержка в синапсах) U, D, F (динамические синапсы) $\Delta w_{min}, \Delta w_{max}, \Delta T_{forget}$ (синапсы с пластичностью) $X_{0LTP}, X_{0LTD}, A_{LTP}, A_{LTD}, B, F_{max}, T_{LTP}, T_{LTD}, \Delta T_{max}, \Delta T_{min}, C, \beta$

Таблица 2

Диапазоны значений основных параметров модели

Table 2

The ranges of the basic parameters of the model

Параметр или модель	Диапазоны значений
Размерность	$n_{nm} = n_x n_y n_z \geq 100$
Связность	$\lambda \in [1, 3], c_{ij} \in [0, 100] \%$
Тип нейронов	$p_{inh} \in [0, 100]$
Сила связей, химические синапсы	$W_{int}, W_{vh,chem} \in [0,05, 0,2]$ (IaF) $W_{int}, W_{vh,chem} \in [0,01, 0,2]$ (Ижикевича)
Сила связей, электрические синапсы	$W_{vh,el}, i_{noise} \in [0,01, 0,05]$ (IaF) $W_{vh,el}, i_{noise} \in [0,005, 0,05]$ (Ижикевича)
Шум	$p_{noise} \in [20,50] \%$ плюс значения 0 и 100 %
Задержки	$\tau_{delay} = 1$ с
IaF нейрон	Утечка есть, $v_{rest} = v_{reset} = 0$, $\theta = 15$ мВ, $T_{refrac} = 3$ мс, $\tau = 30$ мс
Нейрон Ижикевича	модель RS: $a = 0,02, b = 0,2, c = -65, d = 8$ модель FS: $a = 0,1, b = 0,2, c = -65, d = 2$
Синапсы	$\tau_{syn} = [3,10]$ мс
Динамические синапсы	$U = 0,5; D \in [10,100]$ мс; $F \in [10,100]$ мс
Обучение	$T_{learn} \in [0,5; 2]$ с
Синапсы с пластичностью	$w_{max} \in [0,1; 0,5], \Delta w_{min} = 0,0001; \Delta T_{forget} \in [1,10]$ с
Функции обучения	$X_{0LTP}, X_{0LTD}, A_{LTP}, A_{LTD} \in [0,0001; 0,001];$ $T_{LTP}, T_{LTD}, \Delta T_{min}, \Delta T_{max} \in [5; 50]$ мс

Далее для каждого из параметров модели РНС были определены диапазоны их изменения или просто значения. Результаты представлены в табл. 2.

Задачи и показатели, используемые при синтезе РНС

Структура и параметры резервуара подбираются для улучшения качества решения задачи распознавания динамических образов, которая является частным случаем задачи распознавания образов. Ее суть сводится к тому, что на основании обучающих примеров система должна так настроить свои внутренние параметры, чтобы ошибка на тестовой выборке была минимальной. При

этом входной и выходной сигналы представлены последовательностями значений, а ошибка вычисляется как разность между желаемым и реальным выходными сигналами.

Предпочтительным является задание входных данных в импульсной форме. В случае задания данных в непрерывной форме необходимо выполнить преобразование этих данных в импульсную форму. Наилучшими возможностями обладает пространственно-временной алгоритм преобразования на основе рецептивных полей.

Сложность задачи является нечеткой мерой, и для ее характеристики могут использоваться разные показатели: различимость образов, компактность (разнесенность), ди-

намика (медленная или быстрая), корреляции [2]. В нашем случае чем сложнее динамический входной образ, тем сложнее и разнообразней должна быть структура РНС и используемые модели нейронов и синапсов.

Для настройки импульсной РНС используются различные качественные критерии [27, 38, 40].

Первый критерий основан на том принципе, что РНС обладает наибольшими вычислительными возможностями, если находится вблизи границы между хаосом и порядком. Поэтому параметры следует выбирать таким образом, чтобы динамика РНС в ответ на тестовые входные образы была более длительной, но все-таки затухающей.

Второй критерий основан на определении реакции РНС на различные входные динамические тестовые образы. Параметры РНС следует подбирать таким образом, чтобы она лучше разносила в своем состоянии различные входные образы. Показателем выбирается метрика различия состояний РНС в ответ на разные входные образы.

Исследование и анализ влияния параметров

Построенная иерархия позволила разделить множество и диапазоны изменения варьируемых параметров и перейти к следующему этапу построения методики — экспериментальному исследованию. Суть экспериментального исследования в конечном счете свелась к перебору различных наборов значений параметров, формированию соответствующих моделей РНС, определению для них показателей качества и дальнейшему обобщению полученных данных. Была предложена и реализована следующая схема:

для выбранных показателей качества сгенерировали необходимые тестовые входные сигналы;

параметры проанализировали на предмет зависимости друг от друга, что позволило часть из них зафиксировать и не рассматривать в дальнейшем;

для оставшихся параметров задали диапазоны возможных значений;

после этого провели эксперименты по всем возможным сочетаниям параметров

резервуара и для них определили показатели качества;

если какой-то параметр слабо влиял на показатель качества, его перемещали в группу фиксированных параметров, и наоборот, если оказывалось, что изменением текущих варьируемых параметров не удавалось добиться существенного изменения динамики, часть фиксированных параметров могла перейти в статус изменяемых;

если для одного из показателей качества были получены все необходимые зависимости, выбирался следующий показатель качества, и процесс повторялся.

Описанная выше схема приведена на рис. 3.

Далее для упрощения будут использоваться следующие обозначения:

$$W_{\text{int}} = \begin{bmatrix} w_{ii} & w_{ie} \\ w_{ei} & w_{ee} \end{bmatrix}, C = \begin{bmatrix} c_{ii} & c_{ie} \\ c_{ei} & c_{ee} \end{bmatrix},$$

где нижние индексы содержат две буквы: *i* (inhibitory) — подавляющий, *e* (excitatory) — возбуждающий. Первая буква обозначает тип элемента-источника, а вторая — тип элемента-получателя. То есть w_{ie} обозначает силу химических синапсов от подавляющих к возбуждающим нейронам.

$W_{vh(\text{chem})}$ — матрица W_{vh} для химических синапсов. $W_{vh(\text{el})}$ — матрица W_{vh} для электрических синапсов: $W_{vh} = \begin{bmatrix} w_i^{vh} & w_e^{vh} \end{bmatrix}$, где нижний индекс обозначает тип элемента-получателя, то есть $w_i^{vh(\text{el})}$ обозначает силу электрических синапсов от входов к подавляющим нейронам.

РНС из IaF нейронов. Вначале проводились исследования с РНС, состоящей из более простых IaF нейронов. Были заданы следующие начальные значения параметров: $p_{inh} = 20 \%$, $p_{ex} = 80 \%$, $\lambda = 3$,

$$C = \begin{bmatrix} 0 & 0,5 \\ 0,5 & 0,5 \end{bmatrix}, p_{vh} = 100 \%, W_{vh} [0,01; 0,03],$$

$$p_{noise} = 100 \%, v_{noise} = R \cdot i_{noise} = 0,0135, v_0 = 0,0155.$$

В результате экспериментов была получена граница перехода динамики из затухающей в незатухающую для двух параметров (w_{ee} , w_{ie}). Она представлена на рис. 4.

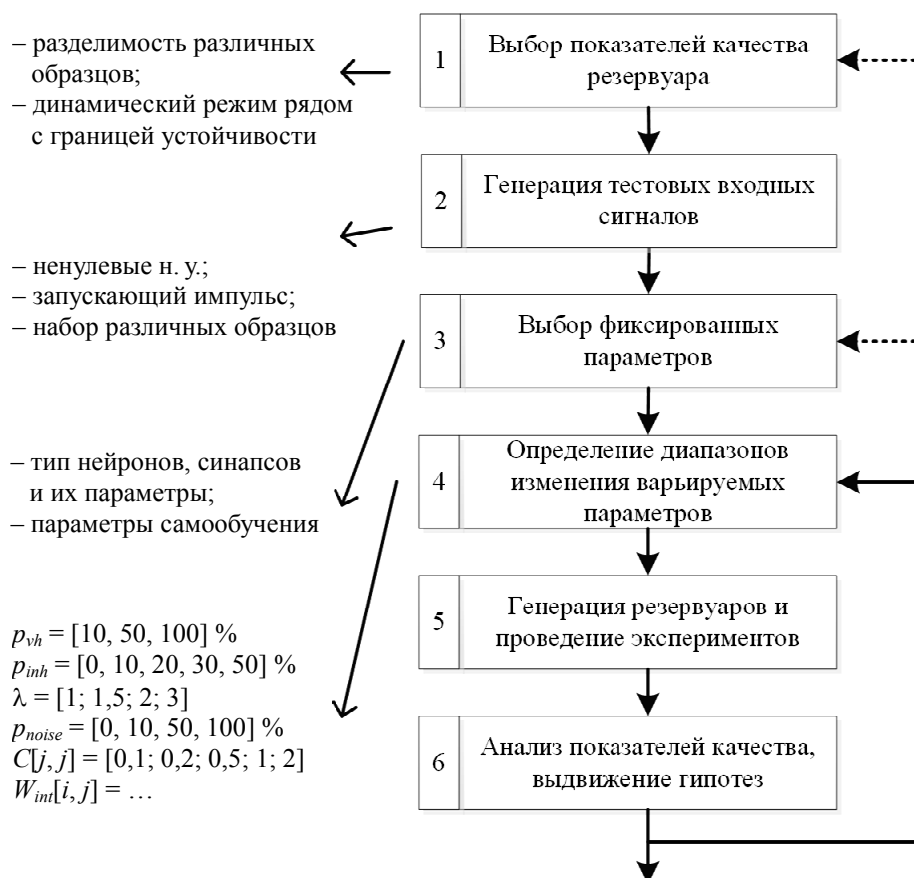


Рис. 3. Схема, иллюстрирующая этап построения методики синтеза импульсной РНС
 Fig. 3. Scheme illustrating the stage of building a method for synthesizing pulsed RNN

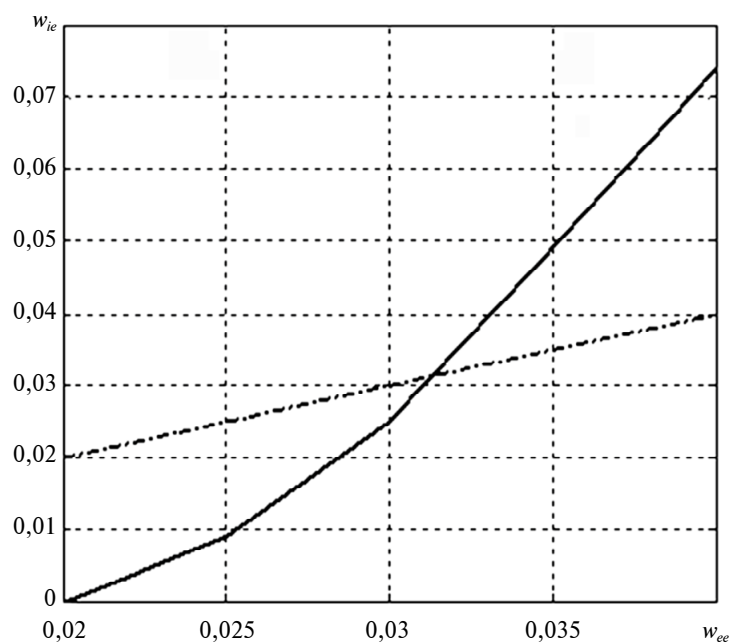


Рис. 4. Граница в плоскости двух параметров, разделяющая динамику на два типа
 Fig. 4. The boundary in the plane of the two parameters, dividing the dynamics into two types
 (—) граница устойчивости; (---) $w_{ee} = w_{ie}$

Все, что лежит ниже кривой, соответствует незатухающей памяти, а выше — затухающей. Уменьшение уровня шума ведет к сдвигу характеристики вниз, а увеличение — к сдвигу вверх. Экспериментально установлено, что данная граница качественно справедлива не только для текущей, но и для большинства остальных вариантов структур.

В среднем для различных структур при отсутствии подавления незатухающая память появляется при значениях w_{ee} в диапазоне от 0,02 до 0,025 (для $\lambda = 3$, $p_{inh} = 20\%$, статических синапсах, $p_{noise} = 100\%$, $c_{ii} = 0$, $c_{ie} = c_{ei} = c_{ee} = 0,5$). В результате обобщения полученных результатов построена таблица зависимости типа динамики от параметров (табл. 3).

В таблице для обозначения типа динамики используются следующие символы:
 — аперiodически затухающая дина-

мика;  — колебательная затухающая динамика;  — колебательная незатухающая динамика.

С помощью этой таблицы могут быть выбраны сочетания параметров C , λ и w_{ee} для нахождения импульсной РНС вблизи границы устойчивости.

РНС из нейронов Ижикевича. После исследования РНС из IaF нейронов было выполнено исследование РНС, состоящих из нейронов Ижикевича.

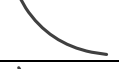
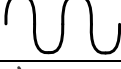
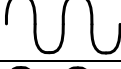
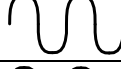
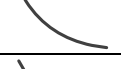
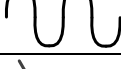
В отличие от IaF нейронов у нейронов Ижикевича важным является момент подачи входного образа. Дело в том, что в самом начале у нейронов Ижикевича идет переходный процесс, поэтому лучше подавать входные сигналы спустя 0,03 с, когда этот процесс закончится. В работе входные сигналы подавались спустя 0,1 с.

Таблица 3

Зависимость динамики от параметров

Table 3

The dependence of the dynamics of the parameters

c_{ee}	$w_{ee} = 0,03$	$w_{ee} = 0,04$	$w_{ee} = 0,05$	$w_{ee} = 0,06$	$w_{ee} = 0,15$
$\lambda = 2$					
0,7, 1					
0,5					
$\lambda = 1,5$					
0,8					
$\lambda = 1$					
5, 10, 15					
3					
2					
0,5, 1					
0,1					



В начальных экспериментах с НС из нейронов Ижикевича исследовалась реакция НС на запускающий импульс, подающийся одновременно на все нейроны. При этом шум на вход НС не подавался, а начальные условия нейронов соответствовали точкам равновесия.

Первым рассматривается случай, при котором отсутствуют подавляющие нейроны ($P_{inh} = 0$). На динамику влияют три параметра: λ , c_{ee} , w_{ee} . В зависимости от этих параметров динамика может быть различной. С увеличением параметров λ , c_{ee} , w_{ee} время активности РНС также растет, что говорит об «усилении» динамики этими параметрами.

С ростом λ увеличивается дальность связей, а с ростом c_{ee} — их плотность.

Можно выделить следующие особенности:

после подачи входного импульса нейроны генерируют импульс, потом уходят в молчание на 2 мс, после чего начинается основная динамика;

основная динамика характеризуется начальным нарастанием частот генерации импульсов с последующим их уменьшением;

нейроны заканчивают генерировать импульсы в порядке, зависящем от структуры РНС. Чаще всего общая динамика сужается в один-два затухающих кластера; кластеры формируются из соседних нейронов, имеющих связи друг с другом.

Влияние подавляющих нейронов. Введем следующие обозначения: $[inh \rightarrow inh]$ — связи от подавляющих к подавляющим нейронам; $[inh \rightarrow ex]$ — связи от подавляющих к возбуждающим нейронам; $[ex \rightarrow inh]$ — связи от возбуждающих к подавляющим нейронам; $[ex \rightarrow ex]$ — связи от возбуждающих к возбуждающим нейронам.

РНС, состоящая только из подавляющих нейронов, демонстрировала нулевую активность: нейроны лишь реагировали на начальный импульс. В случае, если РНС состояла из подавляющих и возбуждающих нейронов, то динамика определялась как

процентом возбуждающих (подавляющих) нейронов, так и плотностью C и силой W связей между нейронами.

С увеличением процента возбуждающих нейронов динамика НС усиливалась. На основании анализа множества экспериментов были сделаны следующие выводы.

1. Динамика возбуждающих нейронов при наличии связей $[ex \rightarrow inh]$ провоцирует динамику подавляющих нейронов.

Начальный импульс вызывает две волны импульсов от возбуждающих и подавляющих нейронов. Возбуждающие нейроны не связаны друг с другом, поэтому их дальнейшая динамика отсутствует. Но они успевают возбудить подавляющие нейроны. Если убрать связи $[inh \rightarrow ex]$, то реакция будет уже другой.

Возбуждающие нейроны слабо возбуждают возбуждающие и сильно — подавляющие. Подавляющие нейроны пассивны. Поэтому реакцию можно интерпретировать как слабую динамику возбуждающих нейронов, активирующую динамику подавляющих нейронов.

2. Увеличение силы связей $[inh \rightarrow inh]$ ускоряет процесс затухания реакции подавляющих нейронов.

3. При отсутствии связей $[inh \rightarrow ex]$ и $[ex \rightarrow inh]$ активность НС распадается на активность возбуждающих и подавляющих нейронов. Подавляющие нейроны при этом ослабляют динамику возбуждающих нейронов лишь тем, что создают «пустоты» в структуре, не реагирующие на возбуждающие нейроны.

4. Эффективность подавления динамики зависит от двух типов связей: от возбуждающих к подавляющим $[ex \rightarrow inh]$ и от подавляющих к возбуждающим $[inh \rightarrow ex]$. В отсутствии связей $[ex \rightarrow inh]$ ($c_{ei} = 0$) подавляющие нейроны быстро «успокаиваются» и больше не «тревожат» возбуждающие нейроны. В отсутствии связей $[inh \rightarrow ex]$ ($c_{ie} = 0$) динамика возбуждающих нейронов не ослабляется подавляющими нейронами. Вместе же связи $[ex \rightarrow inh]$ и $[inh \rightarrow ex]$ ослабляют динамику.

Серия запускающих импульсов. Следующая серия экспериментов была направлена на изучение реакции НС в ответ на несколько запускающих импульсов. Вначале параметры были заданы следующими:

$$\lambda = 3, C = \begin{bmatrix} 0 & 0,5 \\ 0,5 & 0,5 \end{bmatrix}, W_{int} = \begin{bmatrix} 0,02 & 0,02 \\ 0,02 & 0,02 \end{bmatrix}.$$

При подаче импульсов в моменты времени 0,1 и 0,5 с реакция была как на рис. 5.

Можно говорить, что это две примерно идентичные реакции.

Анализ всех проведенных в данном подразделе экспериментов позволяет сформулировать следующие положения:

- реакция РНС на запускающий импульс — результат взаимодействия между подавляющими и возбуждающими нейронами;
- один или несколько возбуждающих импульсов способны запустить динамику РНС на некоторое время;
- чем сильнее и продолжительней возбуждение РНС, тем дольше последующее состояние неактивности, восстановления, и наоборот, чем короче возбуждение, тем быстрее восстановление;
- реакция РНС на второй импульс зависит от момента подачи этого импульса (если РНС возбудить в начальный момент восстановления, то реакция будет практически нулевая; если РНС возбудить ближе к концу момента восстановления, то реакция будет отличаться от

обычной реакции; если РНС возбудить после восстановления, то реакция будет похожа на первую реакцию);

- на появление незатухающей активности влияет как сила связей, так и пропорция между возбуждающими и подавляющими связями. При превалировании возбуждающих связей реакция, как правило, одиночная и длительная (при равной пропорции связей и нормальной силе реакция, как правило, в форме колебаний; при силе связей не больше 0,01 реакция на импульсы всегда затухающая; при силе связей на уровне 0,02 реакция становится более длительной и иногда колебательной; при силе связей на уровне 0,03 реакция в большинстве случаев колебательная);

- на реакцию влияет равномерность распределения связей (при появлении локализованных участков только с возбуждающими связями они могут функционировать независимо от РНС и подчинять себе ее динамику).

Эксперименты с шумом. Эксперименты с запускающими импульсами показали, как реакция НС зависит от пропорции подавляющих нейронов P_{inh} , силы связи W_{int} и плотности связей C . Следующим этапом исследования является подача на вход НС шума различной интенсивности.

При небольшой силе связи (равной 0,01) и сбалансированной плотности связей реакция НС представляет собой случайный процесс (рис. 6).

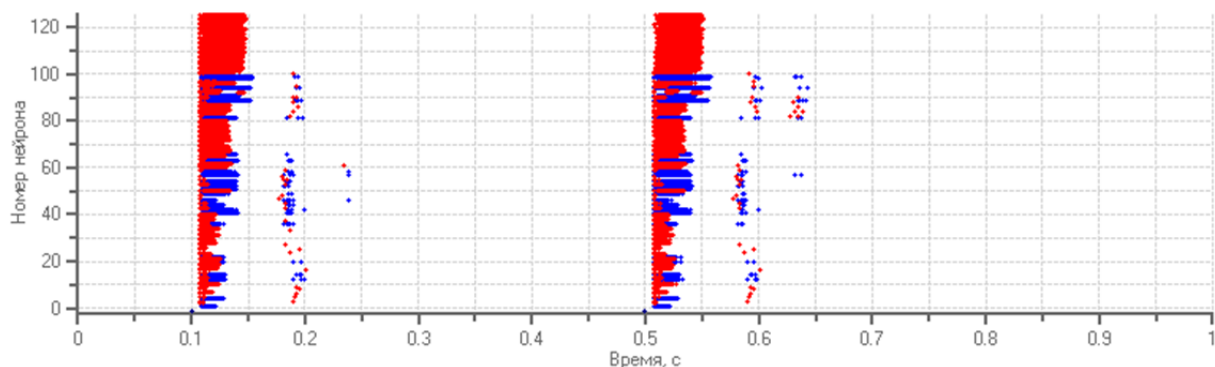


Рис. 5. График активности импульсной РНС

Fig. 5. Graph of the activity of pulsed RNN

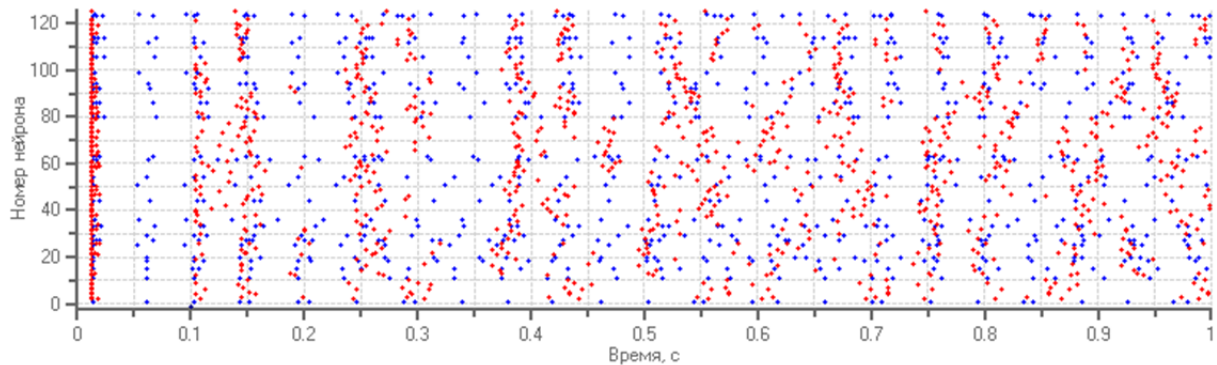


Рис. 6. График активности импульсной РНС

Fig. 6. Graph of the activity of pulsed RNS

Последующие эксперименты позволили сделать следующие выводы.

1. Реакция на шум по сути эквивалентна реакции на большое количество запускающих импульсов.

2. Небольшой, но постоянный уровень шума улучшает показатели импульсной РНС. Это можно объяснить тем, что за счет шума РНС находится в предвозбужденном состоянии, и ей проще реагировать на входные стимулы.

3. Существует граничное значение силы шума, превышая которое РНС не способна нормально реагировать на входные последовательности.

Применение полученных результатов

Анализ результатов экспериментального исследования позволил построить обобщенную методику синтеза импульсных РНС и встроить ее в схему решения задачи распознавания динамических образов. Соответствующая схема представлена на рис. 7.

В самом начале в зависимости от типа, сложности и размерности входных данных выбирается способ их нормализации, кодирования и подачи на вход РНС. Затем осуществляется выбор параметров импульсной РНС исходя из сложности решаемой задачи и используемых показателей качества резервуара. В соответствии с решаемой задачей синтезируются и обучаются считыватели. После этого выполняется

оценка качества распознавания, и при необходимости осуществляется возврат на один из предыдущих шагов.

Пример. Далее на простом примере покажем, как можно решать задачи с помощью рассматриваемой модели импульсной РНС в составе МНС.

В качестве простых данных возьмем два различных импульсных пространственно-временных образа. Эти образы представляют собой волны импульсов, распространяющиеся в дискретном пространстве размерности $n_{in} = 8$ в различных направлениях (рис. 8).

Задача ставится следующим образом: если на входе образ 1, то на выходе системы распознавания должен быть класс 1; если на входе образ 2, то на выходе — класс 2; если на входе что-то другое или ничего нет, то на выходе — класс 3. Общее число классов $n_{class} = 3$.

В качестве нейронов в соответствии с рекомендациями методики выберем нейроны типа IaF, в качестве синапсов — химические синапсы. Количество нейронов в резервуаре зададим равным $n_{nm} = 125$ (решетка $5 \times 5 \times 5$). Параметр $\lambda = 2$ означает, что каждый нейрон связан случайно с соседями в окрестности длины 2. $P_{inh} = 20\%$ нейронов выберем подавляющими, а $p_{ex} = 80\%$ — возбуждающими. Матрицы C , W_{int} равны соответственно

$$C = \begin{bmatrix} 0,1 & 0,4 \\ 0,2 & 0,3 \end{bmatrix}, \quad W_{int} = \begin{bmatrix} 0,02 & 0,01 \\ 0,01 & 0,05 \end{bmatrix}.$$

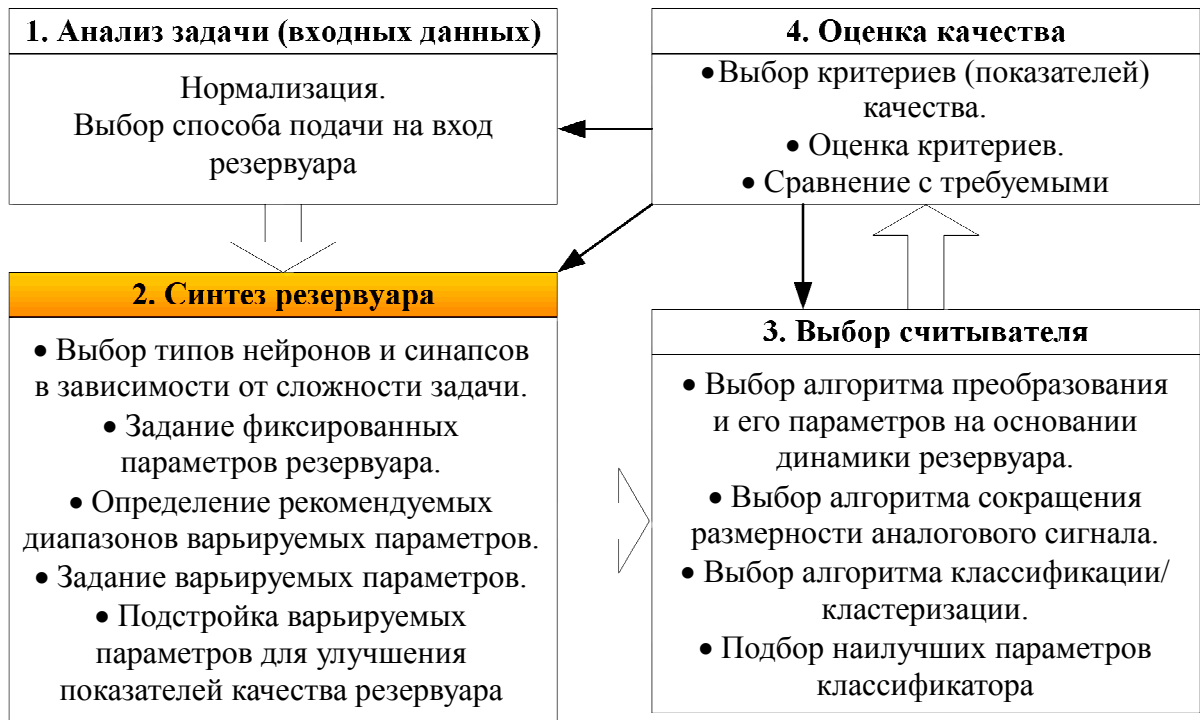


Рис. 7. Схема решения задачи распознавания образов с помощью МНС

Fig. 7. Pattern of solving the problem of pattern recognition using MNN

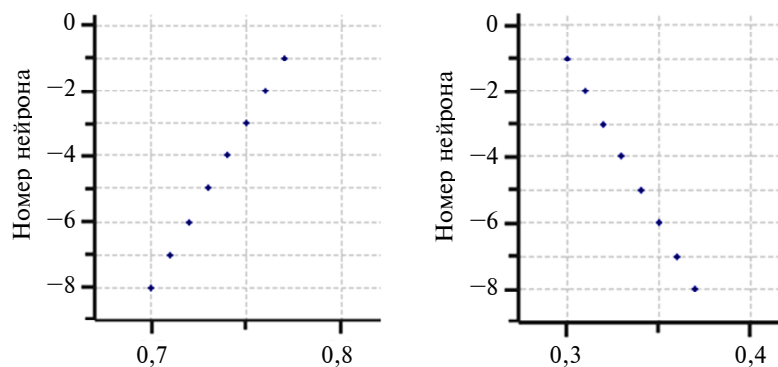


Рис. 8. Примеры классифицируемых образов

Fig. 8. Examples of classified images

Каждый из $n_{vh} = n_{in} = 8$ входов случайно связан с $p_{vh} = 20\%$ нейронов, силы химических синапсов от входов к подавляющим и возбуждающим нейронам равны $w_{vh(chem)}^i = w_{vh(chem)}^e = 0,03$. Постоянная времени $\tau_{syn} = 3$ мс для всех химических синапсов, кроме синапсов от подавляющих нейронов — для них она равна $\tau_{syn} = 6$ мс. Задержка во всех химических синапсах

одинакова и равна $\tau_{delay} = 1$ мс. Электрические синапсы отсутствуют.

Для анализа импульсной динамики и преобразования импульсного сигнала в непрерывный подходит алгоритм на основе замены импульсов затухающими экспонентами. С учетом динамики входного сигнала постоянную времени экспонент зададим равной $\tau_{window} = 5$ мс.

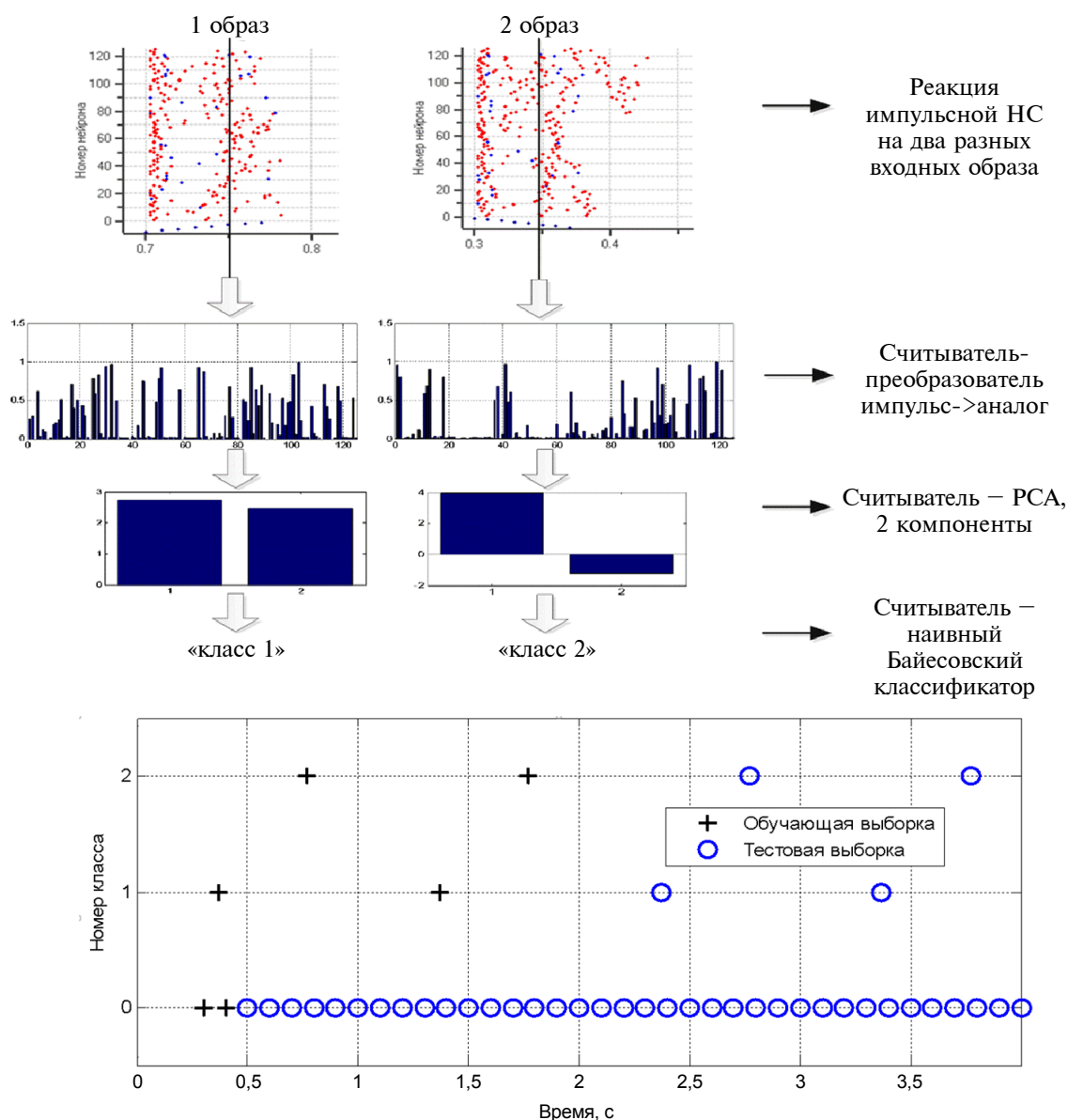


Рис. 9. Процедура работы МНС при подаче на ее вход двух образов разных классов (сверху); результаты классификации (снизу)

Fig. 9. The procedure of the MNN when submitting to its input two images of different classes (above); classification results (below)

Для классификации в целях визуализации процесса используется сжатие по алгоритму PCA до двух главных компонент ($n_{features} = 2$) и наивный Байесовский классификатор.

На рис. 9 показана процедура работы МНС при подаче на ее вход двух образов разных классов. На разные образы импульсная РНС реагирует по-разному, что видно

из верхних графиков. Затем в момент считывания состояние РНС (125 сигналов) преобразуется из импульсного в непрерывное, выполняется сокращение размерности состояния до 2 и осуществляется классификация.

Для тестирования системы она обучалась на небольшом количестве $n_{train} = 6$ обучающих образов, а затем на ее вход бы-

ли поданы $n_{test} = 100$ тестовых образов. На рис. 9 снизу крестиками помечены результаты классификации обучающих примеров, а ноликами — тестовых. Видно, что система правильно различает образы классов 1, 2, 3, т. е. система справилась с поставленной задачей без ошибок.

Анализ приведенных, а также аналогичных им результатов позволил сделать вывод о том, что разработанная методика может успешно применяться для решения самых разнообразных задач распознавания динамических образов (ЭКГ, речевых сигналов, диагностических сигналов и др.) с использованием МНС в качестве основного инструмента.

Заключение

Выделены все основные параметры МНС и определена их иерархия. Построенная декомпозиция параметров импульсной РНС и считывателей позволяет выделить группы взаимосвязанных параметров, разделить все параметры на классы в зависимости от их важности. Сформировано три класса параметров: варьируемые, настраиваемые и фиксированные. Для каждого из параметров определены рекомендуемые значения или диапазоны значений, в которых следует производить их поиск.

Определен класс задач распознавания, предъявлены требования к входным и выходным данным. Для решаемой задачи

распознавания динамических образов введено качественное определение сложности на основании различных эвристических характеристик. Выдвинута гипотеза о том, что сложность импульсной РНС должна соответствовать сложности решаемой задачи, после чего предложены несколько показателей, по которым можно оценивать сложность импульсной РНС.

Выполнен большой объем экспериментов, нацеленных на получение рекомендуемых значений параметров импульсной РНС с точки зрения достижения наилучших показателей качества при решении задачи распознавания динамических образов. Рассмотрены модели импульсных РНС с нейронами IaF и нейронами Ижикевича. Получены характеристики, связывающие показатели динамики импульсных РНС со значениями параметров, задающих структуру РНС, силы связей, тип нейронов, задержки и др. Выявлено и сформулировано множество системных закономерностей между значениями параметров РНС и ее динамическими режимами.

Полученные результаты позволяют в дальнейшем сформулировать формальную методику синтеза импульсных РНС в составе МНС для решения задачи распознавания динамических образов. Применение этой методики упрощает процесс решения аналогичных задач и обеспечивает увеличение показателей качества распознавания.

СПИСОК ЛИТЕРАТУРЫ

1. Hastie T., Tibshirani R., Friedman J. The elements of statistical learning: Data mining, inference, and prediction. 2nd ed. Springer, 2016. 745 p.
2. Kuhn M. Applied predictive modeling. 2nd ed. Springer, 2016. 600 p.
3. De Mulder W., Bethard S., Moens M.-F. A survey on the application of recurrent neural networks to statistical language modeling // Computer Speech and Language. 2015. Vol. 30. No. 1. Pp. 61–98.
4. Бендерская Е.Н., Никитин К.В. Рекуррентная нейронная сеть как динамическая система и подходы к ее обучению // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2013. № 4. С. 29–40.
5. Haykin S. Neural networks and learning machines. 3rd ed. Pearson, 2011. 936 p.
6. Siegelmann H.T. Neural and super-turing computing // Minds and Machines. 2003. Vol. 13. Pp. 103–114.
7. Amari S.A. Dynamics of pattern formation in lateral-inhibition type neural fields // Biological Cybernetics. 1977. Vol. 27. Pp. 77–87.
8. Hopfield J.J. Neural networks and physical systems with emergent collective computational abilities // PNAS. 1982. Vol. 79. Pp. 2554–2558.
9. Amit D.J. Modelling brain function. The world of attractor neural networks. Cambridge: Cambridge University Press, 1989. 504 p.
10. Greff K., Srivastava R., Koutník J., Steunebrink B.R., Schmidhuber J. LSTM: A search space odyssey // IEEE Transactions on Neural Networks and Learning Systems. 2017. Vol. 28. No. 10.



11. **Jaeger H.** Tutorial on training recurrent neural networks, covering BPPT, RTRL, EKF and the «echo state network» approach // GMD Report 159. German National Research Center for Information Technology, 2002. 48 p.
12. **Atiya A.F., Parlos A.G.** New results on recurrent network training-unifying the algorithms and accelerating Convergence // IEEE Transactions on Neural Networks. 2000. Vol. 11. No. 3. Pp. 697–709.
13. **Hammer B., Schrauwen B., Steil J.J.** Recent advances in efficient learning of recurrent networks // ESANN 2009. Bruges, 2009. Pp. 213–226.
14. **Dmitriev A.S., Panas A.I., Starkov S.O.** Storing and recognizing information based on stable cycles of one-dimensional maps // Physics Letters A. 1991. Vol. 155. No. 8, 9. Pp. 494–499.
15. **Борисюк Г.Н., Борисюк Р.М., Казанович Я.Б., Лузянина Т.Б., Турова Т.С., Цымбалюк Г.** Осцилляторные нейронные сети. Математические результаты и приложения // Математическое моделирование. 1992. Т. 4. № 1. С. 3–43.
16. **Kaneko K.** Clustering, coding, switching, hierarchical ordering and control in a network of chaotic elements // Physica D: Nonlinear Phenomena. 1990. Vol. 41. No. 2. Pp. 137–172.
17. **Angelini L., Carlo F., Marangi C., Pellicoro M., Nardullia M., Stramaglia S.** Clustering data by inhomogeneous chaotic map lattices // Physical Review Letters. 2000. Vol. 85. No. 3. Pp. 554–557.
18. **Eckhorn R., Bauer R., Jordan W., Brosch M., Kruse W., Munk M., Reitboeck H.J.** Coherent oscillations: a mechanism of feature linking in the visual cortex? // Biological Cybernetics. 1988. Vol. 60. Pp. 121–130.
19. **Hinton G.E., Sejnowski T.J.** Learning and relearning in Boltzmann machines // Parallel Distributed Processing. Cambridge: The MIT Press, 1986.
20. **Hochreiter S., Schmidhuber J.** Long short-term memory // Neural Computation. 1997. Vol. 9 (8). Pp. 1735–1780.
21. **Buonomano D.V., Maass W.** State-dependent computations spatiotemporal processing in cortical networks // Neuroscience. 2009. Vol. 10. Pp. 113–125.
22. **Gallicchi C., Micheli A., Pedrelli L.** Deep reservoir computing: A critical experimental analysis // Neurocomputing. 2017. Vol. 268. No. 13. Pp. 87–99.
23. **Goudarzi A., Teuscher C.** Reservoir computing: Quo Vadis? // Proc of the 3rd ACM Int. Conf on Nanoscale Computing and Communication. New York, NY, USA. 2016.
24. **Maass W., Joshi P., Sontage E.D.** Principles of real-time computing with feedback applied to cortical microcircuit models // Advances in Neural Information Processing Systems. 2006. Vol. 18. Pp. 835–842.
25. **Walter F., Röhrbein F., Knoll A.** Computation by time // Neural Processing Letters. 2016. Vol. 44. No. 1. Pp. 103–124.
26. **Jaeger H.** The «echo state» approach to analyzing and training recurrent neural networks // GMD report 148. German National Research Center for Information Technology. St. Augustin, 2001. 43 p.
27. **Aoun M.A., Boukadoum M.** Chaotic liquid state machine // Internat. J. of Cognitive Informatics and Natural Intelligence. 2015. Vol. 9. No. 4. Pp. 1–20.
28. **Schrauwen B., Verstraeten D., Campenhout J.V.** An overview of reservoir computing theory, applications and implementations // Proc. of the 15th European Symposium on ANN. 2007. Pp. 471–482.
29. **Larger L., Baylun-Fuentes A., Martinenghi R., Udaltsov V.S., Chembo Y.K., Jacquot M.** High-speed photonic reservoir computing using time-delay-based architecture: Million words per second classification // Phys. Rev. X. 2017. Vol. 7. Art. № 011015.
30. **Najibi E., Rostami H.** SCESN, SPESN, SWESN: Three recurrent neural echo state networks with clustered reservoirs for prediction of nonlinear and chaotic time series // Applied Intelligence. 2015. Vol. 43. No. 2. Pp. 460–472.
31. **Ortín S., Soriano M.C., Pesquera L., Brunner D., San-Martín D., Fischer I., Mirasso C.R., Gutiérrez J.M.** A unified framework for reservoir computing and extreme learning machines based on a single time-delayed neuron // Scientific Reports. 2015. Vol. 5. Art. No. 14945.
32. **Trentin E., Scherer S., Schwenker F.** Emotion recognition from speech signals via a probabilistic echo-state network // Pattern Recognition Letters. 2015. Vol. 66. No. C. Pp. 4–12.
33. **Zhang Y., Li P., Jin Y., Choe Y.** A digital liquid state machine with biologically inspired learning and its application to speech recognition // IEEE Trans Neural Netw Learn Syst. 2015. Vol. 26(11). Pp. 2635–49.
34. **Rosselly J.L., Alomar M.L., Morro A., Oliver A., Canals V.** High-density liquid-state machine circuitry for time-series forecasting // Int. J. Neural Syst. 2016. Vol. 26 (5). Art. No. 1550036.
35. **Charles A.S., Yin D., Rozell C.J.** Distributed sequence memory of multidimensional inputs in recurrent networks // The Journal of Machine Learning Research. 2017. Vol. 18. No. 1. Pp. 181–217.

36. Jin Y., Li P. Performance and robustness of bio-inspired digital liquid state machines // *Neuro-computing*. 2017. Vol. 226. No. C. Pp. 145–160.
37. Løkse S., Bianchi F.M., Jessen R. Training echo state networks with regularization through dimensionality reduction // *Cogn. Comput.* 2017. Vol. 9. Art. No. 364.
38. Bourdoukan R., Deneve S. Enforcing balance allows local supervised learning in spiking recurrent networks // *Proc. of the 28th Int. Conf. on NIPS*. Montreal, Canada. 2015. Pp. 982–990.
39. Wang H., Yan X. Optimizing the echo state network with a binary particle swarm optimization algorithm // *Knowledge-Based Systems*. 2015. Vol. 86. No. C. Pp. 182–193.
40. Wojcik G.M., Wany M. Bray-Curtis metrics as measure of liquid state machine separation ability in function of connections density // *Procedia Computer Science*. 2015. Vol. 51. No. C. Pp. 2979–2983.
41. Бендерская Е.Н., Никитин К.В. Анализ модели импульсной рекуррентной нейронной сети в составе машины неустойчивых состояний // *Нейрокомпьютеры: разработка, применение*. 2014. № 1. С. 17–22.
42. Gerstner W., Kistler W.M., Naud R., Paninski L. *Neuronal dynamics: From single neurons to networks and models of cognition*. Cambridge University Press, 2014.
43. Kasabov N., Capecchi E. Spiking neural network methodology for modelling, classification and understanding of EEG spatio-temporal data measuring cognitive processes // *Information Sciences: an International Journal*. 2015. Vol. 294. No. C. Pp. 565–575.
44. Izhikevich E.M. *Dynamical systems in neuroscience: the geometry of excitability and bursting*. Cambridge: The MIT Press, 2007. 505 p.
45. Izhikevich E.M. Which model to use for cortical spiking neurons? // *IEEE Trans on Neural Networks*. 2004. Vol. 15. No. 5. Pp. 1063–1070.
46. Trappenberg T.P. *Fundamentals of computational neuroscience*. 2nd ed. Oxford University Press, 2010. 416 p.
47. Yousefi A., Dibazar A.A., Berger T.W. Synaptic dynamics: linear model and adaptation algorithm // *Neural Netw.* 2014. Vol. 56. Pp. 49–68.
48. Smith E., Lewicki M.S. Efficient coding of time-relative structure using spikes // *Neural Computation*. 2004. Vol. 17. Pp. 19–45.

Статья поступила в редакцию 04.10.2018.

REFERENCES

1. Hastie T., Tibshirani R., Friedman J. *The elements of statistical learning: data mining, inference, and prediction*, 2nd ed. Springer, 2016. 745 p.
2. Kuhn M. *Applied Predictive Modeling*, 2nd ed. Springer, 2016. 600 p.
3. De Mulder W., Bethard S., Moens M.-F. A survey on the application of recurrent neural networks to statistical language modeling. *Computer Speech and Language*, 2015, Vol. 30, No. 1, Pp. 61–98.
4. Benderskaja E.N., Nikitin K.V. Recurrent neural network as dynamical system and approaches to its training. *St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunication and Control Systems*, 2013, No. 4, Pp. 29–40. (rus)
5. Haykin S. *Neural networks and learning machines*, 3rd ed., Pearson, 2011, 936 p.
6. Siegelmann H.T. Neural and super-turing computing. *Minds and Machines*, 2003, Vol. 13, Pp. 103–114
7. Amari S.A. Dynamics of pattern formation in lateral-inhibition type neural fields. *Biological Cybernetics*, 1977, Vol. 27, Pp. 77–87.
8. Hopfield J.J. Neural networks and physical systems with emergent collective computational abilities, *PNAS*, 1982, Vol. 79, Pp. 2554–2558.
9. Amit D.J. *Modelling brain function. The world of attractor neural networks*. Cambridge: Cambridge University Press, 1989, 504 p.
10. Greff K., Srivastava R., Koutník J., Steunebrink B.R., Schmidhuber J. LSTM: A search space odyssey. *IEEE Transactions on Neural Networks and Learning Systems*, 2017, Vol. 28, No. 10.
11. Jaeger H. Tutorial on training recurrent neural networks, covering BPPT, RTRL, EKF and the «echo state network» approach. *GMD Report 159*, German National Research Center for Information Technology, 2002, 48 p.
12. Atiya A.F., Parlos A.G. New results on recurrent network training-unifying the algorithms and accelerating convergence. *IEEE Transactions on Neural Networks*, 2000, Vol. 11, No. 3, Pp. 697–709.
13. Hammer B., Schrauwen B., Steil J.J. Recent advances in efficient learning of recurrent networks. *ESANN 2009*, Bruges, 2009. Pp. 213–226.
14. Dmitriev A.S., Panas A.I., Starkov S.O. Storing and recognizing information based on stable cycles of one-dimensional maps. *Physics Letters A*, 1991, Vol. 155, No. 8, 9, Pp. 494–499.
15. Borisjuk G.N., Borisjuk R.M., Kazanovich Ja.B., Luzjanina T.B., Turova T.S., Cym-



- baljuk G. Oscillatory neural networks. Mathematics and applications. *Matematicheskoe modelirovanie [Mathematical Models and Computer Simulations]*, 1992, Vol. 4, No. 1, Pp. 3–43. (rus)
16. Kaneko K. Clustering, coding, switching, hierarchical ordering and control in a network of chaotic elements. *Physica D: Nonlinear Phenomena*, 1990, Vol. 41, No. 2, Pp. 137–172.
17. Angelini L., Carlo F., Marangi C., Pellicoro M., Nardullia M., Stramaglia S. Clustering data by inhomogeneous chaotic map lattices. *Physical Review Letters*, 2000, Vol. 85, No. 3, Pp. 554–557.
18. Eckhorn R., Bauer R., Jordan W., Brosch M., Kruse W., Munk M., Reitboeck H.J. Coherent oscillations: a mechanism of feature linking in the visual cortex? *Biological Cybernetics*, 1988, Vol. 60, Pp. 121–130.
19. Hinton G.E., Sejnowski T.J. *Learning and relearning in Boltzmann machines. Parallel Distributed Processing*. Cambridge: The MIT Press, 1986.
20. Hochreiter S., Schmidhuber J. Long short-term memory. *Neural Computation*, 1997, Vol. 9 (8), Pp. 1735–1780.
21. Buonomano D.V., Maass W. State-dependent computations spatio-temporal processing in cortical networks. *Neuroscience*, 2009, Vol. 10, Pp. 113–125.
22. Gallicchi C., Micheli A., Pedrelli L. Deep reservoir computing: A critical experimental analysis. *Neurocomputing*, 2017, Vol. 268, No. 13, Pp. 87–99.
23. Goudarzi A., Teuscher C. Reservoir Computing: Quo Vadis? *Proc. of the 3rd ACM Int. Conf. on Nanoscale Computing and Communication*, New York, NY, USA, 2016.
24. Maass W., Joshi P., Sontage E.D. Principles of real-time computing with feedback applied to cortical microcircuit models. *Advances in Neural Information Processing Systems*, 2006, Vol. 18. Pp. 835–842.
25. Walter F., Röhrbein F., Knoll A. Computation by time. *Neural Processing Letters*, 2016, Vol. 44, No. 1, Pp. 103–124.
26. Jaeger H. *The «echo state» approach to analyzing and training recurrent neural networks: GMD report 148*. German National Research Center for Information Technology. St. Augustin, 2001, 43 p.
27. Aoun M.A., Boukadoum M. Chaotic liquid state machine. *International Journal of Cognitive Informatics and Natural Intelligence*, 2015, Vol. 9, No. 4, Pp. 1–20.
28. Schrauwen B., Verstraeten D., Campenhout J.V. An overview of reservoir computing theory, applications and implementations. *Proc. of the 15th European Symposium on ANN*, 2007, Pp. 471–482.
29. Larger L., Baylyn-Fuentes A., Martinenghi R., Udaltsov V.S., Chembo Y.K., Jacquot M. High-speed photonic reservoir computing using time-delay-based architecture: Million words per second classification. *Phys. Rev. X*, 2017, Vol. 7, No. 011015.
30. Najibi E., Rostami H. SCESN, SPESN, SWESN: Three recurrent neural echo state networks with clustered reservoirs for prediction of nonlinear and chaotic time series. *Applied Intelligence*, 2015, Vol. 43, No. 2, Pp. 460–472.
31. Ortín S., Soriano M.C., Pesquera L., Brunner D., San-Martín D., Fischer I., Mirasso C.R., Gutiérrez J.M. A Unified framework for reservoir computing and extreme learning machines based on a single time-delayed neuron. *Scientific Reports*, 2015, Vol. 5, Art. No. 14945.
32. Trentin E., Scherer S., Schwenker F. Emotion recognition from speech signals via a probabilistic echo-state network. *Pattern Recognition Letters*, 2015, Vol. 66, No. C, Pp. 4–12.
33. Zhang Y., Li P., Jin Y., Choe Y. A digital liquid state machine with biologically inspired learning and its application to speech recognition. *IEEE Trans Neural Netw Learn Syst*, 2015, Vol. 26(11), Pp. 2635–49.
34. Rosselly J.L., Alomar M.L., Morro A., Oliver A., Canals V. High-density liquid-state machine circuitry for time-series forecasting. *Int. J. Neural Syst.*, 2016, Vol. 26 (5), Art. No. 1550036.
35. Charles A.S., Yin D., Rozell C.J. Distributed sequence memory of multidimensional inputs in recurrent networks. *The Journal of Machine Learning Research*, 2017, Vol. 18, No. 1, Pp. 181–217.
36. Jin Y., Li P. Performance and robustness of bio-inspired digital liquid state machines. *Neurocomputing*, 2017, Vol. 226, No. C, Pp. 145–160.
37. Løkse S., Bianchi F.M., Jessen R. Training echo state networks with regularization through dimensionality reduction. *Cogn. Comput.*, 2017, Vol. 9, Art. No. 364.
38. Bourdoukan R., Deneve S. Enforcing balance allows local supervised learning in spiking recurrent networks. *Proc. of the 28th Internat. Conf. on NIPS*, Montreal, Canada, 2015, Pp. 982–990.
39. Wang H., Yan X. Optimizing the echo state network with a binary particle swarm optimization algorithm. *Knowledge-Based Systems*, 2015, Vol. 86, No. C, Pp. 182–193.
40. Wojcik G.M., Wany M. Bray-Curtis metrics as measure of liquid state machine separation ability in function of connections density. *Procedia Computer Science*, 2015, Vol. 51, No. C, Pp. 2979–2983.
41. Benderskaya E.N., Nikitin K.V. Analysis of a spiking recurrent neural network as a part of the liquid state machine. *Neurocomputers*, 2014, No. 1, Pp. 17–22 (rus)

42. Gerstner W., Kistler W.M., Naud R., Paninski L. *Neuronal dynamics: From single neurons to networks and models of cognition*. Cambridge University Press, 2014.

43. Kasabov N., Capecchi E. Spiking neural network methodology for modelling, classification and understanding of EEG spatio-temporal data measuring cognitive processes. *Information Sciences: an International Journal*, 2015, Vol. 294, No. C, Pp. 565–575.

44. Izhikevich E.M. *Dynamical systems in neuroscience: the geometry of excitability and bursting*. Cambridge: The MIT Press, 2007, 505 p.

45. Izhikevich E.M. Which model to use for cortical spiking neurons? *IEEE Trans on Neural Networks*, 2004, Vol. 15, No. 5, Pp. 1063–1070.

46. Trappenberg T.P. *Fundamentals of computational neuroscience*, 2nd ed. Oxford University Press, 2010, 416 p.

47. Yousefi A., Dibazar A.A., Berger T.W. Synaptic dynamics: linear model and adaptation algorithm. *Neural Netw.*, 2014, Vol. 56, Pp. 49–68.

48. Smith E., Lewicki M.S. Efficient coding of time-relative structure using spikes. *Neural Computation*, 2004, Vol. 17, Pp. 19–45.

Received 04.10.2018.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

НИКИТИН Кирилл Вячеславович

NIKITIN Kirill V.

E-mail: execiter@mail.ru

DOI: 10.18721/JCSTCS.11411
УДК 004.93:591.18

ОЦЕНКА УРОВНЯ УМСТВЕННОЙ РАБОТОСПОСОБНОСТИ УЧАЩИХСЯ НА ОСНОВЕ АНАЛИЗА СИГНАЛОВ ЭЭГ

Л.А. Станкевич, С.С. Аманбаева, А.В. Самочадин

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

Представлены результаты исследований в области применения неинвазивных интерфейсов «мозг-компьютер» для анализа степени умственной усталости учащихся. Предложено использовать электроэнцефалографические (ЭЭГ) сигналы, позволяющие определить вызванные событиями потенциалы. Подробно описан набор алгоритмов для предварительной обработки сигналов ЭЭГ и распознавания вызванного потенциала P300, возникающего через 300 мс после зрительного стимула. Основное внимание уделено эксперименту по распознаванию волны P300 по информации, снимаемой прибором типа Muse headset. Приведены предварительные результаты по точности распознавания волны P300 у разных людей с использованием различных типов классификаторов. Разработана методика применения P300 для оценки степени умственной усталости учащихся. Проведен ряд экспериментов, подтверждающих возможность такой оценки по разработанной методике.

Ключевые слова: умственная усталость, интерфейс «мозг-компьютер», электроэнцефалографические сигналы, волна P300, декодирование, Muse headset.

Ссылка при цитировании: Станкевич Л.А., Аманбаева С.С., Самочадин А.В. Оценка уровня умственной работоспособности учащихся на основе анализа сигналов ЭЭГ // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2018. Т. 11. № 4. С. 151–161. DOI: 10.18721/JCSTCS.11411.

EVALUATION OF STUDENTS' MENTAL PERFORMANCE LEVEL BASED ON EEG SIGNAL ANALYSIS

L.A. Stankevich, S.S. Amanbaeva, A.V. Samochadin

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

The article presents the results of studies on using non-invasive brain-computer interfaces (BCI) for analyzing the degree of mental fatigue of students. It is proposed to use electroencephalographic (EEG) signals, allowing to determine the potentials caused by events. A set of algorithms for preprocessing EEG signals and recognizing the evoked potential of P300 arising 300 ms after a visual stimulus is described in detail. The main focus is on the P300 wave recognition experiment from information captured by a Muse headset. Preliminary results on the accuracy of P300 wave recognition in different people using various types of classifiers are given. A methodology has been developed for using P300 to assess the students' mental fatigue. A number of experiments have been carried out confirming the possibility of such assessment using the developed methodology.

Keywords: mental fatigue, brain-computer interface, electroencephalographic signals, P300 wave, decoding, Muse headset.

Citation: Stankevich L.A., Amanbaeva S.S., Samochadin A.V. Evaluation of students' mental performance level based on EEG signal analysis. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2018, Vol. 11, No. 4, Pp. 151–161. DOI: 10.18721/JCSTCS.11411.

Введение

В настоящее время актуальной задачей является определение степени усталости учащихся при выполнении умственной работы, например, при выполнении заданий или подготовке к зачетам и экзаменам. Для оценки умственной усталости могут применяться различные аппаратные и программные средства, такие как портативные измерители частоты пульса, давления, состояния кожи и пр., а также различные компьютерные тесты. Также существуют попытки использовать для этих целей т. н. *нейроинтерфейсы*.

В основе большинства нейроинтерфейсов — принцип считывания и декодирования электрических импульсов, возникающих вследствие активности головного мозга человека. Мыслительные процессы сопровождаются изменением электрической активности, отражающей работу множества активных нейронных сетей мозга, которая может быть считана непосредственно с активных участков мозга инвазивно, т. е. с хирургическим вмешательством, или неинвазивно, например, с поверхности головы. Электрические сигналы, управляющие мышцами, могут считываться также неинвазивно с соответствующих нервов периферийной нервной системы. Декодирование (распознавание и классификация) считанных электрических сигналов может выполняться с использованием развитых в последнее время интеллектуальных методов обработки информации и принятия решений. Поэтому можно считать нейроинтерфейсы развитием интеллектуальных интерфейсов, в которых вместо устройств ввода аудио- или видеоинформации применяются устройства ввода сигналов активности мозга или периферийных отделов нервной системы.

По мнению экспертов, нейроинтерфейсы — перспективная отрасль с огромным потенциалом. По этой причине исследования в данной области ведутся практически во всех развитых странах мира. Такие страны, как США и Китай, не только проводят научные исследования, но и выпу-

стили уже коммерчески доступные нейроинтерфейсы массового спроса, такие как Emotive Insight (США), Neurosky (США) и Brainlink (Китай) с собственными магазинами приложений для настольных и мобильных устройств на популярных платформах типа iOS, Android, Windows, Linux, MacOSX.

В России на сегодняшний день существуют научные коллективы международного уровня, исследующие нейрофизиологические аспекты, позволяющие разрабатывать современные нейроинтерфейсы. Среди таких коллективов, занимающихся проблемами разработки нейроинтерфейсов, можно отметить лабораторию нейрофизиологии и нейрокомпьютерных интерфейсов МГУ имени М.В. Ломоносова (Москва), лабораторию математической нейробиологии обучения Института высшей нервной деятельности и нейрофизиологии РАН (Москва), кафедру нейродинамики и нейробиологии биологического факультета ННГУ (Нижний Новгород), Институт медико-биологических проблем (Москва), НИИ нейрохирургии имени академика Н.Н. Бурденко РАМН (Москва), кафедру системного анализа и управления Санкт-Петербургского политехнического университета Петра Великого, лабораторию нейроэргономики и интерфейсов мозг-компьютер НБИК Курчатовского института (Москва), лабораторию изучения нейрофизиологических механизмов психической деятельности НИИ нейрокибернетики имени А.Б. Когана Южного федерального университета (Ростов-на-Дону), Институт физиологии имени И.П. Павлова (Санкт-Петербург) и др.

В настоящее время созданы неинвазивные нейроинтерфейсы, позволяющие распознавать некоторые моторные команды мозга и мышечные сигналы и использовать результаты их декодирования, в частности, для управления разнообразными внешними устройствами. Исследователи и разработчики нейроинтерфейсов предполагают, что уже в ближайшее время на их основе могут быть созданы эффективные средства управления протезами, средствами реабили-

литации, ассистивными устройствами. Что очень важно отметить в рамках этой статьи, неинвазивные нейроинтерфейсы позволяют также распознавать эмоциональные и ментальные состояния человека.

На практике чаще всего применяются нейроинтерфейсы, регистрирующие и декодирующие сигналы нейрональной активности различных зон мозга. Они различаются способами и средствами регистрации биоэлектрических сигналов мозга и входят в группу т. н. *интерфейсов «мозг-компьютер»* (ИМК). На основе неинвазивных ИМК могут быть созданы достаточно эффективные системы оценки ментальных состояний человека.

Основная цель данной статьи — обоснование возможности оценки степени умственной усталости учащихся с использованием неинвазивных ИМК, а также обсуждение предварительных результатов экспериментов по использованию варианта ИМК, построенного на основе прибора Muse headset, в системе оценки умственной усталости.

Неинвазивные ИМК

При выполнении умственных работ человек может находиться в разных ментальных состояниях, связанных с типом обрабатываемой информации, когнитивной нагрузкой и пр. Некоторые состояния четко определяют его физическую, эмоциональную или умственную усталость. Любой вид усталости, как правило, ведет к снижению работоспособности. Важно определить степень умственной усталости учащихся, анализируя которую можно предложить им некоторые процедуры по снятию усталости. Наиболее пригодны для этих целей компьютерные системы оценки степени усталости, использующие неинвазивные ИМК, классифицирующие ЭЭГ-паттерны ментальных состояний.

Применение неинвазивных способов регистрации активности мозга, в частности, ЭЭГ, перспективно в плане безопасности для пациента, простоты и компактности устройства, а также в экономиче-

ском отношении. ЭЭГ наиболее часто используется в ИМК, предназначенных как для медицинских целей, например, для реабилитации пациентов, так и для немедицинских целей, в частности, для дистанционного управления различными игровыми приложениями здоровыми людьми [1]. Такие ИМК в целом функционируют по схеме, представленной на рис. 1. Регистрация многоканальных сигналов ЭЭГ ведется с поверхности кожи головы (скальпа). Эти сигналы проходят предобработку путем фильтрации, удаления артефактов и выделения ЭЭГ-паттернов, соответствующих ожидаемым реальным или воображаемым действиям. Далее путем специальной обработки сигналов извлекаются признаки, позволяющие отличать ЭЭГ-паттерны друг от друга, и проводится распознавание ожидаемого действия или классификация нескольких действий. Такие ИМК должны быть предварительно обучены на примерах правильного распознавания или классификации действий. В режиме обучения в памяти устройства для каждого действия фиксируются свои наборы признаков (шаблоны). В рабочем режиме эти шаблоны сопоставляются с текущими наборами признаков, и по результатам сопоставления выполняется распознавание или классификация действий.

Следует отметить, что ЭЭГ имеет относительно низкое пространственное разрешение (около одного квадратного сантиметра), по сравнению с инвазивным методом, т. е. регистрацией сигналов с вживляемых электродов, где пространственное разрешение может быть около одного квадратного миллиметра. Это препятствует четкой локализации сигнала, но при предварительной обработке сигнал может быть улучшен путем его пространственной фильтрации и специальных методов выделения источников. Заметим, что ЭЭГ обладает хорошим временным разрешением (сотни миллисекунд), что также в значительной степени определяет перспективность использования данного метода для разработки ИМК.

Методы и приборы регистрации сигналов ЭЭГ и классификации ментальных состояний

Электрическая активность, регистрируемая через ЭЭГ, отображает колебания потенциалов на дендритных переходах нейронов сети мозга. Связь ЭЭГ с элементарными электрическими процессами на уровне нейронов нелинейная. В статистическом плане ЭЭГ представляется как результат суммирования электрических градуальных потенциалов многих нейронов, работающих в значительной степени независимо. Отклонения от случайного распределения в данном случае зависят от функционального состояния мозга (сон, уровни бодрствования) и от характера процессов, вызывающих элементарные потенциалы (спонтанная и вызванная активность). В случае временной синхронизации активности нейронов наблюдается отклонение от случайного распределения. Это может проявляться в повышении амплитуды суммарных потенциалов и увеличении когерентности между элементарными и суммарными процессами.

При исследовании биоэлектрических сигналов, связанных с событием, как экзогенных (реакции на внешние раздражители — звуковые, зрительные стимулы), так и эндогенных (принятие решений, планирование движений), в электрофизиологии используются методы когерентного накопления, при котором увеличивается соотношение сигнал/шум. В этом случае спонтанная ритмическая активность, не связанная с событием (экзогенным или эндогенным), является шумом. Вызванные потенциалы генерируются теми же нейрональными структурами, что и спонтанная ритмика. Для улучшения соотношения сигнал/шум необходимо усреднение большого количества реакций на предъявляемые стимулы и четкая синхронизация по времени события и регистрируемого ответа. Вызванный потенциал имеет малую амплитуду по отношению к спонтанной активности ЭЭГ, которая в данном случае выступает как шум. Например, потенциалы, вызванные зрительными стимулами, имеют амплитуду до 10 мкВ, а средний амплитудный уровень ЭЭГ составляет 50 мкВ.

Таким образом, одной из черт сигналов ЭЭГ является высокая доля шума в виде спонтанной (не связанной с событием) ритмической составляющей, отражающей общее функциональное состояние человека. Другой особенностью ЭЭГ является ее чувствительность к изменению общего состояния человека и условий проведения исследования. Например, на сигнал ЭЭГ могут влиять внешние раздражители, не связанные с выполняемой задачей, такие как шум, освещение и др. Сигналы ЭЭГ чувствительны к состоянию человека: «уровню бодрствования» и мобилизации человека, «внутренним» состояниям типа эмоций, переживаний. Вариативность и изменчивость паттернов ЭЭГ приводит к тому, что каждое регистрируемое событие происходит на разном «уровне» фоновой ритмики ЭЭГ, что может накладывать отпечаток на сигнал при малом количестве накоплений.

При регистрации сигналов ЭЭГ могут применяться разные подходы. В профессиональном варианте неинвазивного ИМК используется стандартный подход. ЭЭГ регистрируется монополярно от 19 отведений: Fp1, Fp2, F7, F3, Fz, F4, F8, T3, C3, Cz, C4, T4, T5, P3, Pz, P4, T6, O1, O2, в соответствии со стандартной системой 10-20. В качестве референта используются электроды, которые закрепляются на мочках обеих ушей; в качестве заземляющего электрода — электрод во фронтальной части головы, чуть выше зоны предполагаемого отведения Fpz. Примером является 32-канальный отечественный электроэнцефалограф Mitsar (Санкт-Петербург). Регистрация сигналов ЭЭГ в нем проводится в выбранных частотных диапазонах, например, в полосе 0,50–30 Гц с применением сетевого фильтра 45–55 Гц. Для записи ЭЭГ используется программа регистрации и обработки ЭЭГ, соответствующая выбранному типу электроэнцефалографа.

На практике для повышения эффективности классификации сигналы ЭЭГ могут подвергаться частотной фильтрации, что позволяет ориентироваться на частотный

диапазон с наибольшим количеством информативных признаков сигнала. Так, при анализе сигналов, соответствующих воображаемым моторным командам, фильтрация может проводиться отдельно в узкой полосе частот (0,5–15 Гц), в частотной полосе сенсомоторного ритма (10–15 Гц) и в широкой полосе частот (0,5–30 Гц), поскольку информативными могут являться как низкочастотные, так и высокочастотные составляющие сигналов ЭЭГ. В [2] показано, что разные типы классификаторов сигналов ЭЭГ демонстрируют различную точность распознавания сигнала в широкой полосе частот.

Для работы с информативным сигналом необходимо очистить его от артефактов. Наиболее вероятным вариантом удаления артефактов в условиях реализации обработки сигнала в реальном времени является удаление движения глаз по амплитудным характеристикам сигнала, т. е. по превышению определенного порога. Например, могут быть удалены фрагменты записи или пробы ЭЭГ, содержащие медленный сигнал мощностью 50 μV в полосе 0–2 Гц, и быстрый сигнал мощностью более 35 μV в полосе 20–35 Гц. Такое удаление артефактов не изменяет первоначальный сигнал ЭЭГ и может успешно применяться в экспериментах реального времени.

Обработка сигналов ЭЭГ с целью извлечения признаков для дальнейшей классификации воображаемых движений включает: поиск областей в наблюдаемых каналах ЭЭГ, где ожидается необходимый сигнал (ЭЭГ-паттернов); преобразование сигналов в частотной или временной областях; извлечение соответствующих наборов признаков. Поиск ЭЭГ-паттернов, т. е. фрагментов сигналов ЭЭГ, соответствующих ожидаемым воображаемым движениям и несущих их образ, требует ручной или автоматической разметки сигналов ЭЭГ. Эта операция проводится в соответствии с процедурой или парадигмой регистрации сигналов ЭЭГ при проведении эксперимента.

Для различения ЭЭГ-паттернов используются наборы признаков, получаемые в частотной или временной областях. Извлечение признаков в частотной области может производиться с использованием спектрального анализа, например, на базе преобразования Фурье. Лучшие признаки могут дать вейвлет-преобразования, которые позволяют учитывать параметр времени. Наряду со спектральным анализом в некоторых ИМК используются методы регрессионного анализа сигналов ЭЭГ, что дает возможность извлекать информацию во времени.

Классификация выполняется путем сопоставления набора признаков, полученного из текущего ЭЭГ-паттерна, с шаблонными наборами признаков, полученными в процессе обучения системы классификации. Для классификации ЭЭГ-паттернов наиболее часто применяются методы линейной дискриминации и опорных векторов, байесовские классификаторы, искусственные нейронные сети. В настоящее время начинают использоваться более продвинутые методы классификации ЭЭГ-паттернов, основанные на нейронных сетях с глубоким обучением, а также на классификаторах, построенных на римановой геометрии [3].

Далее рассмотрим подробнее метод, построенный на римановой геометрии, поскольку он будет использоваться в прикладной части этой работы [4, 5].

Недавно введенные в область ИМК классификаторы, построенные на римановой геометрии, отличаются от традиционно используемых в этой области классификаторов тем, что вместо оценивающих пространственных фильтров и (или) селекции признаков они выполняют отображение данных прямо в геометрическое пространство с некоторой пригодной метрикой. В этом пространстве данные могут быть легко преобразованы путем усреднения, сглаживания, интерполяции, экстраполяции и классификации. Например, в случае ЭЭГ-данных отображение может сводиться к вычислению некоторой формы ковариационной матрицы на этих данных.

В принципе это отображение основано на предположении, что энергия и пространственное распределение ЭЭГ-источников может рассматриваться как фиксированное для данного ментального состояния, и такая информация может быть кодирована ковариационной матрицей. Риманова геометрия изучает пространства с гладкой кривизной, которые могут быть локально и линейно аппроксимированы. Такие пространства называются *многообразием* (manifold), а их аппроксимация в каждой точке — *касательным пространством* (tangent space). В римановом многообразии касательное пространство снабжено метрикой, гладко изменяющейся от точки к точке. В неевклидовой нотации это соответствует дистанции между двумя любыми точками (каждая точка может быть попыткой) и соответствующей нотации центра масс любого числа точек. Поэтому вместо использования евклидовой дистанции применяется риманова метрика, адаптированная к геометрии многообразия и к манере, в которой должны отображаться данные. Из числа матричных многообразий, используемых для ИМК-приложений, предпочтительно многообразие симметричных положительно определенных (Symmetric positive definite — SPD) матриц, которое имеет дело с ковариационными матрицами, полученными из ЭЭГ-сигналов. Проблема обучения классификатора сигналов ЭЭГ может успешно решаться на основе такого многообразия. Чтобы проиллюстрировать эту проблему, рассмотрим случай SPD-матриц. Квадрат дистанции между двумя SPD-матрицами C_1 и C_2 в закрытой форме выражается как

$$\delta^2(C_1, C_2) = \sum_n \log^2 \lambda_n(C_1^{-1}C_2), \quad (1)$$

где $\lambda_n(M)$ означает n собственных значений матрицы M . Для C_1 и C_2 SPD эта дистанция не негативная, симметричная и равная нулю, только в случае, если $C_1 = C_2$. Интересно, что, когда C_1 и C_2 являются средними двух классов, собственные векторы матриц ($C_1^{-1}C_2$) используют-

ся для определения фильтров общих пространственных паттернов (Common spatial pattern — CSP), в то время как собственные их числа используются для вычисления их римановой дистанции. Используя эту дистанцию в форме уравнения (1), центр масс G набора $\{C_1, \dots, C_K\}$ SPD-матриц, также называемый геометрическим средним, является уникальным решением следующей оптимизационной проблемы:

$$\arg \min_G \sum_k \delta^2(C_k, G). \quad (2)$$

Заметим, что это определение — аналог определения арифметического среднего, которое является решением оптимизационной проблемы (2), когда используется евклидова дистанция. В отличие от арифметического среднего, геометрическое среднее не имеет решения в закрытой форме. Существует быстрый и робастный итеративный алгоритм вычисления геометрического среднего.

Наиболее простые методы классификации, основанные на римановой геометрии, позволяют классифицировать попытки, отображаемые через некоторые формы ковариационной матрицы, путем применения простых методов ближайшего соседа, используя эксклюзивно нотацию римановой дистанции (1) и, возможно, нотацию геометрического среднего (2). Например, классификатор на минимальной римановой дистанции от среднего (Minimum distance to mean — MDM) вычисляет геометрическое среднее для каждого класса, используя обучающие данные, а затем назначает неразмеченной попытке класс, соответствующий ближайшему среднему. Другой класс римановых классификаторов включает методы, проектирующие точки данных на касательное пространство и последующую классификацию, используя стандартные классификаторы, такие как LDA (Linear Discriminant Analysis), SVM (Support Vector Machine), логическую регрессию и пр.

Набор алгоритмов для распознавания волны P300

Один из первых подходов к распознаванию волны P300 в ответ на предъявляемый стимул представлен в статье [6]. В настоящее время имеется много вариантов решения этой задачи, но наибольший интерес для нас имеет вариант, предложенный для системы, использующей простые приборы для регистрации сигналов ЭЭГ, например, на базе EmotiveEroc headset [7]. В этом варианте набор алгоритмов для классификации волны P300 по многоканальным сигналам ЭЭГ (используется до 14 каналов-отведений) включает временной высокочастотный фильтр, пространственный фильтр, усреднение попыток, классификатор типа LDA, использующий правило голосования для получения финального решения. Процедура удаления артефактов, связанных с морганием и мускульными движениями, не применялась в этом наборе, поскольку она не улучшала выполнение задачи. За счет усреднения попыток влияние артефактов снижалось до минимума.

В нашем варианте предполагается использовать еще более простой прибор для регистрации сигналов ЭЭГ — Muse headset. Предлагаемый для решения задачи распознавания ответа P300 далее обсуждается подробно.

Эксперименты по использованию Muse headset для работы с P300

В работе [8] приведены результаты исследования возможности декодирования волны P300 с использованием наиболее простого прибора регистрации сигналов ЭЭГ Muse 2016 EEG headset. Каналы ЭЭГ в Muse не являются хорошо позиционированными для P300. Однако ЭЭГ-потенциалы диффузируют на скальп в целом, и оказалось возможным наблюдать P300 ERP даже в случае, когда электроды расположены далеко от зоны интереса. Эксперимент был поставлен для выяснения надежности этих измерений. Применялись про-

стые визуальные стимулы: частотный фоновый стимул, состоящий из вертикальных полос, и редкий целевой стимул с горизонтальными полосами. Задача состояла в автоматическом распознавании волны P300 и подсчете количества раз, когда происходила регистрация этого потенциала, по сравнению с количеством раз, когда субъект видел горизонтальные полосы.

Описанный эксперимент был повторен нами с целью проверки корректности изложенных результатов. Стимулы длиной 200 мс подавались с интервалами 600 мс со случайным сдвигом ± 100 мс. Записывались шесть проб по 2 мин с общим числом стимулов нецелевых 960 и целевых 184. При обработке применялась простая фильтрация сигналов в полосе частот 1–30 Гц. Затем данные регистрировались эпохами от –100 до 800 мс после предъявления стимулов. Не применялись коррекции фильтрованного сигнала, но удалялись эпохи, где сигнал превышал 100 мВ. Это позволило исключить главные артефакты, связанные с морганием. Результаты эксперимента отражены на графиках усредненных вызванных потенциалов для нецелевых стимулов (розовые) и целевых (зеленые) на всех четырех электродах (рис. 1).

Как видно на графиках, имеет место явная идентификация P300 на электродах TP9 и TP10. Можно видеть также слабую активность на электродах AF8 и AF7.

Ссылочный электрод Muse headset FPz находится очень близко к двум фронтальным электродам. Это означает, что любой потенциал, локальный по отношению к ссылочному электроду, будет негативно проецироваться на электроды AF7, AF8. Поэтому на этих электродах видны только моргания, а на электродах TP9, TP10 можно зафиксировать присутствие потенциала P300. Путем усреднения эпох можно явно идентифицировать волну вызванного потенциала P300.

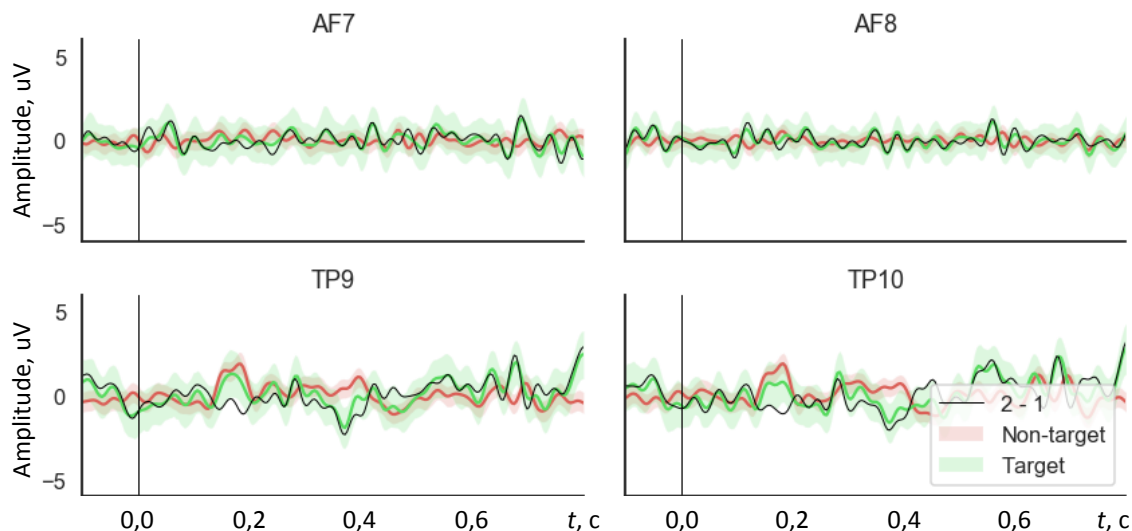


Рис. 1. Результаты эксперимента по выявлению волны P300

Fig. 1. The results of the experiment to identify the wave P300

Чтобы изучить силу ответа P300 в одичных пробах взяли набор специальных алгоритмов для классификации [9, 10]. В частности, использовались семь различных наборов алгоритмов:

- **Vect + LR:** Векторизация + Логистическая регрессия. Этот вариант может рассматриваться как стандартный декодирующий EEG набор алгоритмов.
- **Vect + RegLDA:** Векторизация + Регуляризованный LDA. Этот вариант часто используется в нейроинтерфейсах с P300.
- **Xdawn + RegLDA:** Xdawn + Регуляризованный LDA.
- **XdawnCov + TS:** Xdawn Ковариация + Касательное пространство.
- **XdawnCov + MDM:** Xdawn Ковариация + MDM.
- **ERPCov + TS:** EprКовариация + Касательное пространство. Один из фаворитов набора алгоритмов, использующих классификатор на римановой геометрии.
- **ERPCov + MDM:** EprКовариация + MDM. Очень простой, но эффективный для малого количества каналов классификатор на римановой геометрии.

Оценка выполнялась через кросс-валидацию (25 % данных использовались в качестве валидационного набора), с AUC в качестве метрики (AUC – вероятность наилучшей метрики для бинарной несбалансированной классификационной проблемы) (рис. 2).

Наилучшим набором оказался **ERPCov + TS**, показавший среднее значение AUC, равное 0,63. Этот результат рассматривается далее, как хороший.

Основываясь на этих начальных результатах, можно говорить о том, что имеется возможность наблюдать P300 с помощью Muse headset. При AUC = 0,8 его можно использовать в нейроинтерфейсных приложениях, если не требуется большей точности.

Эксперименты по оценке умственной усталости с использованием Muse headset

В этих экспериментах также использовался Muse headset, с помощью которого регистрировались ЭЭГ по четырем каналам, и специализированный классификатор, обнаруживающий эти команды по волне P300. Была выдвинута гипотеза о том, что при значительной умственной усталости снижается вероятность распознавания визуальных стимулов, вследствие притупления внимания, и замедляется их восприятие. Чтобы доказать это в процессе экспериментов на монитор подавались различные нецелевые и целевые зрительные стимулы, на которые должны были по-разному реагировать субъекты. Нецелевыми символами были наборы горизонтальных линий и черное поле на экране, а целевыми – набор вертикальных линий. Наиболее информативным оказался эксперимент с нецелевыми стимулами в виде черного поля и с целевыми – набором вертикальных линий.

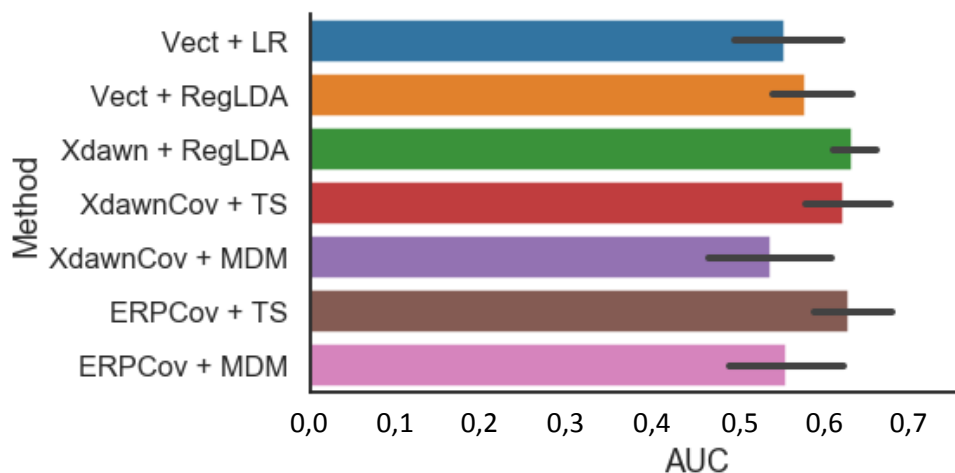


Рис. 2. Оценка точности распознавания P300

Fig. 2. Evaluation of recognition accuracy P300

	TP9	AF7	AF8	TP10
Амплитуда, V	−0,000016	−0,000013	−0,000014	−0,000015
Время, с	0,330679	0,344952	0,356671	0,339694

	TP9	AF7	AF8	TP10
Амплитуда, V	−0,000015	−0,000015	−0,000014	−0,000015
Время, с	0,354718	0,370418	0,373798	0,357347

Рис. 3. Оценка величины временного сдвига отрицательного пика волны P300

Fig. 3. Estimation of the magnitude of the time shift of the negative peak of the P300 wave

Анализировались результаты семи субъектов (студентов СПбПУ), участвующих в экспериментах. Отмечено некоторое уменьшение точности распознавания импульса P300 у некоторых субъектов после значительной умственной нагрузки: от 0,63 до 0,59, что говорит о малой надежности этой оценки для фиксации факта умственной усталости учащихся. Результаты лучшего эксперимента по оценке величины временного сдвига отрицательного пика волны P300 приведены на рис. 3. Здесь видно, что у субъекта в бодром состоянии (верхняя таблица) по всем четырем каналам время фиксации отрицательного пика находится в диапазоне 0,330679–0,356671, а у этого же

субъекта после значительной умственной нагрузки — в диапазоне 0,354718–0,373798. Это подтверждает возможность уверенного распознавания степени умственной усталости по ЭЭГ-сигналам активности мозга путем измерения величины временного сдвига отрицательного пика волны P300.

Заключение

Результаты исследований, представленных в статье, подтвердили возможность оценки степени умственной усталости учащихся с использованием неинвазивных интерфейсов «мозг-компьютер». В эксперименте анализировались электроэнцефалографические сигналы, снимаемые самым

простым и относительно дешевым прибором типа Muse headset. В частности, подтверждена возможность распознавания вызванного потенциала P300, возникающего в диапазоне 200–400 мс после зрительного стимула. Точность распознавания волны P300 у различных людей с использованием различных типов классификаторов достигала 63 %. В экспериментах показано, что точность распознавания P300 при подаче целевых стимулов снижается, при значительной умственной усталости уменьшается от 0,63 до 0,59 по AUC. Оценка величин

временного сдвига отрицательного пика волны P300 при распознавании целевых визуальных стимулов показала увеличение задержки появления отрицательного пика в среднем на 0,025 с. Это свидетельствует, вероятно, о притуплении внимания и замедлении восприятия стимулов при умственной усталости учащихся.

Работа подготовлена в ходе реализации проекта в рамках Постановления Правительства РФ от 09.04.2010 № 218 при финансовой поддержке Министерства образования и науки РФ. Договор № 03.G25.31.0247 от 28.04.2017.

СПИСОК ЛИТЕРАТУРЫ

1. Станкевич Л.А., Сонькин К.М., Шемякина Н.В., Нагорнова Ж.В., Хоменко Ю.Г., Перец Д.С., Коваль А.В. Классификация ЭЭГ-паттернов воображаемых движений пальцами одной руки, выполняемых в заданном ритме // Физиология человека. 2016. Т. 42. № 1. С. 40–51.
2. Sonkin K.M., Stankevich L.A., Khomenko Ju.G., Nagornova Zh.V., Shemyakina N.V. Development of electroencephalographic pattern classifiers for real and imaginary thumb and index finger movements of one hand // Artificial Intelligence in Medicine. 2015. Vol. 63. No. 2. Pp. 107–123.
3. Fabien Lotte, et al. A review of classification algorithms for EEG-based Brain-computer interfaces: A 10-2018 Update // Journal of Neural Engineering. 2018. Vol. 15, No. 3. Pp. 1–28.
4. Congedo M., Barachant A., Bhatia R. Riemannian geometry for EEG-based brain computer interfaces; a primer and a review // Brain-Computer Interfaces, Taylor Francis. 2017. No. 4(3). Pp. 155–174.
5. Cognedo M., et al. Classification of covariance matrices using Riemannian-based kernel for

- BCI applications // IEEE Trans. on Signal Processing. 2016. No. 65 (9). Pp. 2211–2220.
6. Picton W.T. The P300 wave of the human event-related potential // Journal of Clinical Neurophysiology. 1992. No. 9(4). Pp. 456–479.
7. Duvinage M., Castermans T., Petieau M., Hoellinger T., et al. Performance of the Emotiv-Epoc headset for P300-based applications // Bio-Medical Engineering OnLine. 2013. Pp. 3–15.
8. Barachant A. URL: <http://alexandre.barachant.org/blog/2017/02/05/P300-with-muse.html> (Дата обращения: 09.11.2018).
9. Notebooks. URL: https://eeg-notebooks.readthedocs.io/en/latest/available_notebooks.html#old-notebooks (Дата обращения: 09.11.2018).
10. Mayaud L., Cabanilles S., Van Langenhove A., et al. Brain-computer interface for the communication of acute patients: a feasibility study and a randomized controlled trial comparing performance with healthy participants and a traditional assistive device // Brain-Computer Interfaces. 2016. Vol. 3-4. Pp. 197–215.

Статья поступила в редакцию 10.11.2018.

REFERENCES

1. Stankevich L.A., Sonkin K.M., Perets D.S., Koval A.V., Shemyakina N.V., Nagornova Z.V., Khomenko J.G. EEG pattern decoding of rhythmic individual finger imaginary movements of one hand. *Human Physiology*, 2016, Vol. 42, No. 1, Pp. 40–51. (rus)
2. Sonkin K.M., Stankevich L.A., Khomenko Ju.G., Nagornova Zh.V., Shemyakina N.V. Development of electroencephalographic pattern classifiers for real and imaginary thumb and index finger movements of one hand. *Artificial Intelligence in Medicine*, 2015, Vol. 63, No. 2, Pp. 107–123.

3. Fabien Lotte, et al. A review of classification algorithms for EEG-based Brain-computer interfaces: A 10-year update. *Journal of Neural Engineering*, 2018, Vol. 15, No. 3, Pp. 1–28.
4. Congedo M., Barachant A., Bhatia R. Riemannian geometry for EEG-based Brain computer interfaces; a primer and a review. *Brain-Computer Interfaces*, Taylor Francis, 2017, No. 4(3), Pp. 155–174.
5. Cognedo M., et al. Classification of covariance matrices using Riemannian-based kernel for BCI applications. *IEEE Trans. on Signal Processing*, 2016, No. 65 (9), Pp. 2211–2220.

6. **Picton W.T.** The P300 wave of the human event-related potential. *Journal of Clinical Neurophysiology*, 1992, No. 9(4), Pp. 456–479.

7. **Duvinage M., Castermans T., Petieau M., Hoellinger T., et al.** Performance of the EmotivEpoC headset for P300-based applications. *BioMedical Engineering OnLine*, 2013, Pp. 3–15.

8. **Barachant A.** Available: <http://alexandre.barachant.org/blog/2017/02/05/P300-with-muse.html> (Accessed: 09.11.2018).

9. Notebooks. Available: https://eeg-notebooks.readthedocs.io/en/latest/available_notebooks.html#old-notebooks (Accessed: 09.11.2018).

10. **Mayaud L., Cabanilles S., Van Langenhove A., et al.** Brain-computer interface for the communication of acute patients: a feasibility study and a randomized controlled trial comparing performance with healthy participants and a traditional assistive device. *Brain-Computer Interfaces*, 2016, Vol. 3-4, Pp. 197–215.

Received 10.11.2018.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

СТАНКЕВИЧ Лев Александрович

STANKEVICH Lev A.

E-mail: Stankevich_lev@inbox.ru

АМАНБАЕВА Сабина Сергеевна

AMANBAEVA Sabina S.

E-mail: Sabina2704@mail.ru

САМОЧАДИН Александр Викторович

SAMOCHADIN Alexandr V.

E-mail: Samochadin@gmail.com

DOI: 10.18721/JCSTCS.11412

УДК 004

ИЗВЛЕЧЕНИЕ МОДЕЛЕЙ БИЗНЕС-ПРОЦЕССОВ ИЗ ОБУЧАЮЩИХ МАТЕРИАЛОВ

Д.А. Тимофеев, А.В. Самочадин

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

Моделирование играет важную роль в анализе и оптимизации бизнес-процессов организации. Автоматизация построения моделей бизнес-процессов особенно значима в тех сферах деятельности, где преобладает интеллектуальный труд, но выполняемые процессы редко документируются — в частности, в области разработки программного обеспечения. В такой ситуации важным источником информации о процессах являются обучающие материалы, создаваемые разработчиками как часть общей документации, а также для обмена опытом и популяризации технологий. Существующие алгоритмы извлечения моделей бизнес-процессов накладывают серьезные ограничения на вид и структуру входных данных. В статье рассмотрен подход к извлечению моделей процессов из обучающих материалов, в которых могут одновременно присутствовать описания различных процессов, а также фрагменты текста, которые не должны учитываться при моделировании. Для учета этих факторов предлагаемый метод принимает во внимание не только стандартные извлекаемые из текста признаки, но и структуру, и разметку документа.

Ключевые слова: бизнес-процесс, моделирование бизнес-процессов, извлечение процессов, обработка естественного языка, анализ текста.

Ссылка при цитировании: Тимофеев Д.А., Самочадин А.В. Извлечение моделей бизнес-процессов из обучающих материалов // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2018. Т. 11. № 4. С. 162–170. DOI: 10.18721/JCSTCS.11412.

PROCESS EXTRACTION FROM EDUCATIONAL TEXTS

D.A. Timofeev, A.V. Samochadin

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

Business process modeling plays an important role in analysis and optimization of organizational processes. Automation of process models is particularly crucial in domains where processes involve mostly intellectual activity that is not properly documented. Software development is an example of a domain with these properties. Educational materials like instructions and guides, blog posts, or conference talks are an important source of information about the processes in this case. Known algorithms of process extraction pose strict requirements to the input text. In this paper, we propose an approach to process extraction from complex sources containing descriptions of multiple processes and text blocks unrelated to the process model. In order to account for these aspects, the method considers not only standard lexical and syntactic properties of the text but also its structure and markup.

Keywords: business process, process modeling, process extraction, natural language processing, text analysis.

Citation: Timofeev D.A., Samochadin A.V. Process extraction from educational texts. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2018, Vol. 11, No. 4, Pp. 162–170. DOI: 10.18721/JCSTCS.11412.

Введение

Для анализа и оптимизации процессов, выполняемых работниками в ходе профессиональной деятельности, необходимо иметь возможность описывать эти процессы. В промышленности такие описания (технологические процессы, технологические карты) разрабатываются технологами заранее. В сферах, где преобладает интеллектуальная деятельность, регламентирование последовательности операций применяется существенно реже. Например, в области разработки программного обеспечения конкретные предписания для исполнителя могут создаваться при тестировании программного обеспечения (тест-план). При этом сама возможность строго описать процесс решения задачи приводит к возможности автоматизировать выполнение этого процесса, исключив участие человека. В частности, инженеры по обеспечению качества программного обеспечения в настоящее время преимущественно занимаются не ручным тестированием, а разработкой тестовых планов, которые выполняются с помощью специализированных программных систем, таких как Selenium [1, 2]. Как следствие, большая часть выполняемых процессов явно не описывается, а часто остается скрытой от наблюдателя в силу своего преимущественно ментального характера. В то же время явное описание процессов остается важным не только для их оптимизации, но и с организационной точки зрения: для обучения студентов и начинающих профессиональную деятельность специалистов, адаптации новых сотрудников, обмена опытом, документирования процессов, что в конечном итоге приводит к повышению производительности труда и снижению зависимости организации от знаний конкретных сотрудников. Таким образом, сбор и формализация описаний выполняемых в организации процессов становится важной задачей.

С точки зрения анализа и автоматизации деятельности предпочтительнее описание процессов на формальном языке, таком как BPMN [3] или UML [4]. Моде-

лирование процессов на этом уровне детализации требует привлечения аналитиков, чьей задачей становится не только построение формальной модели процесса, но и в первую очередь извлечение знаний о процессах путем наблюдения, интервьюирования работников и анализа существующих нормативных документов и других видов описаний процессов. Для упрощения моделирования целесообразно автоматизировать эту деятельность, в частности, использовать средства анализа текста на естественном языке для извлечения описаний процессов из документов и записей интервью. При этом необходимо учитывать вид обрабатываемого текста: нормативные документы обычно имеют стандартную структуру и написаны в едином стиле, в то время как описания процессов, сделанные самими работниками, могут существенно отличаться друг от друга и содержать большое количество фрагментов, не относящихся напрямую к выполняемым действиям, а описывающих или обосновывающих выполняемые шаги.

Данная статья посвящена задаче извлечения моделей процессов в области разработки программного обеспечения. Основной особенностью этой области является то, что в ней редко используются нормативные документы, регламентирующие рабочие процессы, в особенности индивидуальные процессы, выполняемые разработчиками: проектирование систем, написание и модификация программного кода, тестирование и отладка. Как следствие, при построении формальных моделей этих процессов необходимо опираться на взаимодействие с самими разработчиками или на описания тех или иных аспектов их деятельности, которые они создают для других целей. Среди таких описаний особое место занимают обучающие материалы: лекции, доклады на профессиональных конференциях, руководства по установке и использованию программ.

Цель создания таких материалов — передача опыта и обучение других людей решению определенных задач или использо-

ванию инструментальных средств. Этот класс документов представляется особенно интересным с точки зрения моделирования индивидуальных процессов: создаваемые разработчиками описания, с одной стороны, отражают часть их повседневной деятельности, обычно не документируемой, а с другой — специально создаются в расчете на восприятие другими людьми, в том числе менее опытными, благодаря чему в явном виде описывают часть очевидных для самого разработчика операций, сведений о контексте и обоснований принятых решений.

Подходы к извлечению моделей процессов из текста

Можно выделить два основных подхода к построению формальных моделей процессов на основе слабоструктурированных данных. Первый подход, обычно называемый извлечением процессов, состоит в построении модели на основе последовательности событий [5]. В качестве исходных данных при этом могут использоваться протоколы взаимодействия пользователей с информационными системами или полученные другими способами записи выполняемых участниками процесса действий. При использовании второго подхода исходными данными служат тексты на естественном языке, которые содержат описания процессов. В этом случае модель процесса восстанавливается путем извлечения из текста конструкций, описывающих участников процесса, выполняемые действия и связи между ними [6, 7]. Результатом является набор правил (в частности, в соответствии со стандартом SBVR [8]), промежуточные текстовые представления процессов [7, 9] или метамодели [10, 11], или непосредственно модели процессов на одном из стандартных языков моделирования [12, 13].

Известные подходы также отличаются друг от друга требованиями, предъявляемыми к исходным данным. Часть решений опирается на контролируемый язык, допускающий только ограниченное подмно-

жество синтаксических конструкций [11, 13]. В работе [10] релевантные фрагменты текстового описания должен выделять пользователь системы. Большая часть систем работает в предположении, что подаваемый на вход документ содержит описание одного процесса и не содержит существенного количества дополнительной информации, которая может быть ошибочно интерпретирована. Частным случаем является работа с исходными документами определенной структуры — например, сценариями использования. В [14] предложен подход, позволяющий выделить релевантные фрагменты текста путем сопоставления извлекаемых из текста слов с набором слов, входящем в концептуальную модель предметной области.

Основной метод извлечения описаний процессов из текста — поиск шаблонов или синтаксических конструкций, соответствующих действующим лицам, действиям и их аргументам, а также маркеров дискурса («и», «или», «затем», «в противном случае» и т. д.) [15, 16]. В работе [12] применяется семантический анализ текста, а для получения информации о семантических отношениях используются базы данных WordNet и FrameNet.

Формирование модели процесса на основе извлеченных из текста конструкций производится с помощью маркеров дискурса, ключевых слов или извлекаемых отношений между предикатом (глаголом) и аргументами (именными группами). При этом обычно не гарантируется целостность и непротиворечивость построенной модели. Основная причина этого — свойственная естественному языку неоднозначность. Анализ такой неоднозначности, возникающей при описании бизнес-процессов, проводится в [17], авторы вводят понятие «поведенческого пространства», включающего все возможные интерпретации извлеченной из текста последовательности событий. Результаты исследования корпуса описаний процессов, представленного в [12], выявили неоднозначно интерпретируемые конструкции в 32 из 47 текстов.

Извлечение моделей процессов из обучающих материалов

Хотя существующие методы извлечения процессов позволяют достичь достаточно высокой для практических целей точности извлечения моделей процессов (77 % на корпусе [12]), они имеют ряд существенных ограничений, в первую очередь касающихся объема и содержания текста. Для обучающих материалов (доклады, видеоуроки) характерно наличие отступлений, в которых автор объясняет принятые решения, демонстрирует существующие возможности или высказывается на отвлеченные темы. Эта особенность существенно меньше выражена в пошаговых руководствах, являющихся частью документации продукта (например, инструкции по установке программы), но и они содержат фрагменты, не описывающие действия участников процесса или состояния системы. Такие фрагменты часто включают те языковые конструкции, на которые опирается алгоритм извлечения процессов, что приводит к включению в модель неверных элементов.

Еще одна важная особенность реальных инструкций и обучающих текстов — одновременное присутствие в тексте описаний нескольких процессов, которые могут быть как связаны друг с другом (процесс и подпроцессы), так и быть независимыми (например, процессы установки приложения в операционных системах Windows и Unix).

Для решения этих проблем предлагается следующий подход.

1. Для анализа используется не только текст, но и разметка документа. Инструкции, руководства и обучающие материалы создаются для использования людьми и поэтому применяют стандартные структурные элементы (разделы, нумерованные и маркированные списки), которые выделяются шрифтом, нумерацией, условными обозначениями или положением на странице. Преимущественным способом распространения документов при этом является их публикация в сети Интернет или передача пользователям в электронной форме, что позволяет учитывать при ана-

лизе документов существующую машинно-читаемую разметку, такую как теги и стили. Особенно важную роль играют следующие характеристики фрагментов текста.

- **Заголовки.** Заголовок раздела часто обозначает выполняемое действие или подпроцесс, которые описываются или комментируются в тексте раздела. Пример: руководство по установке системы управления проектами Bitbucket [18] содержит заголовки «Install Bitbucket Server» и «Set up Bitbucket», в соответствующих им разделах описываются соответствующие процессы установки и настройки приложения. Сами заголовки при этом можно рассматривать как наименования шагов процесса более высокого уровня.

- **Нумерованные списки.** Нумерация действий обычно используется для явного указания порядка их выполнения, что позволяет использовать сам факт нумерации в качестве признака, позволяющего классифицировать фрагмент текста как описание действия, а также установить связь между различными действиями.

- **Маркированные (нумерованные) списки.** Как и нумерованные списки, маркированные фрагменты текста обычно соответствуют выполняемым действиям или вариантам ветвления процесса. В отличие от нумерованных элементов, маркированные элементы могут отражать как упорядочение действий в порядке их перечисления, так и равнозначные альтернативы.

- **Гиперссылки.** Оформление фрагмента текста в качестве гиперссылки может использоваться для указания выполняемого подпроцесса или действия, но чаще просто указывает на наличие дополнительной информации. При анализе текста фрагмент, помеченный как ссылка, должен в ряде случаев обрабатываться особым образом. Например, в предложении «Check that your system meets the minimum requirements shown nearby; Installation requirements has more details» [18] текст ссылки «Installation requirements» является названием раздела, буквальная интерпретация этого словосо-

четания затрудняет как синтаксический анализ (использование формы единственного числа глагола *to have* с существительным во множественном числе), так и семантический анализ текста (является ли фрагмент фразы описанием состояния системы или комментарием, который не следует включать в модель процесса).

- Выделение фрагментов кода, вводимых команд, примеров текста. Учет этого вида разметки необходим для корректной интерпретации соответствующих конструкций, в частности, для их исключения из дальнейшего лингвистического анализа.

2. При анализе документов используется гипотеза локальности: предполагается, что при наличии в документе описаний нескольких процессов каждый из них описывается отдельно. При этом учитывается деление текста на разделы и подразделы: предполагается, что разделы одного уровня вложенности играют одинаковую роль в описании процесса.

3. Анализ текста выполняется по предложениям. Мы предполагаем, что каждое предложение можно отнести к одному из трех классов: «содержит наименование процесса», «содержит описание действия» и «не содержит описания действия».

Наименование процесса может быть представлено в документе как заголовок раздела, обычно достаточно высокого уровня, или как фрагмент предложения, не содержащего описания действия (пример: «But what if you need to install the software on a machine that does NOT have an Internet connection?» [19]).

Описание действия обязательно включает глагольную группу, определяющую выполняемое действие. В зависимости от действия описание может содержать одно или несколько дополнений, описывающих аргументы (параметры) операции, а также именную группу (подлежащее), которая соответствует выполняющему операцию действующему лицу. Если выполнять операцию должен читатель документа, глагол часто ставится в повелительное наклонение, а подлежащее отсутствует. К описа-

ниям действий относятся также предложения, описывающие текущее состояние или реакцию системы (пример: «You should see a blinking indicator in the upper left corner of Live's screen» [20]), т. к. с точки зрения моделирования процесса изменение состояния является действием, выполняемым одним из участников взаимодействия. При интерпретации предложения бывает необходимо изменить действующее лицо, если действие описывается глаголами восприятия (такими, как *see, hear, feel*).

Хотя данную структуру можно считать обязательной для описания действий, она не является уникальной для них. Большая часть комментариев, обоснований и других, не учитываемых при построении модели процесса фрагментов текста, содержат те же самые элементы. Для принятия решения о принадлежности к одному из классов строится классификатор, использующий следующие признаки:

- векторные представления для входящих в предложение слов;

- фрагменты синтаксического дерева фиксированной глубины;

- наличие кореференциальных связей в рамках контекста;

- наличие или отсутствие предопределенных маркеров дискурса;

- характеристика фрагмента текста (наличие или отсутствие разметки);

- класс предложения, стоящего на предшествующем уровне вложенности (например, класс заголовка раздела при распознавании класса предложений в теле раздела).

Распознавание класса каждого предложения должно выполняться в контексте, т. е. с учетом классов других предложений в заданной окрестности. В соответствии с принципом локальности контекст включает предложения на одном уровне структуры документа. Так как решение в этом случае должно согласованно приниматься для каждого из рассматриваемых предложений, для распознавания используется метод МЕММ (Maximum Entropy Markov Model) [21]. Для обучения формируется корпус документов с разметкой классов предложений.

**Начало**

Инициализировать набор описаний процессов пустым множеством.

Создать новое описание процесса без наименования и сделать его текущим описанием.

Разбить текст на блоки в соответствии с вложенностью разделов документа.

Для каждого блока:

Разбить блок на предложения.

Для каждого предложения:

Выделить вектор признаков предложения.

Выполнить классификацию предложения.

Если предложение классифицировано как наименование процесса:

Добавить текущее описание в набор описаний.

Создать новое описание процесса с наименованием, соответствующим предложению, и сделать его текущим описанием.

Иначе, если предложение классифицировано как содержащее описание действия:

Присвоить предложению номер шага.

Извлечь упоминания участников и заменить их на идентификаторы с учетом ранее найденных участников.

Извлечь подлежащее, сказуемое и дополнение и сформировать на их основе описание действия.

Извлечь маркеры условий и ветвлений.

Если найден маркер ветвления:

Определить номер следующего шага, к которому должен быть выполнен переход.

Добавить пятерку <номер шага, участник, действие, условие, следующий шаг> к текущему описанию.

Иначе:

Пропустить предложение.

Если текущее описание не пусто:

Добавить текущее описание в набор описаний.

Для каждого описания в наборе описаний:

Сформировать описание в нотации BPMN на основе упорядоченного набора пятерок.

Конец

Псевдокод алгоритма извлечения описаний процессов
Pseudocode of process description extraction algorithm

Предложения, классифицированные как не описывающие действия, исключаются из дальнейшего рассмотрения. Предложения, классифицированные как наименования процессов, также исключаются из дальнейшего рассмотрения, однако в ходе их обработки производится разбиение документа на фрагменты, соответствующие каждому процессу. Для оставшихся предложений используется алгоритм построения модели процесса, описанный в [7]. Псевдокод алгоритма приведен на рисунке.

В случае, если в одном документе присутствуют несколько описаний процессов,

возможны два подхода: автоматическое построение моделей всех описанных процессов или явный выбор нужного описания пользователем. Хотя явный выбор релевантных описаний пользователем увеличивает трудоемкость процесса, он позволяет уменьшить количество ошибок, связанных с неверным определением границ описаний процессов. В случаях, когда описания процессов оказываются недостаточно разделены на уровне структуры, явное указание границ процесса пользователем оказывается необходимым для корректного извлечения описаний процессов.

Заключение

Задача извлечения моделей процессов из текстовых описаний играет важную роль для автоматизации работы аналитика при изучении и оптимизации процессов в организации, особенно в тех сферах деятельности, где преобладает интеллектуальный труд, но реализуемые сотрудниками процессы не регламентируются и редко описываются. Характерный пример такой сферы — разработка программного обеспечения.

Известные подходы к извлечению моделей процессов демонстрируют достаточную для практического использования точность, но рассчитаны на работу с текстами, подчиняющимися строгим ограничениям по размеру, структуре и подмножеству используемого языка. В данной статье сделана попытка обобщить методы извлечения моделей на случай текстов, которые могут содержать описания более чем одного процесса (при условии выполнения гипотезы локальности), а также содержать фрагменты, не описывающие никакие аспекты процесса (пояснения, обоснования, примеры). Для этого предлагается использовать не только стандартные для задач анализа текста признаки, такие как леммы, части речи, синтаксические зависимости, но и признаки, основанные на структуре и разметке самого документа.

Предложенный подход предназначен в первую очередь для анализа документов со структурной разметкой (HTML, PDF, Microsoft Word, LaTeX). Такие документы широко представлены в сети Интернет, а

также составляют значительную часть электронного документооборота организаций. Возможно также применение этого подхода для анализа текстов без явной структурной разметки за счет использования знаков препинания, отступов и символов перевода строки в качестве признаков. Тем не менее, существенное использование предположения локальности описания приводит к трудностям при анализе документов, в которых присутствуют описания нескольких процессов. В этом случае рекомендуется привлечение пользователя для указания границ процесса и выбора релевантных процессов.

В дальнейшем предполагается развивать предложенный подход в двух основных направлениях:

расширение набора признаков для классификации предложений, в первую очередь использование маркеров дискурса для идентификации фрагментов, похожих на описания действий, но не относящихся к извлекаемому процессу (в частности, действий, приводимых в качестве примеров или нереализованных альтернатив);

использование тезауруса и онтологии предметной области для принятия решений о присутствии в тексте описаний процессов, соответствующих этой предметной области, и более точного определения границ описания процесса.

Работа подготовлена в ходе реализации проекта в рамках Постановления Правительства РФ от 09.04.2010 № 218 при финансовой поддержке Министерства образования и науки РФ. Договор № 03.G25.31.0247 от 28.04.2017.

СПИСОК ЛИТЕРАТУРЫ

1. Holmes A., Kellogg M. Automating functional tests using selenium // Agile Conference 2006. IEEE, 2006.
2. SeleniumHQ Browser Automation // URL: <https://www.seleniumhq.org/> (Дата обращения: 10.11.2018).
3. Object Management Group. Business Process Model And Notation, version 2.0.2. — 2014 // URL: <https://www.omg.org/spec/BPMN/2.0.2/> (Дата обращения: 10.11.2018).
4. Object Management Group. Unified Modeling Language, version 2.5.1. — 2017 // URL: <https://www.omg.org/spec/UML/> (Дата обращения: 10.11.2018).
5. van der Aalst W.M.P. Process mining: Discovery, conformance and enhancement of business processes. Berlin: Springer-Verlag, 2011.
6. Riefer M., Ternis S.F., Thaler T. Mining process models from natural language text: A state-of-the-art analysis // Multikonferenz Wirtschaftsinformatik. March 2016. Pp. 9–11.

7. Honkisz K., Kluza K., Wiśniewski P. A concept for generating business process models from natural language description // *Internat. Conf. on Knowledge Science, Engineering and Management*. Springer, Cham, 2018. Pp. 91–103.
8. Object Management Group. Semantics of Business Vocabulary and Rules, version 1.4. – 2017 // URL: <https://www.omg.org/spec/SBVR/1.4/> (Дата обращения: 10.11.2018).
9. Mogos A.H., Urzica A. TN4PM: A textual notation for process modelling // *Intelligent Distributed Computing III*. Springer, Berlin, Heidelberg, 2009. Pp. 263–268.
10. Kop C., et al. Tool Supported Extraction of Behavior Models // *ISTA*. 2005. Vol. 63. Pp. 114–123.
11. Yue T., Briand L.C., Labiche Y. Automatically deriving a UML analysis model from a use case model. Carleton University, 2010.
12. Friedrich F., Mendling J., Puhmann F. Process model generation from natural language text // *Internat. Conf. on Advanced Information Systems Engineering*. Springer, Berlin, Heidelberg, 2011. Pp. 482–496.
13. Caporale T. A tool for natural language oriented business process modeling // *ZEUS*. 2016. Pp. 49–52.
14. Annervaz K.M., et al. Natural language requirements quality analysis based on business domain models // *Proc. of the 28th IEEE/ACM Internat. Conf. on Automated Software Engineering*. IEEE Press, 2013. Pp. 676–681.
15. Ghose A., Koliadis G., Chueng A. Process discovery from model and text artefacts // *Services, 2007 IEEE Congress on*. IEEE, 2007. Pp. 167–174.
16. Gonçalves J.C.A.R., Santoro F.M., Bairo F.A. Business process mining from group stories // *Proc. of the 2009 13th Internat. Conf. on Computer Supported Cooperative Work in Design*, 2009.
17. van Der A.H., Leopold H., Reijers H.A. Dealing with behavioral ambiguity in textual process descriptions // *Internat. Conf. on Business Process Management*. Springer, Cham, 2016. Pp. 271–288.
18. Atlassian. Install Bitbucket Server on Linux // URL: <https://confluence.atlassian.com/bi-tbucketserver/install-bitbucket-server-on-linux-868976991.html> (Дата обращения: 10.11.2018).
19. Nuts About Nets FAQ. How to Register Software Without An Internet Connection // URL: <http://nutsaboutnets.com/faqs/register-without-internet/> (Дата обращения: 21.11.2018).
20. HomeRecording.com Forum. SeaGtGruff post: Help – basic DAW setup help // URL: <https://homerecording.com/bbs/general-discussions/newbies/help-basic-daw-setup-help-384095/3/#post4359360> (Дата обращения: 21.11.2018).
21. McCallum A., Freitag D., Pereira F.C.N. Maximum entropy Markov models for information extraction and segmentation // *ICML*. 2000. Vol. 17. No. 2000. Pp. 591–598.

Статья поступила в редакцию 22.11.2018.

REFERENCES

1. Holmes A., Kellogg M. Automating functional tests using selenium. *Agile Conference*, IEEE, 2006.
2. SeleniumHQ Browser Automation. Available: <https://www.seleniumhq.org/> (Accessed: 10.11.2018).
3. Object Management Group. Business Process Model And Notation, version 2.0.2. – 2014. Available: <https://www.omg.org/spec/BPMN/2.0.2/> (Accessed: 10.11.2018).
4. Object Management Group. Unified Modeling Language, version 2.5.1. – 2017. Available: <https://www.omg.org/spec/UML> (Accessed: 10.11.2018).
5. van der Aalst W.M.P. *Process mining: Discovery, conformance and enhancement of business processes*. Berlin: Springer-Verlag, 2011.
6. Riefer M., Ternis S.F., Thaler T. Mining process models from natural language text: A state-of-the-art analysis. *Multikonferenz Wirtschaftsinformatik*, March 2016, Pp. 9–11.
7. Honkisz K., Kluza K., Wiśniewski P. A concept for generating business process models from natural language description. *International Conference on Knowledge Science, Engineering and Management*. Springer, Cham, 2018, Pp. 91–103.
8. Object Management Group. Semantics of Business Vocabulary and Rules, version 1.4. – 2017. Available: <https://www.omg.org/spec/SBVR/1.4/> (Accessed: 10.11.2018).
9. Mogos A.H., Urzica A. TN4PM: A textual notation for process modeling. *Intelligent Distributed Computing III*. Springer, Berlin, Heidelberg, 2009, Pp. 263–268.
10. Kop C., et al. Tool supported extraction of behavior models. *ISTA*, 2005, Vol. 63, Pp. 114–123.
11. Yue T., Briand L.C., Labiche Y. *Automatically deriving a UML analysis model from a use case model*. Carleton University, 2010.
12. Friedrich F., Mendling J., Puhmann F. Process model generation from natural language text. *International Conference on Advanced Infor-*

mation Systems Engineering. Springer, Berlin, Heidelberg, 2011, Pp. 482–496.

13. **Caporale T.** A tool for natural language oriented business process modeling. *ZEUS*, 2016, Pp. 49–52.

14. **Annervaz K.M., et al.** Natural language requirements quality analysis based on business domain models. *Proceedings of the 28th IEEE/ACM International Conference on Automated Software Engineering*. IEEE Press, 2013, Pp. 676–681.

15. **Ghose A., Koliadis G., Chueng A.** Process discovery from model and text artefacts. *Services, 2007 IEEE Congress on*. IEEE, 2007, Pp. 167–174.

16. **Gonçalves J.C.A.R., Santoro F.M., Bairo F.A.** Business Process Mining from Group Stories. *Proceedings of the 2009 13th International Conference on Computer Supported Cooperative Work in Design*, 2009.

17. **van Der A.H., Leopold H., Reijers H.A.** Dealing with behavioral ambiguity in textual pro-

cess descriptions. *International Conference on Business Process Management*. Springer, Cham, 2016, Pp. 271–288.

18. Atlassian. Install Bitbucket Server on Linux. Available: <https://confluence.atlassian.com/bitbucketserver/install-bitbucket-server-on-linux-868976991.html> (Accessed: 10.11.2018).

19. Nuts About Nets FAQ. How to Register Software Without An Internet Connection. Available: <http://nutsaboutnets.com/faqs/register-without-internet/> (Accessed: 21.11.2018).

20. HomeRecording.com Forum. SeaGtGruff post: Help – basic DAW setup help. Available: <https://homerecording.com/bbs/general-discussions/newbies/help-basic-daw-setup-help-384095/3/#post4359360> (Accessed: 21.11.2018).

21. **McCallum A., Freitag D., Pereira F.C.N.** Maximum entropy Markov models for information extraction and segmentation. *ICML*, 2000, Vol. 17, No. 2000, Pp. 591–598.

Received 22.11.2018.

СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

ТИМОФЕЕВ Дмитрий Андреевич
TIMOFEEV Dmitrii A.
 E-mail: dtim@dcn.icc.spbstu.ru

САМОЧАДИН Александр Викторович
SAMOCHADIN Alexander V.
 E-mail: samochadin@gmail.com



2nd Workshop on Information Management in Human-Centric Systems (IMHCS'19)

An event of the 18th IEEE International Conference on Ubiquitous Computing and Communications, Shenyang, Oct 21 – 23, 2019

<http://iucc2019.sau.edu.cn/>

Organizers

Workshop chairs: **Evgeny Pyshkin**, *University of Aizu, Japan*, and **Natalia Bogach**, *Peter the Great St. Petersburg Polytechnic University, Russia*

Scope and Objectives

The 2nd Workshop on Information Management in Human-Centric Systems (IMHCS'19) is focused on discussing current trends in information acquisition, representation and processing in human-centric systems and applications. We particularly welcome contributions from researchers and practitioners presenting new results in the big area of information and data management with special attention paid to human-centric approaches. The papers presented in the 1st workshop were included to the proceedings the 3rd IEEE International Conference on Cybernetics (CYBCONF 2017) published by IEEE Computer Society Press and indexed in Scopus and CPCI (ISI conferences and part of Web of Science).

We expect to organize a multi-disciplinary discussion of high quality contributions related to the topics, which include but not limited to:

- Information Management
- Data and Subject Domain Modeling
- Innovation and Digital Transformation
- Information Retrieval and Data Science
- Human-Centric Design and Architectures
- Data Management and Artificial Intelligence
- Computer Assisted Learning and Agile Education
- Intelligent Internet-of-Things and AI for IoT
- Intelligent User Interfaces and Usability
- Ambient User Experience
- Data Representation and Visualization
- Collaborative Environments
- Recommendation Systems
- Knowledge Engineering
- Software Development and Engineering
- Social Networking

Important Dates

Paper submission due: **May 15, 2019**

Acceptance notification: **July 31, 2019**

Conference dates: **October 21 – 23, 2019**

Program Committee

Ivan Blekanov, *St. Petersburg State University, Russia*

Andrey Kuznetsov, *T-Systems Rus, Russia*

Natalia Bogach, *Peter the Great St. Petersburg Polytechnic University, Russia*

Maxim Mozgovoy, *University of Aizu, Japan*

Alfredo Capozucca, *University of Luxembourg, Luxembourg*

Evgeny Pyshkin, *University of Aizu, Japan*

Leonid Chechurin, *Lappeenranta University of Technology, Finland*

Sergey Redko, *St. Petersburg Polytechnic University*

Vlatko Davidovski, *Cognizant Business Consulting, Switzerland*

Juan Miguel Rubio, *Universitat Politècnica de València, Spain*

Alina Gutnova, *North Ossetian State University, Russia*

Vadim Tsesko, *Odnoklassniki, Russia*

Vitaly Klyuev, *University of Aizu, Japan*

Alexander Vazhenin, *University of Aizu, Japan*

Sergey Krutolevich, *Belarusian Russian University, Belarus*

Takako Yasuta, *University of Aizu, Japan*

Rüdiger Lunde, *Ulm University of Applied Sciences, Germany*

Liang Zhao, *Shenyang Aerospace University, China*

Submission and Publication

Authors are invited to submit original previously unpublished research papers written in English, of up to 6 pages (or 8 pages with over length charge) including figures and references using IEEE Computer Society Proceedings Manuscripts style (two columns, single-spaced, 10 fonts). You can find the submission templates and author instructions on the IUCC-2019 conference webpage. At least one of the authors of every accepted paper must be registered for the conference, pay the conference fee and present the accepted paper.

Presented papers will be included into the IEEE conference proceedings published by the IEEE Computer Society Press and submitted for indexing in IEEE Xplore and Engineering Index (EI).

Paper submission system: <https://easychair.org/conferences/?conf=imhcs19>



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 11, № 4, 2018

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. Корнуковой*

Перевод на английский язык *А.С. Колгатина*

Лицензия ЛР № 020593 от 07.08.97

Подписано в печать 28.12.2018. Формат 60×84 1/8. Бум. тип. № 1."
Печать офсетная. Усл. печ. л. 20,46. Уч.-изд. л. 20,46. 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