



# НАУЧНО-ТЕХНИЧЕСКИЕ ВЕДОМОСТИ

САНКТ-ПЕТЕРБУРГСКОГО ГОСУДАРСТВЕННОГО  
ПОЛИТЕХНИЧЕСКОГО УНИВЕРСИТЕТА

---

---

Информатика. Телекоммуникации.  
Управление

---

---

**Том 10, № 4**  
**2017**

# 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:

Бабкин А.В., д-р экон. наук, профессор, Санкт-Петербургский политехнический университет Петра Великого, Россия;

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

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

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

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;

Трифонов П.В., канд. 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;

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 1995 года издается под научно-методическим руководством Российской академии наук. С 2008 года выпускается в составе сериального периодического издания «Научно-технические ведомости СПбГПУ» ISSN 1994-2354.*

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 Российской 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, 2017



**ST. PETERSBURG STATE  
POLYTECHNICAL UNIVERSITY  
JOURNAL**

---

---

Computer Science.  
Telecommunications and Control Systems

---

---

**Vol. 10, No. 4  
2017**

# 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:

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

Assoc. Prof. Dr. *Vladimir M. Itsyson*, 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;

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

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

Assoc. 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.

*The journal is published under scientific and methodical guidance of the Russian Academy of Sciences since 1995. The journal is published since 2008 as part of the periodical edition «Nauchno-tekhnicheskie vedomosti SPbGPU» (ISSN 1994-2354).*

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

**Лаптев В.В., Орлов П.А., Драгунова О.В.** Визуализация динамических структур данных с помощью потоковых диаграмм в веб-аналитике ..... 7

**Самочадин А.В., Лебедев Е.В., Пономарева Е.В., Мищенко К.В.** Исследование технологии визуализации данных программно-технологической системы интерактивного стратегирования и бизнес-анализа ..... 17

## Телекоммуникационные системы и компьютерные сети

**Омётов А.Я., Беззатеев С.В., Кучерявый Е.А.** Протокол аутентификации для передачи прав пользования электронными устройствами ..... 29

## Устройства и системы передачи, приема и обработки сигналов

**Никитин А.Б., Хабитуева Е.И.** Особенности разработки сверхширокополосных перестраиваемых генераторов СВЧ диапазона в гибридном исполнении ..... 41

## Программное обеспечение вычислительных, телекоммуникационных и управляющих систем

**Подкопаев А.В., Лахав О., Вафеедис В.** О корректности компиляции подмножества обещающей модели памяти в аксиоматическую модель ARMv8.3 ..... 51

**Тимофеев Д.А., Самочадин А.В.** Описание процессов разработки программ с помощью скрытых марковских моделей ..... 70

## Информационные, управляющие и измерительные системы

**Курочкин Л.М., Курочкин М.А., Попов С.Г., Попов М.В.** Результаты экспериментальных исследований точности позиционирования при использовании различных систем спутниковой навигации..... 79

## Интеллектуальные системы и технологии

**Мархотин А.А., Кривошейкин А.В., Осипенко И.Н., Рогозинский Г.Г.** Система сонификации в задаче нахождения пути и обхода препятствий ..... 89

**Онуфриев В.А.** Управление группой автономных роботов с использованием полярных координат ... 97

**Попов С.Г., Коненков А.В., Самочадина А.В.** Редактор визуального проектирования схем и процедур заполнения OLAP-кубов многомерных баз данных ..... 107

## Contents

### Information Technology

|                                                                                                                                                                                                            |    |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| <b>Laptev V.V., Orlov P.A., Dragunova O.V.</b> <i>Visualization of Dynamic Data Structures with Flow Charts in Web Analytics</i> .....                                                                     | 7  |
| <b>Samochadin A.V., Lebedev E.V., Ponomareva E.V., Mishchenkova K.V.</b> <i>The Study of the Data Visualization Technology of a Software System for Interactive Strategies and Business Analysis</i> ..... | 17 |

### Telecommunication Systems and Computer Networks

|                                                                                                                                                           |    |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| <b>Ometov A.Ya., Bezzateev S.V., Koucheryavy Y.A.</b> <i>Authenticating Electronic Devices for Temporary Use: Access Rights Management Protocol</i> ..... | 29 |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------|----|

### Circuits and Systems for Receiving, Transmitting, and Signal Processing

|                                                                                                                    |    |
|--------------------------------------------------------------------------------------------------------------------|----|
| <b>Nikitin A.B., Khabitueva E.I.</b> <i>The Features of Design of an Ultra-Wideband Microwave Hybrid VCO</i> ..... | 41 |
|--------------------------------------------------------------------------------------------------------------------|----|

### Computer Software, Telecommunications, and Control Systems

|                                                                                                                                                            |    |
|------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| <b>Podkopaev A.V., Lahav O., Vafeiadis V.</b> <i>On Compilation Correctness for a Subset of a Promising Memory Model to the ARMv8.3 Memory Model</i> ..... | 51 |
| <b>Timofeev D.A., Samochadin A.V.</b> <i>Modeling of Software Development Processes with Hidden Markov Models</i> .....                                    | 70 |

### Information, Control and Measurement Systems

|                                                                                                                                                                               |    |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| <b>Kurochkin L.M., Kurochkin M.A., Popov S.G., Popov M.V.</b> <i>Results of Experimental Studies of Positioning Accuracy Using Various Satellite Navigation Systems</i> ..... | 79 |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|

### Intelligent Systems and Technologies

|                                                                                                                                                                     |     |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| <b>Markhotin A.A., Krivosheykin A.V., Osipenko I.N., Rogozinsky G.G.</b> <i>Sonification System in Finding Route and Detour of Obstacles</i> .....                  | 89  |
| <b>Onufriev V.A.</b> <i>Autonomous Robots Group Control Using Polar Coordinates</i> .....                                                                           | 97  |
| <b>Popov S.G., Konenkov A.K., Samochadina T.N.</b> <i>Visual Design Editor for Schemes and Procedures of Filling OLAP Cubes in Multidimensional Databases</i> ..... | 107 |

DOI: 10.18721/JCSTCS.10401

УДК 004.92:311.218

## ВИЗУАЛИЗАЦИЯ ДИНАМИЧЕСКИХ СТРУКТУР ДАННЫХ С ПОМОЩЬЮ ПОТОКОВЫХ ДИАГРАММ В ВЕБ-АНАЛИТИКЕ

*В.В. Лаптев, П.А. Орлов, О.В. Драгунова*

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

Потоковые диаграммы относятся к инструменту графического представления структуры данных, которая имеет динамический семантический контекст. Он основан на идее деления целого на составные части и их визуальное разделение между ними. Структуры данных часто играют роль индикаторов в управлении и веб-аналитике. В статье описана форма потоковых диаграмм в веб-аналитике, исследованная на основе методов ай-трекинга. Получено подтверждение эффективности использования рекомендаций по построению потоковых диаграмм. Выявлены проблемы формы потоковых диаграмм в системах Google Analytics и предложено решение в виде концепта сервиса по визуализации статистической информации.

**Ключевые слова:** инфографика; визуализация данных; потоковые диаграммы; диаграммы сэнкей; веб-аналитика; гугл аналитикс; ай-трекинг.

**Ссылка при цитировании:** Лаптев В.В., Орлов П.А., Драгунова О.В. Визуализация динамических структур данных с помощью потоковых диаграмм в веб-аналитике // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2017. Т. 10. № 4. С. 7–16. DOI: 10.18721/JCSTCS.10401

## VISUALIZATION OF DYNAMIC DATA STRUCTURES WITH FLOW CHARTS IN WEB ANALYTICS

*V.V. Laptev, P.A. Orlov, O.V. Dragunova*

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

Flow visualization is a tool of graphic presentation of data structures, which has a dynamic semantic context. It is based on the idea of dividing the whole into its component parts and visual separation between them. Data structures are frequent indicators in management and web analytics. This article focuses on the shape of flow charts in web analytics, which was investigated by the eye-tracking methods. We have confirmed the effectiveness of using recommendations for building flow charts, identified the problems of the shape of flow charts in Google Analytics, and offered a solution in the form of a concept for visualization of statistical information.

**Keywords:** infographics; data visualization; flow charts; sankey diagram; web analytics; google analytics; eye-tracking.

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

Управление информацией подразумевает построение структур данных различного рода и сценариев их восприятия, что является важной прикладной задачей. Вопросы визуальной интерпретации актуальны в связи с вариативностью формы представления. Существуют проблемы выбора инструмента визуализации числовых данных — соответствующего типа диаграммы — определение их оптимальной формы и расположения относительно других элементов композиции. Так, один и тот же пример структуры данных может быть представлен в различном виде: брусковых диаграммах, секторных диаграммах, древовидных картах или потоковых диаграммах. Правильный выбор формы основывается на учете контекста, семантических связей между числовым массивом и его графическим образом, удобочитаемости графика, его простоте и ясности восприятия. Это можно экстраполировать на классификационные или иерархические схемы с количественным анализом.

Потоковая диаграмма относится к инструменту графического представления структуры данных, который имеет явно выраженный семантический контекст динамики процесса. Основанные на идее разделения целого на его составные части и их визуального разделения между собой, потоковые диаграммы процессов часто именуют «диаграммами Сэнкей» (Sankey diagrams), по имени Мэтью Сэнкея (Matthew Sankey), в 1898 г. применившего на практике этот способ графического представления информации для анализа эффективности использования энергии парового двигателя.

При визуализации изменяемой в динамике структуры целого потоковые диаграммы расставляют визуальные акценты на передаче данных между различными узлами системы. Они выявляют доминирующие потоки, полезны в поиске и анализе недостатков, показывают балансы показателей процесса. Динамические структуры данных часто используются в качестве индикаторов

в управлении и аналитике, а также для иллюстрации энергетических и материальных балансов сложных систем. Поэтому блок-схемы, состоящие из потоковых диаграмм, широко распространены в системах управления промышленным производством [1], в энергетическом секторе [2], на транспорте [3], в аналитических исследованиях [4], в вопросах экологии [5], в бизнес-проектах и веб-аналитике [6].

Веб-аналитика, как правило, использует большие наборы данных с многочисленными связями. Эти наборы данных нередко представляют собой сложные структуры, состоящие из большого количества мест выборки и нескольких иерархических уровней [7], что требует простой и ясной визуализации. Потоковые диаграммы позволяют наглядно показать распределение внимания посетителей сайта, оценить их количество в отдельных узлах, выявить пути перемещения. Потоковые диаграммы являются стандартным решением визуализации путей пользователей в Google Analytics, а также используются для визуализации карты поведения пользователей.

В данной статье сравниваются потоковые диаграммы Google Analytics и разработанного нами прототипа, исследованные на основе методов ай-трекинга. Это является продолжением процесса систематического анализа различных аспектов проектирования продуктов потоковой визуализации для веб-аналитики [8, 9]. В основе наших исследований лежит изучение вопросов восприятия информации, что также поможет разработать рекомендации по улучшению способов потоковой визуализации информации в других областях аналитики и управления.

### Анализ проблемы

Существуют хорошо известные рекомендации по проектированию потоковых диаграмм. Это поддержание стабильной толщины непрерывного потока, едино-

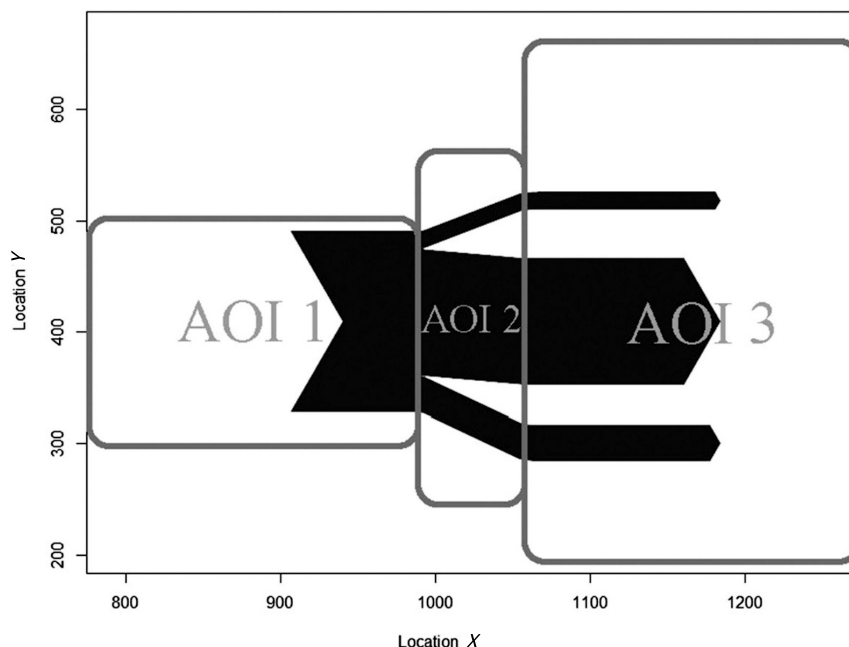


Рис. 1. Зоны интереса пользователя (АОИ) в потоковой диаграмме, полученные на основании кластерного анализа (данные по осям  $X$  и  $Y$  показывают местоположение диаграммы в пикселах)

Fig. 1. Areas of Interest (AOI) in the flow chart obtained from cluster analysis (data on the  $X$ - and  $Y$ -axes show the position of the chart in pixels)

образное формирование корней потоков, пространственная компоновка узлов [10], целостность композиции при взаимодействии потока с внешними ориентирами [11]. Используя метод ай-трекинга и кластерный анализ визуального восприятия потоковых диаграмм, получены рекомендации по соблюдению трех зон интереса (АОИ). Входная или начальная, объединяющая группу зрительных фиксаций в области исходного потока (АОИ 1), выходная, соответствующая интересу зрителя по отношению к финишным потокам (АОИ 3), и средняя центральная, в которой происходит разветвление блок-схемы (АОИ 2) — эти зоны определены для потоковых диаграмм различной степени сложности [8]. Центральный кластер зрительных фиксаций АОИ 2 (рис. 1) показывает, что зритель сравнивает части между собой не только в выходной зоне, но и в динамической части диаграммы, когда происходит процесс разделения на части.

Потоковая диаграмма может быть интерпретирована как динамическая модифи-

кация гистограммы и представлена как движущаяся часть стека — отдельных блоков бруска. Поэтому мы обратились к исследованию формы стековых гистограмм, представляющих структуру данных. Доказано, что изменение базовой формы столбиковой диаграммы оказывает влияние на точность абсолютных и относительных суждений в графиках [12]. Даже небольшие изменения, такие как округление или заострение вершины прямоугольника, приводят к увеличению количества ошибок.

По аналогии с брусковыми диаграммами для потоковых диаграмм может быть рекомендована необходимость формирования четкой картины, т. к. изменение основной формы потока влияет на визуальное восприятие динамики структуры данных. Дальнейшее исследование формы потоков [9] выявило их индифферентность к разделению его на составные части в корневой начальной зоне интереса (АОИ 1), что позволяет безболезненно использовать тоновое выделение отдельных

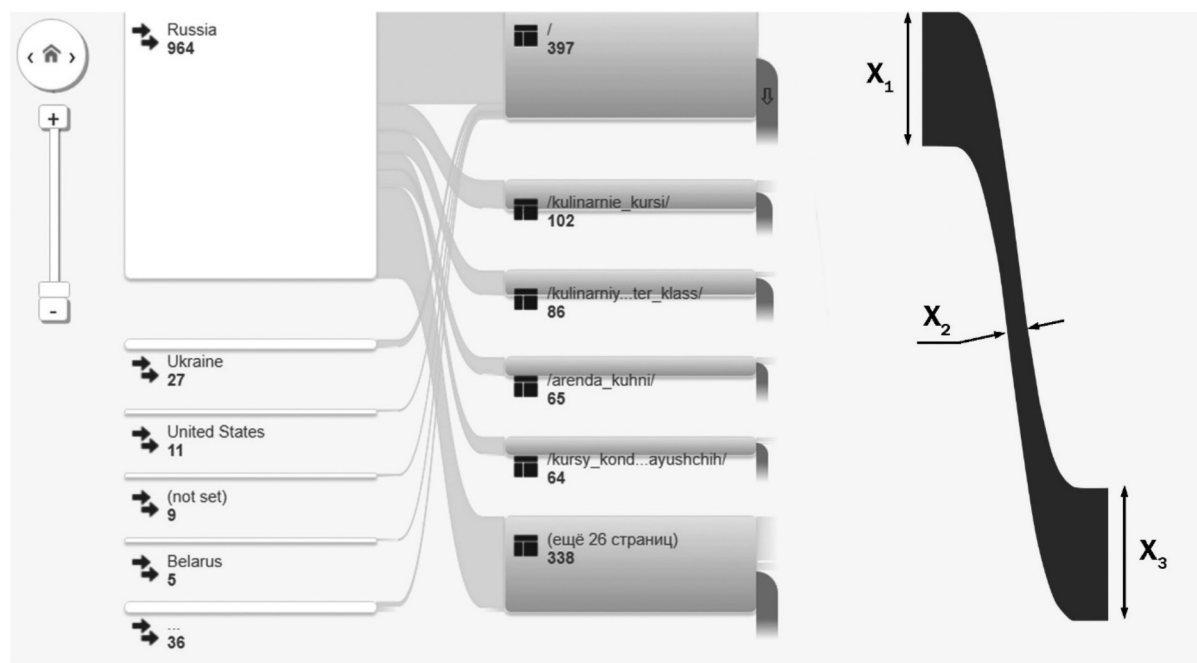


Рис. 2. Потокная диаграмма путей пользователей Google Analytics  
(фрагмент показывает изменение толщины потока в средней части в несколько раз)

Fig. 2. The flow chart of the ways of users of Google Analytics  
(the fragment shows the change in the thickness of the flow in the middle part several times)

потоков в статичных схемах или в интерактивных версиях потоковых диаграмм. Показано, что более критична для визуального восприятия деформация формы потока в средней части: возникает изменение поведенческих реакций, связанное с потерей семантических связей между процессом и способом визуализации [9].

Предварительный анализ потоковых диаграмм путей пользователей в Google Analytics продемонстрировал, что одним из недостатков визуализации является нестабильная по толщине форма потоков. В этом случае визуальное сравнение долей структуры данных между собой и с целым будет представлять затруднения в центральной (средней) части потока (рис. 2), т. е. будет снижаться эффективность визуализации данных. По мнению Ж. Бертина, под эффективностью восприятия информации понимается точность и скорость решения задачи [13, с.139].

Еще одним недостатком визуализации путей пользователей в Google Analytics яв-

ляется отсутствие графических и колористических приемов выделения данных в потоковой диаграмме, что приводит к необходимости сравнения потоков, используя только числовые показатели. Использование ширины потока для формирования высоты прямоугольника блока-узла нельзя признать удачным решением из-за возможного ограничения площади для представления экспликационных данных.

Одним из решений повышения эффективности визуального восприятия потоковой диаграммы может стать цветовое кодирование информации, например, как в гибридных диаграммах Сэнкей [14]. Однако использование только лишь колористического инструмента может привести к излишне пестрой картине, ухудшающей визуальное восприятие диаграммы [15, с. 72]. Поэтому сформулируем предположение-гипотезу, что эффективность визуализации потоковых диаграмм в Google Analytics можно улучшить, изменив композицию потоковой диаграммы, форму потоков и узлов.



### Постановка эксперимента

Для того чтобы подтвердить или опровергнуть утверждение о том, что текущая визуализация диаграммы путей пользователей Google Analytics может быть улучшена с целью повышения точности и скорости ее чтения, необходимо провести экспериментальное сравнение двух диаграмм — исходного графика и диаграммы, разработанной с учетом рекомендаций по построению потоковых диаграмм.

Визуальный прототип диаграммы был подготовлен на основании статистических данных, экспортированных из соответствующей диаграммы Google Analytics. Основные отличия разработанного прототипа диаграммы от оригинального решения:

- наличие трех выделенных зон потока: начальной, центральной зоны перехода и завершающей выходной зоны;
- отсутствие деформации толщины потока во всех зонах;
- блоки-узлы, обозначающие страницы, сделаны одного размера;
- вместо URL-адреса в описании узла указано название страницы;
- места пересечения потоков выполнены с эффектом наложения;
- слева и справа на блоке добавлено графическое представление суммарного потока в виде темного прямоугольника;
- добавлено числовое отображение исходящего потока и ушедших пользователей.

Так как оригинальная диаграмма путей пользователей предполагает возможность интерактивного взаимодействия (при нажатии на определенный узел диаграммы пользователь может выделить весь поток проходящего через узел трафика), разработан интерактивный вид альтернативной диаграммы, демонстрирующий численные показатели определенных потоков пользователей (рис. 3). Подготовка стимульного материала представляла собой визуализацию диаграммы в различных состояниях. Под различными состояниями в данном случае понимаются варианты отображения различных выделенных потоков диаграммы. Для проведения эксперимента был

подготовлен рабочий прототип на сервисе InVision, с которым взаимодействовал испытуемый.

Было проведено ай-трекинговое исследование, т. к. помимо точности и скорости важно было посмотреть скрытые характеристики, такие как субъективную сложность. Субъективную сложность мы оцениваем опосредованно, по длительности фиксации взора. Также, зная положение фиксации на стимуле, можно говорить об оценке визуального внимания человека. Таким образом, в анализ было включено и количество фиксаций при решении задач. Для определения субъективной общей нагрузки использован NASA TLX тест нагрузки.

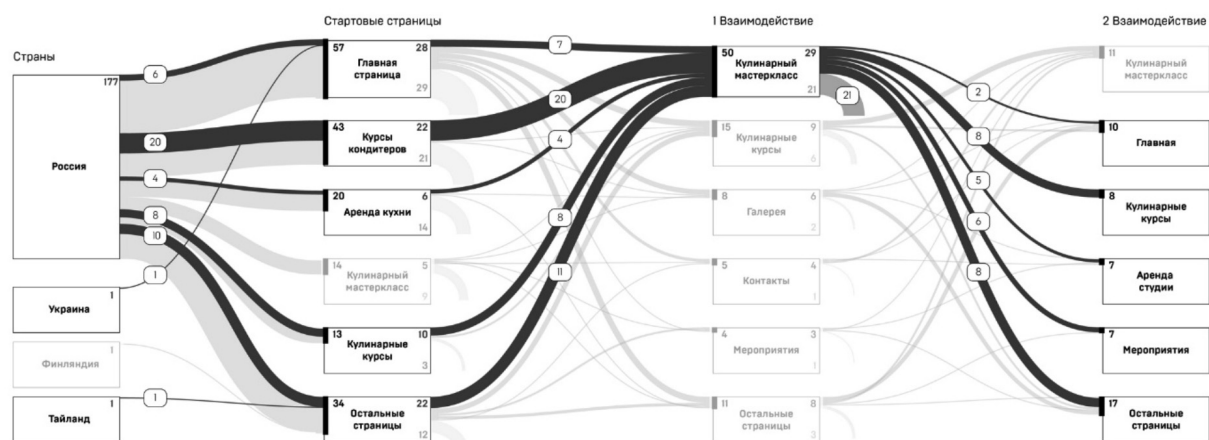
В эксперименте участвовало 22 человека (средний возраст 19 лет, 8 мужчин и 14 женщин). Испытуемые были разбиты на две группы по 11 человек случайным образом. Первая группа взаимодействовала с диаграммой Google Analytics, вторая группа — с разработанным альтернативным вариантом диаграммы. Испытуемые являлись опытными пользователями компьютера, при этом каждый из них взаимодействовал с подобным интерфейсом впервые.

Стимулы предъявлялись испытуемым с помощью специально разработанного программного обеспечения. Это ПО включало в себя десять вопросов с четырьмя вариантами ответа. Вопросы с первого по пятый не требовали интерактивного взаимодействия с диаграммой: для ответа участнику было достаточно рассмотреть диаграмму. Вопросы с шестого по десятый предполагали взаимодействие участника с диаграммой: например, требовалось выделить конкретный узел диаграммы и сравнить два потока. Все вопросы показывались последовательно всем испытуемым (не в случайном порядке). После ответов на первые пять вопросов тест ставился на паузу, и производился короткий инструктаж по интерактивным возможностям диаграммы, которые необходимо знать испытуемому для ответа на вопросы с шестого по десятый.

Программа осуществляла запись таких параметров, как фамилия, имя, возраст, правильность ответа и время решения задачи. Для дальнейшей обработки данных



а)



б)

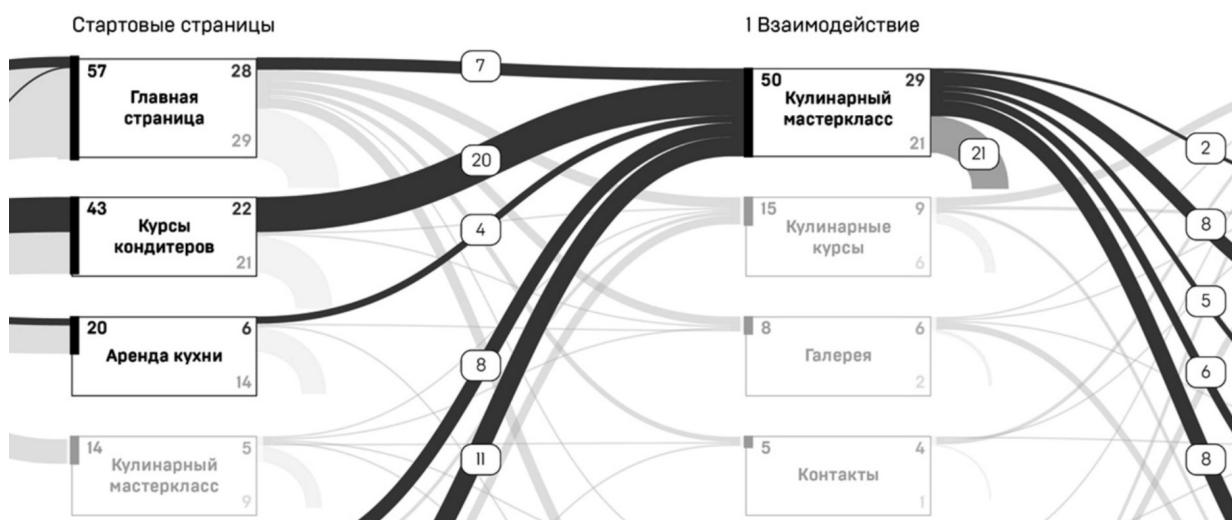


Рис. 3. Прототип диаграммы путей пользователей (интерактивный вариант):

а – общий вид; б – фрагмент диаграммы

Fig. 3. The prototype of the chart of the ways of users (interactive version):

а – general view; б – a fragment of the chart

учитывались фиксации только стимульной зоны. Испытуемый производил выбор с помощью компьютерного манипулятора мышью. Подразумевалось, что испытуемый являлся опытным пользователем компьютера, и задание с точки зрения человеко-компьютерного взаимодействия не было для него новым.

В процессе решения задач окуломоторная активность регистрировалась в лаборатории на стационарном ай-трекере SMI RED250 (с частотой 250 Гц). Фиксации определялись встроенным алгоритмом SMI.

Кроме того, автоматически фиксировалась правильность ответа и длительность решения задачи. Ай-трекер располагался под монитором стимульного компьютера, в одной плоскости с ним. Обнаружение фиксации выполнялось с помощью программного обеспечения SMI iViewX, и информация о начале и конце фиксации отправлялась на инструмент обратной связи с глазами. Глаза испытуемого находились на расстоянии 70 см от экрана монитора BENQ XL2411 (диагональ 24", размеры экрана 53×30 см, 1920×1080 пикселей, частота 144 Гц). Голо-

ва испытуемого фиксировалась на офтальмологической подставке.

Предварительная обработка испытательного и калибровочного оборудования соответствовала методологии [16] и техническим условиям, предъявляемым к экспериментам с использованием айтрекингового оборудования в Лаборатории человеко-компьютерного взаимодействия Санкт-Петербургского политехнического университета Петра Великого [17]. Для определения субъективной общей нагрузки использовался NASA TLX тест нагрузки [18] в виде субъективных отчетов. Испытуемый давал общую оценку работы.

### Результаты эксперимента

Оценивалось влияние типа диаграмм на эффективность решения задачи и на поведение при решении задачи. Под эффективностью подразумевается количество правильных ответов и скорость решения задачи. Оценка влияния факторов производилась в программе R-Studio, предназначенной для статистической обработки данных и работы с графикой.

*Общее время решения задач.* Данные о средней скорости решения задач распределены по нормальному закону ( $p < .05$ ). Для проверки распределения здесь и далее использован тест Шапиро-Уилка. Дисперсионный анализ показал отсутствие влияния фактора «тип диаграммы» на скорость решения задач (One Way ANOVA:  $F(1, 20) = 1.464$ ,  $p = .24$ ). Испытуемые решали задачи с разной скоростью. Общее время решения задач для группы Google (работающей с интерфейсом Google Analytics) — 409,83 с (179.19 SD), для группы Ours (работающей с разработанным прототипом диаграммы) — 337,05 с (87.62 SD).

*Субъективная общая нагрузка (тест NASA-TLX).* Значения оценки субъективной нагрузки распределены по нормальному закону для каждой группы. Среднее значение субъективной общей нагрузки для группы Google — 55.52 (14.61 SD), для группы Ours — 49.24 (14.57 SD). Дисперсионный анализ демонстрирует отсутствие влияния фактора «тип диаграммы» на субъектив-

ную общую нагрузку (One Way ANOVA:  $F(1,20) = 1.464$ ,  $p = .24$ ).

*Средняя длительность фиксации и стандартное отклонение средней длительности фиксации.* Данные о средней длительности фиксации распределены по нормальному закону ( $p < .05$ ) согласно тесту Шапиро-Уилка. Дисперсионный анализ показал отсутствие влияния фактора «тип диаграммы» на среднюю длительность фиксации (One way ANOVA:  $F(2,15) = .095$ ,  $p = .761$ ), так же, как и на стандартное отклонение средней длительности фиксации (One way ANOVA:  $F(2,15) = .137$ ,  $p = .715$ ). Значение средней длительности фиксации для группы Google (мс) — 286.68 (45.63 SD), для группы Ours — 293.28 (54.34 SD). Значение стандартного отклонения средней длительности фиксации для группы Google — 59.33 (17.59 SD), для группы Ours — 65.59 (53.13 SD).

*Количество правильных ответов при решении задач.* Данные о количестве правильных ответов распределены по нормальному закону ( $p < .05$ ). Дисперсионный анализ показал, что фактор «тип диаграммы» влияет на количество правильных ответов (One way ANOVA:  $F(1,20) = 28.82$ ,  $p = 0.0000296$ ). Количество правильных ответов для группы Google — 5 (из 10), для группы Ours — 8 (из 10).

На рис. 4 для визуальной оценки результатов показана зависимость от фактора «тип диаграммы» распределения значений: коэффициента субъективной общей нагрузки по тесту NASA-TLX; средней длительности фиксации; стандартного отклонения средней длительности фиксации; количества правильных ответов; общего времени решения задач.

### Интерпретация результатов и выводы

В результате проанализированных данных на влияние разных факторов на эффективность решения задачи и на поведение при решении задачи, сделаны следующие выводы. Статистический анализ показал что фактор «тип диаграммы» незначительно влияет на среднее время решения задачи. Фактор «тип диаграммы» не оказал существенного влияния на общую

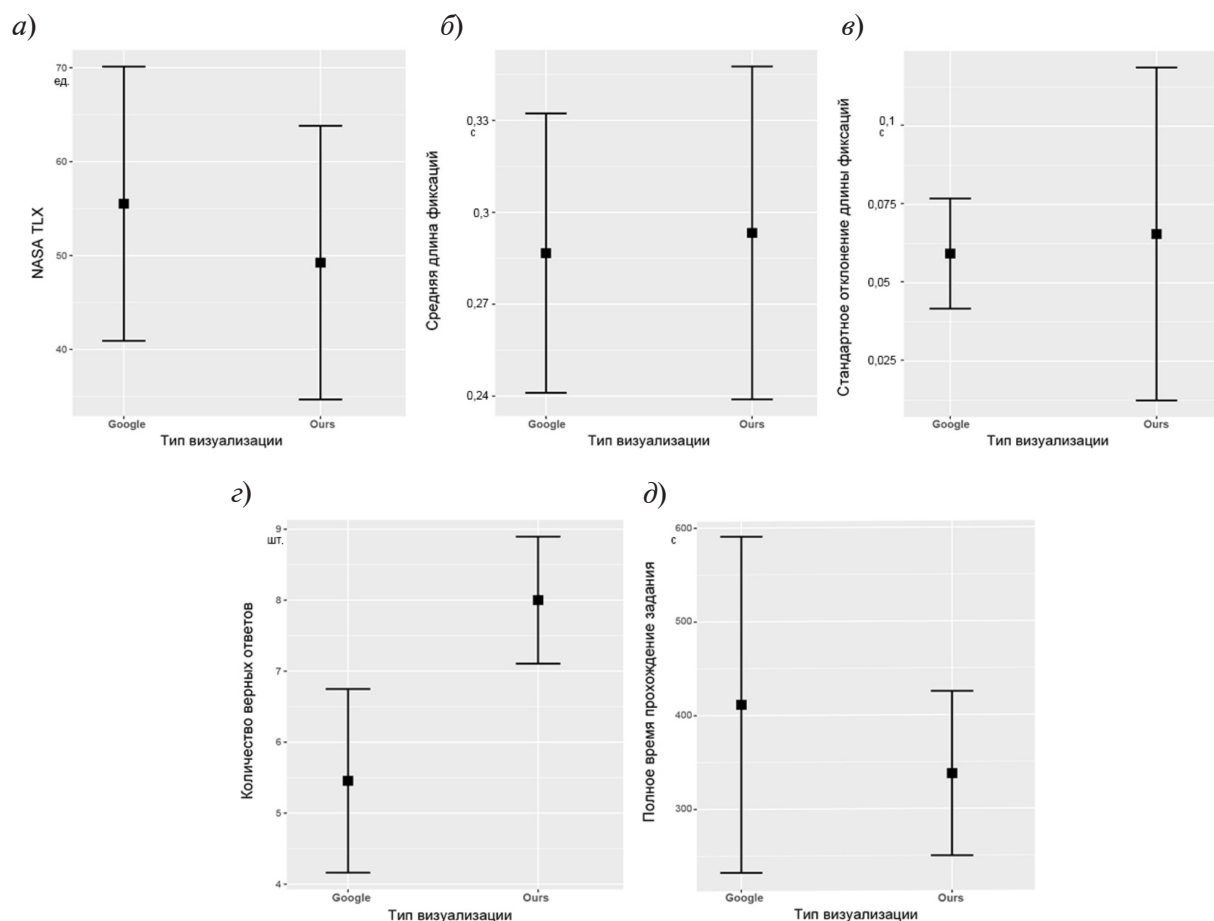


Рис. 4. Распределение значений в зависимости от фактора «тип диаграммы»: а – коэффициента субъективной общей нагрузки по тесту NASA-TLX; б – средней длительности фиксации; в – стандартного отклонения средней длительности фиксации; г – количества правильных ответов; д – общего времени решения задач

Fig. 4. Allocation of values depending on the «chart type» factor: а – the subjective total load factor for the NASA-TLX test; б – the average duration of fixation; в – the standard deviation of the mean duration of fixation; г – the number of correct answers; д – the total time for solving problems

субъективную нагрузку испытуемых. Дисперсионный анализ не обнаружил влияния типа диаграмм как на среднюю длительность фиксации, так и на стандартное отклонение в длительности фиксации. Существенное влияние фактора «тип диаграммы» было оказано на среднее количество правильных ответов.

Исходя из результатов эксперимента можно сделать следующий вывод. Несмотря на использование рекомендаций по построению потоковых диаграмм при разработке альтернативной визуализации, достичь существенной разницы в эффек-

тивности не удалось. Тем не менее следует отметить, что по таким параметрам, как скорость решения задач и количество правильных ответов, разработанный вариант эффективнее оригинальной диаграммы Google Analytics. Это подтверждает гипотезу о том, что текущая визуализация диаграммы путей пользователей Google Analytics может быть улучшена с целью повышения точности и скорости ее чтения посредством реализации следующих рекомендаций по построению потоковых диаграмм:

- наличие трех выделенных зон потока: начальной, центральной зоны перехода

и завершающей выходной зоны;

- отсутствие деформации толщины потока во всех зонах;
- идентичность блоков-узлов по размеру;
- дублирование визуальной информации, представленной потоками, числовыми показателями и дополнительной графикой.

За рамками настоящего исследования остались вопросы цветового кодирования потоков информации, формирования рекомендаций по колористике диаграмм. Это требует дальнейшего изучения.

Потоковая диаграмма путей пользователей является прикладным инструментом веб-аналитики, а значит, полученные данные можно использовать для повышения качества работы специалистов. Дальнейшая работа по повышению эффективности визуализации может затрагивать такие области, как разработка интерфейсов веб-приложений и исследование юзабилити, т. к. работа с потоковой диаграммой осуществляется в интерфейсе веб-приложения посредством использования браузера.

### СПИСОК ЛИТЕРАТУРЫ

1. Schmidt M. The Sankey diagram in energy and material flow management // J. of Industrial Ecology. 2008. No. 12 (1). Pp. 82–94. No. 12 (2). Pp. 173–185. DOI: 10.1111/j.1530-9290.2008.00004
2. Subramanyam Veena, Deepak Paramshivan, Amit Kumar, Md. Alam Hossain Mondal Using Sankey diagrams to map energy flow from primary fuel to end use // Energy Conversion and Management. 2015. No. 91. Pp. 342–352.
3. Andrienko N., Andrienko G. Visual analytics of movement: an overview of methods, tools and procedures // Information Visualization. 2013. No. 12 (1). Pp. 3–24.
4. Cuba N. Research Note: Sankey diagrams for visualizing land cover dynamics // Landscape and Urban Planning. 2015. No. 139. Pp. 163–167. DOI: 10.1016/j.landurbplan.2015.03.010
5. Fana Jing-Li, Xin Pana, Jia-Quan Lia Production-based and consumption-based CO2 transfers among major economies: A flow chart analysis // Energy Procedia. 2017. No. 105. Pp. 3499–3506.
6. Wongsuphasawat K., Gotz D. Exploring flow, factors, and outcomes of temporal event sequences with the outflow visualization // Proc. IEEE Conf. on Information Visualization / IEEE Transactions on Visualization and Computer Graphics. 2012. No. 18 (12). Pp. 2659–2668.
7. Viégas F., Wattenberg M., Hebert J., Borggaard G., Cichowlas A., Feinberg J., Orwant J., Wren Ch. Google+ Ripples: A native visualization of information flow // Proc. of the 22nd Internat. World Wide Web Conf. Rio de Janiero, 2013. Pp. 1389–1398.
8. Лаптев В.В., Орлов П.А. Кластерный анализ визуального восприятия структуры данных // Бизнес-информатика. 2015. № 3 (33). С. 34–43.
9. Laptev V.V., Orlov P.A., Zhmailova U.M., Ivanov V. Visual strategies of viewing flow visualisations under different workload conditions and representation types // Proc. of the 39th European Conf. on Visual Perception. Barcelona, 2016. 10 p.
10. Doantam Phan, Ling Xiao, Yeh R., Hanrahan P., Winograd T. Flow map layout // Proc. IEEE Symp. on Information Visualization. 2005. Pp. 219–224.
11. Buchin K., Speckmann B., Verbeek K. Flow map layout via spiral trees // IEEE Transactions on Visualization and Computer Graphics. 2011. No. 17 (12). Pp. 2536–2544. DOI: 10.1109/TVCG.2011.202.
12. Skau D., Harrison L., Kosara R. An evaluation of the impact of visual embellishments in bar charts // Proc. of Eurographics Conf. on Visualization. 2015. Vol. 34. No. 3. Pp. 221–230.
13. Bertin J. Semiology of graphics. Diagrams. Networks. Maps. Redlans: Esri Press, 2011. 438 p.
14. Lupton R.C., Allwood J.M. Hybrid Sankey diagrams: Visual analysis of multidimensional data for understanding resource use // Resources, Conservation & Recycling. 2017. No. 124. Pp. 141–151. DOI: 10.1016/j.resconrec.2017.05.002
15. Imhof E. Cartographic relief presentation. Berlin-New York: Walter de Gruyter, 2007. 388 p.
16. Duchowski A. Eye Tracking methodology. London: Springer Science+Business Media, 2007. 336 p.
17. Орлов П.А., Лаптев В.В., Иванов В.М. К вопросу о применении систем ай-трекинга // Научно-технические ведомости Санкт-Петербургского государственного политехнического университета. Информатика. Телекоммуникации. Управление. 2014. № 5 (205). С. 82–92.
18. Hart S.G. Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research // Advances in Psychology. 1988. No. 52. Pp. 139–183.

Статья поступила в редакцию 11.06.2017.



## REFERENCES

1. Schmidt M. The Sankey diagram in energy and material flow management. *Journal of Industrial Ecology*, 2008, No. 12 (1), Pp. 82–94, No. 12 (2), Pp. 173–185. DOI: 10.1111/j.1530-9290.2008.00004
2. Subramanyam Veena, Deepak Paramshivan, Amit Kumar, Md. Alam Hossain Mondal Using Sankey diagrams to map energy flow from primary fuel to end use. *Energy Conversion and Management*, 2015, No. 91, Pp. 342–352.
3. Andrienko N., Andrienko G. Visual analytics of movement: an overview of methods, tools and procedures. *Information Visualization*, 2013, No. 12(1), Pp. 3–24.
4. Cuba N. Research Note: Sankey diagrams for visualizing land cover dynamics. *Landscape and Urban Planning*, 2015, No. 139, Pp. 163–167. DOI: 10.1016/j.landurbplan.2015.03.010
5. Fana Jing-Li, Xin Pana, Jia-Quan Lia Production-based and consumption-based CO2 transfers among major economies: A flow chart analysis. *Energy Procedia*, 2017, No. 105, Pp. 3499–3506.
6. Wongsuphasawat K., Gotz D. Exploring flow, factors, and outcomes of temporal event sequences with the outflow visualization. Proceedings IEEE Conference on Information Visualization. *IEEE Transactions on Visualization and Computer Graphics*, 2012, No. 18 (12), Pp. 2659–2668.
7. Viégas F., Wattenberg M., Hebert J., Borggaard G., Cichowlas A., Feinberg J., Orwant J., Wren Ch. Google+ Ripples: A native visualization of information flow. *Proceedings of the 22nd Internat. World Wide Web Conference*, Rio de Janiero, 2013, Pp. 1389–1398.
8. Laptev V.V., Orlov P.A. Klasternyi analiz vizual'nogo vosprijatija struktury dannyh [Cluster analysis of visual perception of data structure]. *Business Informatics*, 2015, No. 3(33), Pp. 34–43. (rus)
9. Laptev V.V., Orlov P.A., Zhmailova U.M., Ivanov V. Visual strategies of viewing flow visualisations under different workload conditions and representation types. *Proceedings of the 39th European Conference on Visual Perception*, Barcelona, 2016.
10. Doantam Phan, Ling Xiao, Yeh R., Hanrahan P., Winograd T. Flow map layout. *Proceedings IEEE Symposium on Information Visualization*, 2015, Pp. 219–224.
11. Buchin K., Speckmann B., Verbeek K. Flow map layout via spiral trees. *IEEE Transactions on Visualization and Computer Graphics*, 2011, No. 17 (12), Pp. 2536–2544. DOI: 10.1109/TVCG.2011.202.
12. Skau D., Harrison L., Kosara R. An evaluation of the impact of visual embellishments in bar charts. *Proceedings of Eurographics Conference on Visualization*, 2015, Vol. 34, No. 3, Pp. 221–230.
13. Bertin J. *Semiology of Graphics. Diagrams. Networks. Maps*. Redlands: Esri Press, 2011, 438 p.
14. Lupton R.C., Allwood J.M. Hybrid Sankey diagrams: Visual analysis of multidimensional data for understanding resource use. *Resources, Conservation & Recycling*, 2017, No. 124, Pp. 141–151. DOI: 10.1016/j.resconrec.2017.05.002
15. Imhof E. *Cartographic relief presentation*. Berlin-New York: Walter de Gruyter, 2007, 388 p.
16. Duchowski A. *Eye Tracking Methodology*. London: Springer Science+Business Media, 2007, 336 p.
17. Orlov P.A., Laptev V.V., Ivanov V.M. Revisiting the Issue of Eye-Tracking Applying. *St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunication and Control Systems*, 2014, No. 5 (205), Pp. 82–92. (rus)
18. Hart S.G. Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. *Advances in Psychology*, 1988, No. 52, Pp. 139–183.

Received 11.06.2017.

## СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

ЛАПТЕВ Владимир Владимирович

LAPTEV Vladimir V.

E-mail: laptevsee@yandex.ru

ОРЛОВ Павел Анатольевич

ORLOV Pavel A.

E-mail: paul.a.orlov@gmail.com

ДРАГУНОВА Ольга Владимировна

DRAGUNOVA Olga V.

E-mail: oletta1@gmail.com

## ИССЛЕДОВАНИЕ ТЕХНОЛОГИИ ВИЗУАЛИЗАЦИИ ДАННЫХ ПРОГРАММНО-ТЕХНОЛОГИЧЕСКОЙ СИСТЕМЫ ИНТЕРАКТИВНОГО СТРАТЕГИРОВАНИЯ И БИЗНЕС-АНАЛИЗА

*А.В. Самочадин<sup>1</sup>, Е.В. Лебедев<sup>1</sup>,  
Е.В. Пономарева<sup>2</sup>, К.В. Мищенко<sup>1</sup>*

<sup>1</sup>Санкт-Петербургский политехнический университет Петра Великого,  
Санкт-Петербург, Российская Федерация;

<sup>2</sup>ИБС Экспертиза, Москва, Российская Федерация

Задачи интерактивного стратегирования предполагают активное обобщение больших объёмов структурированных данных, что требует новых архитектурных подходов к построению систем бизнес-анализа. В статье рассмотрен подход к построению виджета интерактивного стратегирования, основанный на применении в качестве корпоративного хранилища сервера OLAP, поддерживающего два процесса: процесс передачи данных для загрузки в куб и процесс выборки данных из куба для его интерактивного представления в виджете мобильных и стационарных клиентских приложений. Сформулирована общая архитектура системы, структура виджета, описаны процессы загрузки и выгрузки данных, поставлены и проведены натурные эксперименты для определения границ применимости методов управления данными в процессе интерактивного уточнения стратегий.

**Ключевые слова:** архитектура системы бизнес-анализа; виджет; OLAP; стратегирование; реляционные базы данных; файлы; ETL; интерактивные панели; отчёты; WEB-клиент.

**Ссылка при цитировании:** Самочадин А.В., Лебедев Е.В., Пономарева Е.В., Мищенко К.В. Исследование технологии визуализации данных программно-технологической системы интерактивного стратегирования и бизнес-анализа // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2017. Т. 10. № 4. С. 17–28. DOI: 10.18721/JCSTCS.10402

## THE STUDY OF THE DATA VISUALIZATION TECHNOLOGY OF A SOFTWARE SYSTEM FOR INTERACTIVE STRATEGIES AND BUSINESS ANALYSIS

*A.V. Samochadin<sup>1</sup>, E.V. Lebedev<sup>1</sup>,  
E.V. Ponomareva<sup>2</sup>, K.V. Mishchenkova<sup>1</sup>*

<sup>1</sup>Peter the Great St. Petersburg Polytechnic University,  
St. Petersburg, Russian Federation;

<sup>2</sup>IBS Expertise, Moscow, Russian Federation

Interactive strategizing tasks involve active generalization of large amounts of structured data. This requires new architectural approaches to building business analysis systems. The paper considers an approach to building a widget's architecture, based on the use of OLAP corporate storage. The storage implements two processes: generalizing the data during the download to the cube and retrieving the data from the



cube for its interactive representation in mobile and stationary client applications. In this paper, the system and widget architecture is formulated, the processes of loading and unloading of data are described and experiments were performed to determine the possibility of using data management methods in the process of interactive refinement of strategies. The results of the work can be used to select technologies for accessing data when building business analysis systems with the process of strategies.

**Keywords:** business intelligence system architecture; widget; OLAP; strategy; relational databases; files; ETL; interactive panels; reports; WEB-client.

**Citation:** Samochadin A.V., Lebedev E.V., Ponomareva E.V., Mishchenkova K.V. The study of the data visualization technology of a software system for interactive strategies and business analysis. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2017, Vol. 10, No. 4, Pp. 17–28. DOI: 10.18721/JCSTCS.10402

### Введение

Современные механизмы стратегирования сочетают управление и прогнозирование с формированием методов оценки и контроля управленческой деятельности\* [5]. Им свойственна не только фиксация конечных целей, но и описание механизма их достижения, включая мониторинг поэтапного достижения планируемых индикаторов [6]. Результат стратегирования в настоящее время представляется в виде сформированного плана действий для достижения цели [5, 12, 13]. Отдельной задачей стратегирования является мониторинг сформированной стратегии. Для этого формируется набор индикаторов\* [14], позволяющих оценить успешность процесса достижения цели.

Реализация функций отображения и мониторинга индикаторов может быть возложена на системы бизнес-аналитики, обеспечивающие сбор, хранение и управление гетерогенными данными для предоставления их конечным пользователям — руководителям верхнего звена государственных и коммерческих организаций, принимающим стратегические решения относительно путей реализации задач. Такой подход к применению информационных технологий в задачах стратегирования предложен

в работах [15, 16]. В целом, такие системы являются совокупностью инструментов, обеспечивающих интеграцию, обобщение, хранение и визуализацию данных, позволяя осуществить информационно-аналитическую поддержку интерактивного стратегирования, как механизма анализа информации из различных источников, путём формирования специализированных интерактивных стратегических карт.

Классические технологии автоматизации стратегирования представлены в программных решениях Oracle Scorecard and Strategy Management, позволяющих создать стратегическую карту, объединяющую стратегические цели и процедуры их достижения, сформировать перечень индикаторов и запустить процесс мониторинга; SAP Strategy Management, обеспечивающем графовое представление стратегической карты и набора графических отображений значений индикаторов, а также реализующем набор интерактивных графических средств отображения значений индикаторов; IBM Web Sphere Business Modeler Publishing [11] с набором средств отображения Business Leader и IBM Cognos Analytics, обеспечивающих формирование карты стратегии в терминах предметной области: цели, действия, показатели, возможности, угрозы.

Описанные программные продукты являются коммерческими приложениями с закрытым исходным кодом и облачными хранилищами, распределёнными в мировых центрах обработки данных. Применение механизмов стратегирования в задачах государственного управления предполагает использование отечественных информаци-

---

\* Соколинская Ю.М. Обеспечение экономической безопасности системы государственного регулирования социально-экономическим развитием регионов и предприятий (на примере Воронежской области): автореф. дис. канд. экон. наук. Воронеж: Воронежский гос. ун-т инженерных технологий, 2017. 22 с.

онных систем бизнес-анализа с целью обеспечения верифицируемости, технологической независимости и целостности данных в процессе государственного и корпоративного управления, что требует разработки отечественных программных продуктов интерактивного стратегирования.

### **Архитектура программно-аппаратной системы бизнес-анализа**

Рассмотренный класс систем бизнес-анализа обладает общей особенностью: для отображения данных они используют графические элементы — виджеты, позволяющие аналитику строить визуализацию данных на основе хранимых в информационной системе значений. Расширением функции мониторинга индикаторов является процедура интерактивного стратегирования, обеспечивающая проверку и оптимизацию сценариев развития ситуации в процессе взаимодействия пользователей с системой «на лету».

Предложенный подход позволяет, основываясь на уже реализованных базовых функциях отображения и агрегации данных, развивать технологии мониторинга, прогнозирования и, отчасти, моделирования. В этом случае реализация процедур интерактивного стратегирования осуществляется на двух уровнях: технологическом — подготовки исходных данных и процедур стратегирования, и на уровне анализа стратегической карты для принятия решений.

Уровень технологической подготовки данных обеспечивает обобщение и агрегацию исходных данных для формирования фактических значений индикаторов и формирования стратегических карт. Технологический уровень поддерживается классическими технологиями бизнес-анализа и реализуется системными администраторами баз данных и аналитиками данных. Уровень анализа стратегии для принятия решений предполагает взаимодействие с интерактивной стратегической картой для анализа достижимости запланированных индикаторов, поиска эффективного распределения или определения потребного объема ресурсов при реализации анализируемой задачи. Эти задачи могут быть ре-

шены руководителями при помощи стратегических карт самостоятельно, без участия аналитиков, что является отличительной чертой рассматриваемого в статье подхода к реализации механизма интерактивного стратегирования.

Функционально разрабатываемая система бизнес-аналитики обеспечивает выполнение процессов загрузки, обработки, визуализации данных, выгрузку данных и хранение служебной информации [1]. Анализ архитектурных решений систем бизнес-аналитики на примере [2–4] показал, что они характеризуются общими технологическими приёмами, к которым относятся: интеграция разнородных источников данных, таких как реляционные базы данных, OLAP [9], плоские файлы при помощи ETL-механизмов сбора и трансформации [8]; представление собранных данных в корпоративном хранилище, например, OLAP [7, 9] или репозитории метаданных [10]; графические средства формирования интерактивных панелей; автоматическое и ручное формирование отчётов в различных форматах; наличие расширенных механизмов обеспечения безопасности функционирования системы, включая управление пользователями и разграничение прав доступа к объектам.

Большое число одновременно загружаемых, редактируемых и выгружаемых панелей и отчётов, интерактивно взаимодействующих виджетов, пользователей в системе выдвигают дополнительные условия к пропускной способности компонентов системы бизнес-анализа и требуют реализации архитектурных подходов, обеспечивающих выполнение требований к скорости загрузки, обработки и выгрузки визуализируемых данных.

Реализация функции визуализации данных в интерактивных панелях обеспечивается функционированием двух параллельных процессов: процесса загрузки данных из внешних хранилищ во внутреннее и обмена данными с внутренним хранилищем, сочетающем процесс чтения данных для визуализации и записи изменённых значений из интерактивной панели. Для максимального разделения этих процессов систе-

ма реализована в виде набора компонентов, выполнение функций которых может быть распараллелено: компоненты, обрабатывающие внешние данные и помещающие их во внутреннее хранилище, и компоненты, обрабатывающие сохранённые данные с целью их визуализации и публикации. В этом случае хранилище данных является приёмником данных как из внешних источников, так и от виджетов. Тогда за взаимодействие со внешними источниками отвечает компонент загрузки и обработки, а управление внутренними потоками обеспечивают компоненты выгрузки, хранения служебных данных, визуализации и управления.

Компонент загрузки и обработки реализует функцию подключения к внешним источникам данных, исполнения ETL-цепочек загрузки, в том числе очистки и агрегации данных. Результат выполнения записывается во внутреннее хранилище данных, являющееся частью компонента хранения служебных данных и состоящее из OLAP-куба, и хранилища данных пользователей во внутренней реляционной СУБД. Компонент выгрузки обеспечивает формирование запросов на создание экземпляров отчётов по расписанию или запросу пользователей. Компонент визуализации и управления обеспечивает взаимодействие с пользователями, применяющими мобильные устройства и стационарные компьютеры для решения задачи построения интерактивных панелей и отчётов. В нём используется функция чтения и записи данных в OLAP-куб для реализации интерактивных панелей стратегирования.

Процесс загрузки данных из внешних источников начинается с создания аналитиком новой модели данных — описания представления загружаемых данных в форме OLAP-куба через интерфейс компонента визуализации и управления. Затем аналитик создаёт одну или несколько цепочек загрузки и сохраняет их в компоненте хранения служебных данных. Выполнение цепочки загрузки обеспечивает извлечение данных из внешних баз, банков и хранилищ данных, очистку, консолидацию и агрегацию данных, заполнение модели полученными данными и сохранение её как набора дан-

ных. Интерфейс создания и выполнения цепочек реализован в компоненте визуализации и управления, а процедуры выполнения цепочек загрузки — в компоненте загрузки и обработки.

Процесс создания экземпляра интерактивной панели реализуется в два этапа: создание интерактивной панели и генерация экземпляра панели. Для создания интерактивной панели аналитик выбирает один или несколько наборов данных из хранилища и один или несколько виджетов, с его точки зрения, наиболее полно отражающих показатели стратегирования. Выбранная комбинация виджетов обеспечивает извлечение данных из наборов в интерактивном режиме. Созданная панель или отчёт хранятся в реляционной базе данных компонента хранения служебных данных. Генерация экземпляра отчёта осуществляется по запросу пользователя или по расписанию. Запрос пользователя выполняется в интерфейсе визуализации и управления со стационарного или мобильного устройства, поступает в компонент выгрузки, который обращается к компоненту загрузки и обработки для получения отчёта или панели и актуализации данных из OLAP-хранилища. Расписание создаётся в процессе подготовки отчёта или панели, сохраняется вместе с отчётом в компоненте хранения служебных данных. Компонент выгрузки хранит актуальное расписание создания отчётов и по наступлению события вызывает автоматическую генерацию отчёта. Созданный актуальный экземпляр отчёта сохраняется в локальном хранилище, может быть напечатан или отправлен по электронной почте.

Выполнение процессов взаимодействия виджета и OLAP-куба обеспечивает параллельную реализацию операций загрузки и извлечения данных из OLAP-хранилища компонента загрузки и обработки и должно выполняться в интерактивном режиме. В этом случае OLAP-хранилище становится ключевым элементом в процессе функционирования виджета стратегирования, и исследование времени выполнения запросов к нему позволит определить требования к масштабированию хранилищ компоненты хранения служебных данных для интерак-

тивного взаимодействия с компонентой загрузки и обработки.

### Виджет интерактивного стратегирования системы бизнес-анализа

Формирование интерактивных стратегических карт основано на работе специализированного виджета, в функции которого входят: выбор перечня индикаторов, требуемых для построения стратегии; выбор и определение значений достигаемых индикаторов; определение перечня и объёмов требуемых ресурсов; задание дерева «и-или» для определения пути достижения стратегии, привязки узлов дерева к потребным ресурсам и определения правил потребления ресурса при переходе от одного узла дерева к другому. Схема функционирования

виджета представлена на рис. 1.

Перечень индикаторов определяется аналитиком, а значения индикаторов извлекаются средствами загрузки данных и обеспечивают демонстрацию изменения индикаторов во времени. Перечень достигаемых индикаторов и потребных ресурсов определяется аналитиком, значения индикаторов и ресурсов могут быть введены вручную или определены автоматически в процессе функционирования виджета. Дерево «и-или» определяет порядок действий для достижения результата. Узлами дерева являются действия, а рёбрами — переходы между ними. В общем случае дерево представляет поток работ, обеспечивающий достижение требуемых показателей. С каждым узлом дерева связан управляющий элемент,

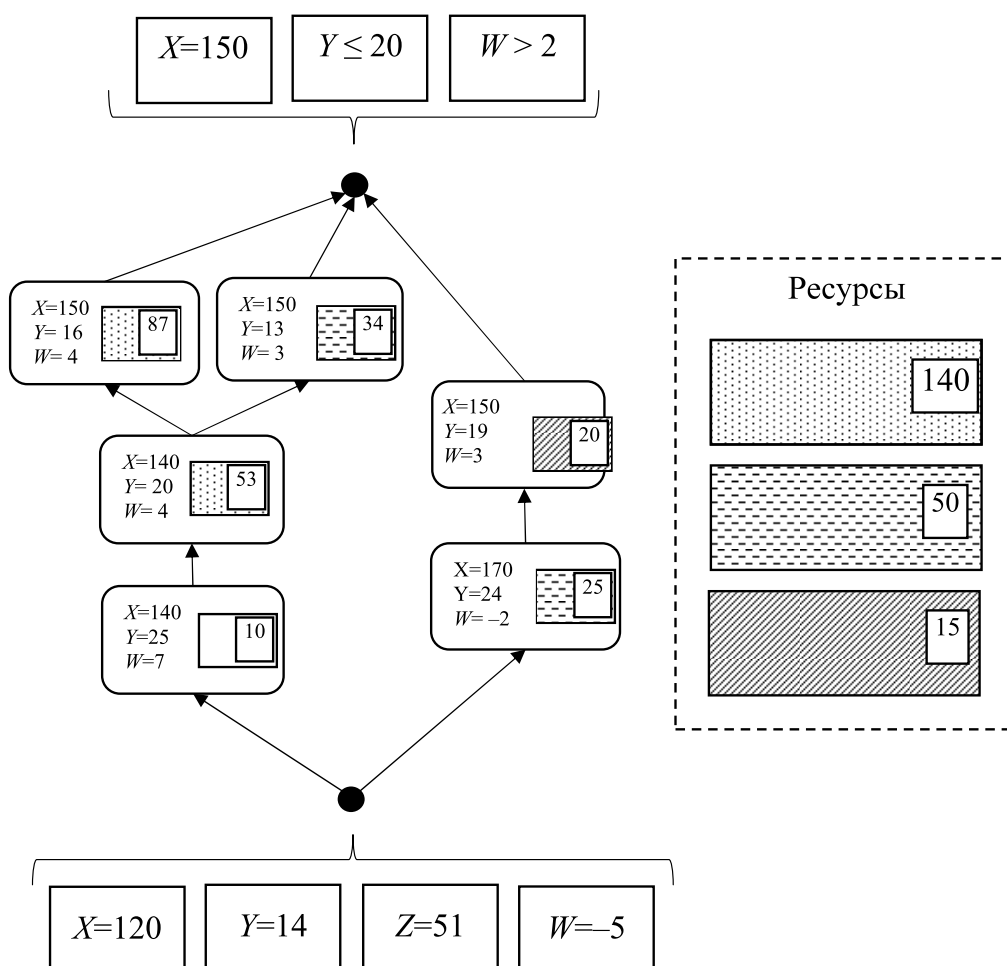


Рис. 1. Схема функционирования виджета стратегирования

Fig. 1. Strategic planning widget. Schema of execute



демонстрирующий правило потребления и размер выделяемого ресурса.

Виджет функционирует в двух режимах: редактирования и управления. В режиме редактирования определяются исходные и результирующие индикаторы, определяются потребные ресурсы и строится дерево достижения целей. К узлам дерева прикрепляются ресурсы и определяется правило потребления ресурсов на каждом шаге. В режиме выполнения виджет обеспечивает решение прямой и обратной задач: прямая задача состоит в определении достижимости цели при заданных ресурсах. Аналитик вводит значения ресурсов и запускает процесс поиска решения. В результате отображаются возможные пути достижения результирующих показателей. Обратная задача решается от результатов: аналитик вводит потребные значения результирующих индикаторов и запускает решение. В результате отображаются потребные ресурсы для получения результирующих индикаторов. В режиме управления по окончании вычислений в каждом из узлов отображается рассчитанный уровень потребления ресурса каждого типа.

Предложенный к реализации виджет обеспечивает решение задачи интерактивного стратегирования, причём последовательное ручное и автоматическое изменение параметров и наборов данных в виджетах с целью оптимизации предлагаемых решений требует обеспечения двустороннего обмена командами и данными между виджетом и хранилищем данных в интерактивном режиме, что требует исследования технологических и реализационных ограничений, накладываемых на процедуры их взаимодействия.

#### **Исследование времени выполнения операций чтения, обновления и записи данных в OLAP-хранилище из виджета**

Типичным сценарием построения интерактивной панели стратегирования является добавление виджета, определение в нём массивов желаемых значений индикаторов, извлечение из куба текущих значений с целью их последующего ручного или автоматического сравнения. В результате

при функционировании каждой информационной панели возникают два встречных потока — записи и чтения данных в куб. Функционирование куба в таком режиме является нетипичным для классических систем бизнес-анализа ввиду большого числа потоков с малыми объёмами данных.

С целью определения максимального объёма данных, при котором взаимодействие с виджетом сохраняет интерактивный характер, проведён натурный эксперимент над данными кубов. Было выделено два ключевых процесса: загрузка данных в куб из виджета и демонстрация данных, состоящая из подпроцессов извлечения данных из куба и их последующей обработки в браузере.

Загрузка данных в куб моделировала процедуры функционирования компонентов в части определения скорости обращения к хранилищу данных в OLAP из виджета. Целью эксперимента было определение зависимости времени записи данных в куб от объёма передаваемых из виджета данных. Процедура выгрузки данных из куба моделировала процесс получения данных виджетами интерактивных панелей. Цель эксперимента состояла в определении вклада каждого этапа извлечения и отображения данных в общее время получения данных.

Для демонстрации работоспособности технологии взаимодействия с OLAP-кубом выбран виджет таблицы как элемент, отображающий наибольший объём выбираемых данных. Пример реализации интерфейса виджета «таблица» приведён на рис. 2.

Для реализации эксперимента собран программно-аппаратный стенд натурального моделирования, состоящий из OLAP-сервера и клиентской части, размещённой на рабочей станции. Сервер и рабочая станция были подключены к сети по каналу связи со скоростью 10 Мбит/с и соединены реальными каналами Интернета, находясь на расстоянии около 1,5 тыс. км друг от друга, что позволило имитировать реальную работу локальной и глобальной сетей. В качестве сервера OLAP использовался ROLAP PostgreSQL [7] на однопроцессорной машине Core i5 с 32 Гб оперативной памяти. Таблицы куба располагались

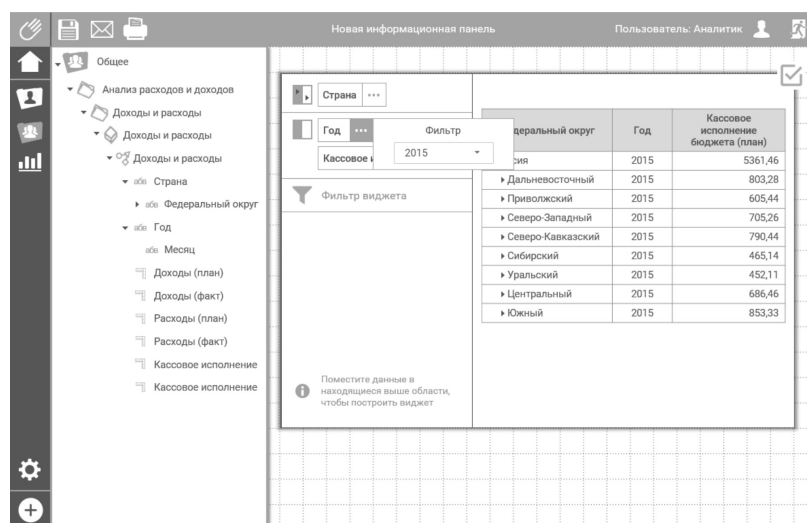


Рис. 2. Пример интерфейса для виджета «таблица»

Fig. 2. «Table» widget interface example

в оперативной памяти с использованием технологии «in memoгу». Клиентская часть выполнялась на мобильном компьютере с процессором Core i5 и 4 Гб оперативной памяти. Загрузка данных в куб осуществлялась через WEB API, реализованный вызовами функций из MS Excel. Получение данных осуществлялось при помощи специально разработанного на языке Java программного обеспечения, которое содержало вызовы виджета «таблица» из би-

блиотеки PrimeFace [8]. Проверка времени визуализации проводилась одновременно в нескольких браузерах. В эксперименте использовались браузеры: Safari 11.0.1 (x64) с визуализаторами WebKit, Nitro; Chrome 63.0.3239.84(x64) с визуализатором Blink V8, Opera 49.0.2725.39(x64) с визуализатором Blink V8 и Firefox 57.0.2(x64) с визуализаторами Quantum и SpiderMonkey. Параметры и константы серии экспериментов приведены в таблице.

#### Параметры и константы экспериментов The experiment constants and parameters

| Параметр, константа                 | Значение                                                               |
|-------------------------------------|------------------------------------------------------------------------|
| Технические характеристики          |                                                                        |
| Сервер ROLAP                        | Core i5, 32 Мб RAM, OS Debian                                          |
| Клиенты                             | 2х1,4 ГГц Core i5, 4 Гб RAM, OS iOS, Windows                           |
| Браузеры                            | Safari 11.0.1, Chrome 63.0.3239.84, Opera 49.0.2725.39, Firefox 57.0.2 |
| Скорость подключения к Сети, Мбит/с | 10                                                                     |
| Средства разработки                 | MS Excel, PrimeFaces, Java, JavaScript                                 |
| Характеристики данных               |                                                                        |
| Число экспериментов в серии, шт.    | 10                                                                     |
| Размер куба данных                  | 1×1×1 ... 30×30×10                                                     |
| Объем данных, байт                  | 8 ... 72000                                                            |
| Число записей, шт.                  | 1 ... 9000                                                             |
| Тип данных                          | Double precision                                                       |



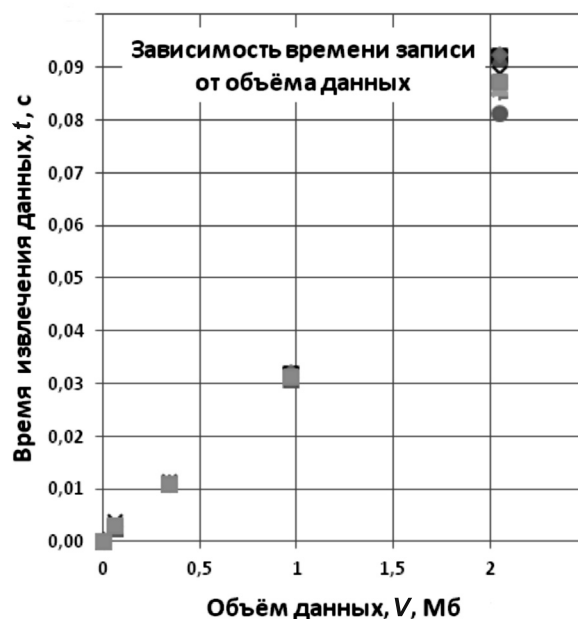


Рис. 3. Зависимость времени записи данных от объема выборки

Fig. 3. The dependent of the data recording time on the block size

В ходе первой части эксперимента осуществлялась загрузка данных типа *double precision* в терминальные ячейки куба, начиная с области размером  $1 \times 1 \times 1$  (8 байтов), заканчивая размером  $30 \times 30 \times 10$  (72 Кбайт). Для записи каждого размера выполнялась

серия экспериментов и фиксировалось время начала и окончания записи данных в OLAP на стороне клиента. Результаты измерений приведены на рис. 3.

Значения времени записи, указанные на графике, показывают, что обеспечивается

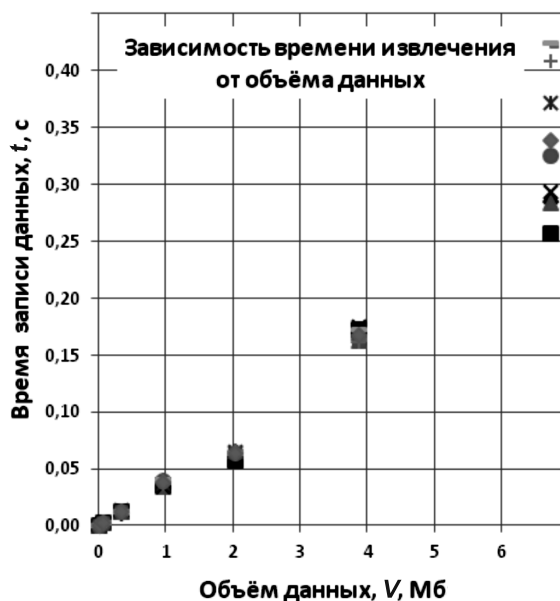


Рис. 4. Зависимость времени извлечения от объема данных

Fig. 4. The dependent of the data reading time on the data size

линейная зависимость времени выполнения операции от объёма отправляемых данных. На больших объёмах записываемых данных наблюдается увеличение разброса времени выполнения операции, связанное с необходимостью корректировки нетерминальных ячеек по результатам изменения терминальных. Из результатов эксперимента можно сделать вывод о том, что запись данных в OLAP-куб может выполняться за линейное от объёма данных время.

Чтение данных из терминальных ячеек куба осуществлялось из клиента OLAP путём фиксации времени начала и окончания операции, в аналогичном записи объёме. Результаты измерений приведены на рис. 4.

Из графиков видно, что обеспечивается линейная зависимость времени выполнения операции извлечения от объёма извлекаемых данных. На больших объёмах данных наблюдается увеличение разброса времени выполнения операции, связанное с ожиданием запроса данных из кеша сервера. Из результатов эксперимента можно сделать вывод о том, что время на запрос типичного объёма данных для виджета выполняет-

ся за время, обеспечивающее комфортную реакцию пользователя при интерактивной обработке данных.

Вторая серия экспериментов проводилась с целью исследования времени преобразования данных на стороне сервера и отображения данных на стороне клиента в разных браузерах. Суммарное время складывалось из формирования страницы с виджетом на стороне сервера и времени обработки виджета в браузере на клиентской стороне. Для сравнения времени обработки были выбраны типичные браузеры последних, на момент проведения экспериментов, стабильных версий. Результаты сравнения времени обработки приведены на рис. 5.

Из графиков видно, что время обработки данных в коде Java для генерации страницы, содержащей визуализационный компонент PrimeFaces, практически не зависит от объёма данных и составляет около 0,1 мс. Время работы браузеров с визуализационными компонентами зависит от технологии визуализации в браузере: **Chrome** и **Opera**, использующие визуализатор Blink, отображали виджет на

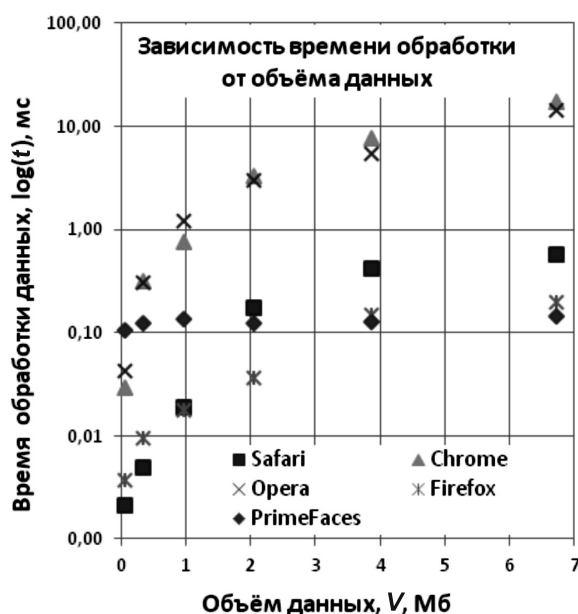


Рис. 5. Зависимость времени обработки данных клиентской и серверной сторонами от объёма входных данных

Fig. 5. The dependent the data processing time by the client and server sides on the input data amount

порядок медленнее, чем браузеры Safari и Firefox. Несмотря на это, все браузеры обеспечивают отображение данных с комфортным интерактивным откликом при отображении в браузере не менее 10 тыс. значений ячеек таблицы. Дополнительное тестирование браузера Safari на интерпретацию максимального объема данных показало, что границей времени интерактивного отклика является загрузка примерно 6 Мб данных, что соответствует примерно 0,75 млн данных типа *double precision*, что можно считать верхним ограничением при принятии решения о продолжении работы с интерактивной панелью.

### Заключение

В статье рассмотрены и исследованы два ключевых процесса работы виджета интерактивного стратегирования в системе бизнес-анализа: загрузки данных в OLAP-куб из виджета и визуализации данных в интерактивном виджете. Представлено описание виджета интерактивного стратегирования, архитектура системы в части выполнения задачи взаимодействия с многомерным хранилищем данных, реализован

макет системы, поставлены и проведены натурные эксперименты по определению времени взаимодействия прототипов компонентов интерактивной панели стратегирования в процессе загрузки и выгрузки данных из куба. В ходе натурных экспериментов анализ времени записи и получения данных из локального OLAP-хранилища показал, что комфортное взаимодействие с пользователем обеспечивается при объеме обмена данными не более 10 тыс. значений ячеек таблицы, а 0,75 млн значений ячеек является максимальным объемом данных, при котором сохраняется интерактивное взаимодействие пользователя с системой.

Проведенные исследования показывают, что предложенные архитектурные решения применимы для анализа данных путём формирования интерактивных форм для реализации задач интерактивного стратегирования.

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

### СПИСОК ЛИТЕРАТУРЫ

1. Архитектура BI-систем // URL: <https://qleverolutions.ru/%D0%B0%D1%80%D1%85%D0%B8%D1%82%D0%B5%D0%BA%D1%82%D1%83%D1%80%D0%B0-bi-%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC/>
2. Архитектура Галактика BI // URL: <https://www.galaktika.ru/bi/arxitektura.html>
3. Шмаков А. Сервис-ориентированный подход в бизнес-аналитике от Oracle // URL: <http://citforum.ru/database/oracle/soaway>.
4. Marín-Ortega P.M., et al. ELTA: New approach in designing business intelligence solutions in era of big data // *Procedia Technology*. 2014. Vol. 16. Pp. 667–674.
5. Соколинская Ю.М. Механизмы стратегирования в повышении экономической безопасности предприятий // *Экономика и предпринимательство*. 2017. № 8-2. С. 480–488.
6. Зельднер А.Г. Место стратегирования в понятийно-категориальной системе прогнозирования // *Экономические науки*. 2012. № 93. С. 7–15.
7. Sabtu A., et al. The challenges of extract, transform and loading (ETL) system implementation for near real-time environment // *Proc. of the Internat. Conf. on Research and Innovation in Information Systems*. Langkawi, 2017. Pp. 1–5.
8. Ioannidis L., Bratsas C., Karabatakis S., Filippidis P., Bamidis P. Rudolf: an HTTP API for exposing semantically represented fiscal OLAP cubes // *11th International Workshop on Semantic and Social Media Adaptation and Personalization*. Thessaloniki, 2016. Pp. 177–182.
9. Скворцова Д.А., Чмырь Д.А. Использование многомерных OLAP-кубов, как инструмента Business Intelligence, при стратегическом управлении бизнес-процессами компании // *Экономика: теория и практика*. 2016. № 3. С. 70–77.
10. Попов С.Г. Методика построения оптимального репозитория схем реляционных баз данных // *Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление*. 2010. № 5 (108). С. 91–97.
11. IBM Knowledge Center, карты стратегий // URL: <https://www.ibm.com/support/>

knowledgecenter/ru/SSCRHV\_6.2.0/com.ibm.websphere.wbpm.fep.bmps.620.doc/doc/aboutstrategymaps.html

12. **Мишарин Ю.В.** Стратегирование в инновационной деятельности // *Фундаментальные исследования*. 2013. № 11-5.

13. **Марача В.Г.** Региональное стратегирование как метод повышения эффективности государственного управления региональным развитием в Российской Федерации // *Труды ИСА РАН*. 2008. Т. 34. С. 179–214.

14. **Доргущаова А.К.** Развитие индикативно-

го подхода к региональному стратегированию в целях наращивания потенциала импортозамещения // *Крымский научный вестник*. 2015. № 5-1.

15. **Гальченко С.А.** Место и роль информационных систем управления в стратегировании регионального развития // *Политика, экономика и инновации*. 2016. № 4.

16. **Цыбатов В.А.** Стратегирование регионального развития: методы, модели, информационные технологии // *Региональная экономика: теория и практика*. 2015. № 27 (402).

*Статья поступила в редакцию 11.11.2017.*

## REFERENCES

1. *BI-system architecture* (rus). Available: <https://qlerversolutions.ru/%D0%B0%D1%80%D1%85%D0%B8%D1%82%D0%B5%D0%BA%D1%82%D1%83%D1%80%D0%B0-bi-%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC/>

2. *Architecture Galaktika BI system* (rus). Available: <https://www.galaktika.ru/bi/arxitektura.html>

3. **Shmakov A.** *Oracle's service-oriented approach in business analytics* (rus). Available: <http://citforum.ru/database/oracle/soaway>.

4. **Marín-Ortega P.M., et al.** ELTA: New approach in designing business intelligence solutions in era of big data. *Procedia Technology*, 2014, Vol. 16, Pp. 667–674.

5. **Sokolinskaya Yu.M.** Mechanisms of strategic development in improving the economic security of enterprises. *Economics and Entrepreneurship*, 2017, No. 8-2, Pp. 480–488. (rus)

6. **Zeldner A.G.** The place of strategy in the conceptual-categorical forecasting system. *Economic Sciences*, 2012, No. 93, Pp. 7–15. (rus)

7. **Sabtu A., et al.** The challenges of extract, transform and loading (ETL) system implementation for near real-time environment. *Proceedings of the International Conference on Research and Innovation in Information Systems*, Langkawi, 2017, Pp. 1–5.

8. **Ioannidis L., Bratsas C., Karabatakis S., Filippidis P., Bamidis P.** Rudolf: An HTTP API for exposing semantically represented fiscal OLAP cubes. *11th International Workshop on Semantic and Social Media Adaptation and Personalization*, Thessaloniki, 2016, Pp. 177–182.

*Received 11.11.2017.*

9. **Skvortsova D.A., Chmyr D.A.** Using multidimensional OLAP cubes as a tool of Business Intelligence, in the strategic management of the company's business processes. *Economics: Theory and Practice*, 2016, No. 3, Pp. 70–77. (rus)

10. **Popov S.G.** Method of construction of optimal repository of relational schemas databases. *St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunication and Control Systems*, 2010, No. 5 (108), Pp. 91–97. (rus)

11. *IBM Knowledge Center, karty strategiy*. Available: [https://www.ibm.com/support/knowledgecenter/ru/SSCRHV\\_6.2.0/com.ibm.websphere.wbpm.fep.bmps.620.doc/doc/aboutstrategymaps.html](https://www.ibm.com/support/knowledgecenter/ru/SSCRHV_6.2.0/com.ibm.websphere.wbpm.fep.bmps.620.doc/doc/aboutstrategymaps.html)

12. **Misharin Yu.V.** Strategy in innovation activity. *Fundamental research*, 2013, No. 11-5. (rus)

13. **Maracha V.G.** Regional strategy as a method of increasing the efficiency of state management of regional development in the Russian Federation. *Proceedings of the ISA RAS*, 2008, Vol. 34, Pp. 179–214. (rus)

14. **Dorgushaova A.K.** Development of an indicative approach to regional strategy for building the capacity of import substitution. *Crimean Scientific Bulletin*, 2015, No. 5-1. (rus)

15. **Galchenko S.A.** Place and role of information management systems in the strategic development of regional development. *Politics, Economics and Innovation*, 2016, No. 4. (rus)

16. **Tsybatov V.A.** Strategic development of regional development: methods, models, information technologies. *Regional Economy: Theory and Practice*, 2015, No. 27 (402). (rus)

**СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS**

**САМОЧАДИН Александр Викторович**  
**SAMOCHADIN Alexander V.**  
E-mail: samochadin@gmail.com

**ЛЕБЕДЕВ Евгений Владимирович**  
**LEBEDEV Evgeny V.**  
E-mail: eugenelebedevrus@gmail.com

**ПОНОМАРЕВА Елена Всеволодовна**  
**PONOMAREVA Elena V.**  
E-mail: eponomareva@ibs.ru

**МИЩЕНКОВА Катерина Владимировна**  
**MISHCHENKOVA Katerina V.**  
E-mail: mishenkova.katya@yandex.ru



## ПРОТОКОЛ АУТЕНТИФИКАЦИИ ДЛЯ ПЕРЕДАЧИ ПРАВ ПОЛЬЗОВАНИЯ ЭЛЕКТРОННЫМИ УСТРОЙСТВАМИ

*А.Я. Омётов<sup>1</sup>, С.В. Беззатеев<sup>2</sup>, Е.А. Кучерявый<sup>3</sup>*

<sup>1</sup>Санкт-Петербургский государственный университет телекоммуникаций  
имени профессора М.А. Бонч-Бруевича, Санкт-Петербург, Российская Федерация;

<sup>2</sup>Санкт-Петербургский национальный исследовательский университет информационных технологий,  
механики и оптики, Санкт-Петербург, Российская Федерация;

<sup>3</sup>Национальный исследовательский университет «Высшая школа экономики»,  
Москва, Российская Федерация

Количество носимой электроники в современном мире неуклонно растет. Традиционно за обеспечение безопасности электроники такого типа в условиях Интернета вещей отвечали алгоритмы аутентификации. В статье представлен протокол аутентификации, специально разработанный для функционирования в условиях нестабильного соединения с удостоверяющим центром (центром сертификации). Подобные сценарии могут возникать и в случае повышенной нагрузки на существующую сеть, и в местах с низким сотовым покрытием или при функционировании в труднодоступных (удаленных) зонах. Рассмотрены ближайшие аналоги протокола, отмечены его преимущества относительного рассматриваемого сценария функционирования. Приведено детальное описание рабочих этапов протокола для обеспечения целостности и конфиденциальности данных владельца и пользователя устройства.

**Ключевые слова:** аутентификация; протокол; Интернет вещей; беспроводная связь; нестабильный канал.

**Ссылка при цитировании:** Омётов А.Я., Беззатеев С.В., Кучерявый Е.А. Протокол аутентификации для передачи прав пользования электронными устройствами // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2017. Т. 10. № 4. С. 29–40. DOI: 10.18721/JCSTCS.10403

## AUTHENTICATING ELECTRONIC DEVICES FOR TEMPORARY USE: ACCESS RIGHTS MANAGEMENT PROTOCOL

*A.Ya. Ometov<sup>1</sup>, S.V. Bezzateev<sup>2</sup>, Y.A. Koucheryavy<sup>3</sup>*

<sup>1</sup> Bonch-Bruevich St. Petersburg State University of Telecommunications,  
St. Petersburg, Russian Federation;

<sup>2</sup> St. Petersburg National Research University of Information Technologies, Mechanics and Optics,  
St. Petersburg, Russian Federation;

<sup>3</sup> National Research University Higher School of Economics, Moscow, Russian Federation

The number of active wearable devices per user is growing daily. Conventionally, authentication algorithms were responsible for the security of such systems especially in terms of the Internet of Things (IoT) paradigm. This manuscript proposes an authentication protocol specifically designed for operation under unreliable connectivity to trusted authority constraints. Such situations may occur both in cases of network overload and in distant areas with low network coverage. The summary on existing solutions as well as the benefits brought by the solution are discussed. A detailed description of the protocol execution phases enabling integrity and privacy for the owner and the temporary user is also given.

**Keywords:** authentication; protocol; Internet of Things; wireless communication; unreliable connection.

**Citation:** Ometov A.Ya., Bezzateev S.V., Koucheryavy Y.A. Authenticating electronic devices for temporary use: access rights management protocol. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2017, Vol. 10, No. 4, Pp. 29–40. DOI: 10.18721/JCSTCS.10403

## Введение

На протяжении развития сотовой связи и электронной торговли обеспечение безопасности при передаче данных является важнейшей задачей [1]. В настоящее время проверка подлинности пользователей, устройств Интернета вещей, событий и систем производится с помощью протоколов аутентификации [2]. Отличительная черта протоколов аутентификации, используемых в современных IP сетях, — необходимость наличия удостоверяющего центра (УЦ), который несёт ответственность также и за маршрутизацию сообщений между удалёнными узлами [3]. Основное достоинство таких систем — абсолютная доверительность УЦ, который берет на себя ответственность за генерацию, распределение, обновление и отзыв ключей/сертификатов всех пользователей системы [4].

В то же время развитие рынка носимой электроники диктует новые тренды в мире беспроводных технологий и информационной безопасности [5]. Однако сравнительно небольшие по размеру устройства, такие как носимая электроника и объекты умного дома, все ещё обладают невысокими вычислительными мощностями и ограничены в возможностях управления [6]. Рынок носимой электроники только начал развиваться, поэтому целевая аудитория пользователей ещё не достигла больших масштабов [7]. Общественность не готова приобретать новые дорогостоящие продукты носимой электроники. В связи с этим, для увеличения потребительской базы, не-

долговременная аренда таких устройств может послужить выгодным сценарием. С другой стороны, по стандартам современных производителей, смена пользователей может быть осуществлена только при условии полной очистки устройства до заводского состояния [8].

Данный сценарий — временная передача управления устройствами — может происходить как в местах с отличным сотовым покрытием (городах), так и в труднодоступных районах [9, 10] (горные курорты, морские круизы и т. п.). Во втором случае необходимость постоянного соединения с УЦ может оказаться серьёзной проблемой, что влечёт за собой трудности однозначного подтверждения владения устройством, обновления правил передачи, а также продления условий аренды.

Таким образом, задача данного исследования заключается в обеспечении возможности временной передачи управления устройствами в условиях непостоянного соединения с УЦ с целью предоставления гарантий целостности и конфиденциальности данных. Однако полный отказ от взаимодействия с УЦ и переход на полностью распределённые решения также является невозможным. Например, аренда устройств в контексте таких бизнес-моделей, как B2B или B2C, предполагает необходимость формального подтверждения факта передачи и возврата носимой электроники не только на нижнем уровне взаимодействия, но и на централизованном узле (сервере). Основная цель исследования — разработ-

ка семейства протоколов информационной безопасности, устраняющего указанную выше проблему.

### **Обзор ближайших аналогов и постановка задачи**

К сожалению, современная научная литература не рассматривает проблему делегирования устройств в контексте нестабильного соединения с УЦ. Тем не менее далее приведён обзор работ, наиболее близких по проблематике.

В [11] представлен протокол, позволяющий снизить объёмы вычислений для устройств с существенно низким уровнем вычислительных ресурсов. В данном протоколе агрегирующий узел (АУ) отвечает за выпуск ключей дочерних устройств. Таким образом снижаются энергозатраты на устройствах с менее ёмкой батареей. Основным преимуществом предложенного решения является передача трудоёмкой процедуры рукопожатий генерации ключа на АУ. С другой стороны, недостатком является необходимость стабильного соединения с УЦ. Работа при наличии такого соединения требует прохождения минимум 7 и максимум 15 рукопожатий в фазе ассоциации.

В работе [12] представлен алгоритм, основанный на эллиптических кривых, который позволяет осуществлять аутентификацию устройств при помощи технологий связи беспроводных локальных сетей. Основными недостатками данной методики являются: необходимость постоянного наличия защищённого канала между аутентифицируемым устройством и УЦ, а также отсутствие возможности проверки истории взаимных аутентификаций.

Протокол, предлагаемый в данной статье, разработан с учётом проблем, описанных выше. Его основная задача — обеспечение аутентификации с целью последующего делегирования электронных устройств в обоих случаях — при наличии и отсутствии защищённого канала до УЦ. Благодаря возможности работы системы даже в случае отказа соединения с УЦ во время установления связи между владельцем устройства и потенциальным арендатором, протокол

позволяет устранить недостатки известных технических решений. Система нацелена на обеспечение целостности данных и противодействие ряду атак на передаваемое устройство как со стороны владельца, так и со стороны арендатора устройства.

### **Описание протокола аутентификации**

Предлагаемый протокол обеспечивает функционирование в сценариях с потенциально нестабильным соединением от АУ к УЦ, а также в случае наличия возможности осуществления прямых соединений между АУ и делегируемым устройством и АУ разных пользователей.

Поставленная задача решается за счёт комбинирования решений, рассматриваемых в системах с инфраструктурой открытых ключей, которые отвечают за начальную генерацию ключей и сертификатов, а также решений, позволяющих осуществлять прямое соединение при отсутствии связи с инфраструктурой открытых ключей.

Основные варианты криптографических решений для потенциального использования в данном протоколе предложены ниже. Для получения результата хеш-функции данных могут быть использованы алгоритмы ГОСТ Р 34.11-2012, SHA-2, SHA-3, BLAKE2 [13], для ЦУ возможно использование классических протоколов криптосистем с открытым ключом [14–16]. Защита от атак типа «человек посередине» (атака посредника) может осуществляться посредством классического протокола Diffie-Hellman [17] или Elliptic curve Diffie-Hellman (ECDH) [18].

Протокол аутентификации для делегации прав пользования электронными устройствами состоит из определённых операционных этапов (шагов): (1) начальная ассоциация нового устройства; (2) передача устройства на определённый промежуток времени; (3) возврат устройства в пользование владельцу; (4) диссоциация устройства.

**Начальная ассоциация нового устройства.** Исходными данными являются уникальный идентификатор пользователя ID (например, user@mail.com), который используется для взаимодействия между участниками протокола (УЦ, пользователями системы и

делегируемым устройством). Участниками протокола на этом этапе являются ранее не используемое делегируемое электронное устройство (ДЭУ) и агрегирующий узел (например, смартфон пользователя). Далее, УЦ выпускает сертификаты для АУ пользователя и нового ДЭУ, заверяя целостность прошивки ДЭУ и его однозначную принадлежность к АУ.

Шаги протокола данного этапа показаны на рис. 1. Здесь и далее владельцем ДЭУ считается пользователь Alice; арендатором устройства — пользователь Bob. Защищённым каналом определяется устойчивое к атаке «человек посередине» соединение. Каждый пользователь обладает уникальным идентификатором  $ID_A$  типа `alice@mail.com`, собственным секретным и публичным ключами  $SK_A$  и  $PK_A$ . УЦ хранит публичные ключи всех пользователей и соответствующие сертификаты  $cert_A = sign_{cloud}(PK_A)$ . Каждое ДЭУ имеет предустановленное заводское ПО и заводской ключ для сброса к начальным настройкам. Предполагается наличие безопасного таймера и защищённого хранилища, которые считаются доверенными. На защищённую часть устройства возлагаются следующие функции:

1) Безопасный таймер/часы — предоставление доверенного значения времени. Без данного свойства невозможно отсле-

живание основного параметра делегации — времени.

2) Защищённое хранилище — необходимо для хранения конфиденциальной информации, ключей и т. п.

3) Доверенный функционал устройства — функции, связанные напрямую с информационной безопасностью, такие как, например, хеш-функции, генераторы псевдослучайных последовательностей и т. п.

Также в процессе функционирования протокола вводится ключ  $S_A$ , ассоциированный с пользователем Alice. Данный ключ позволяет обеспечить симметричное шифрование передаваемых и хранимых на ДЭУ данных, а также служит дополнительной защитой от утечки конфиденциальной информации пользователя Alice, хранимой на отдельно взятом ДЭУ. Дальнейшие шаги протокола обозначены соответствующими цифрами на рис. 1.

1. Пользователь Alice генерирует секретный ключ  $S_A$  для нового ДЭУ  $w_i$  и передаёт его по защищённому каналу.

2. ДЭУ  $w_i$  передаёт результат исполнения хеш-функции от собственного программного обеспечения на УЦ с использованием защищённого канала ( $hash(SW_i)$ ).

3. Пользователь Alice передаёт свой публичный ключ  $PK_A$  и идентификатор  $ID_A$  на УЦ.

4. УЦ генерирует сертификат УЦ

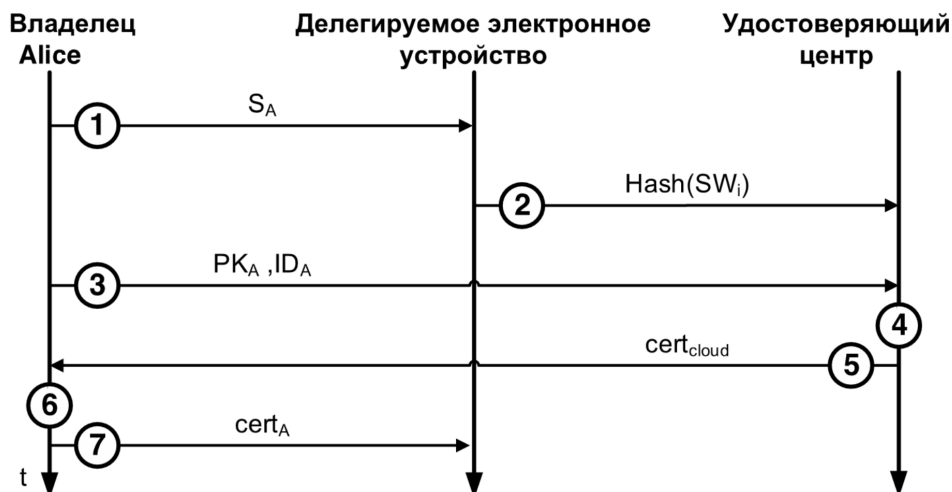


Рис. 1. Схема пошагового выполнения этапа ассоциации ДЭУ

Fig. 1. Association phase execution

для Alice. Сертификат имеет вид  $cert_{cloud} = sign_{cloud}(w_i, ID_A, hash(SW_i))$ . Данный шаг выполняется с целью обеспечения целостности системных данных.

5. УЦ, используя защищённый канал, передаёт  $cert_{cloud}$  пользователю Alice.

6. Пользователь Alice подписывает сертификат, полученный от УЦ, своим секретным ключом  $cert_A = sign_A(cert_{cloud})$ .

7. Пользователь Alice передаёт сертификат на ДЭУ.

**Передача устройства на некоторое время в пользование другому пользователю.** Функционирование системы может происходить в двух режимах: при наличии соединения с УЦ и при отсутствии соединения. В первом случае за аутентификацию полностью отвечает УЦ, соответственно рис. 1. В случае отсутствия стабильного соединения с УЦ протокол делегирования требует наличия прямого соединения между АУ пользователей и делегируемым ДЭУ. Шаги протокола, выполняемые на данном этапе, показаны на рис. 2 и 3, протоколы 2А и 2Б, соответственно. Данный сценарий описывает передачу электронного устройства в аренду с указанием условий и времени пользования ДЭУ.

*Протокол 2А — наличие стабильного соединения у обоих пользователей (рис. 2).*

1. Пользователь Alice устанавлива-

ет максимальное время делегирования ( $t_d$ ) для ДЭУ  $w_i$ , используя служебное сообщение, подписанное  $SK_A$ , как  $m[D]_A = sign_A(w_i, t_d, ID_A, ID_B, \{\text{прочие условия делегации}\})$ , согласно шагу 1.

2. Пользователь Alice передаёт  $m[D]_A$  на УЦ, используя защищённый канал, согласно шагу 2.

3. УЦ проверяет подлинность сообщения от Alice, используя  $PK_A$ . В случае непрохождения проверки протокол прекращает работу, согласно шагу 3.

4. УЦ подписывает сообщение делегирования  $m[D]_{cloud} = sign_{cloud}(m[D]_A)$ , согласно шагу 4.

5. УЦ передаёт  $m[D]_{cloud}$  и  $cert_A$  для Bob, в соответствии с шагом 5.

6. Пользователь Alice передаёт на ДЭУ служебное сообщение  $m[C(S_A)]_A$ , удаляющее секретный ключ  $S_A$  с ДЭУ, в соответствии с шагом 6.

7. Если пользователь Bob не доверяет Alice, протокол выполняет шаг 7. ДЭУ сбрасывается к заводским установкам. Сброс происходит с сохранением  $m[D]_{cloud}$  и сертификата  $cert_{cloud}$  в защищённом хранилище. Данные значения получены на шаге 6 протокола инициализации с целью сохранения целостности данных и подтверждения права собственности при отсутствии соединения с УЦ. АУ пользователя Bob сравни-

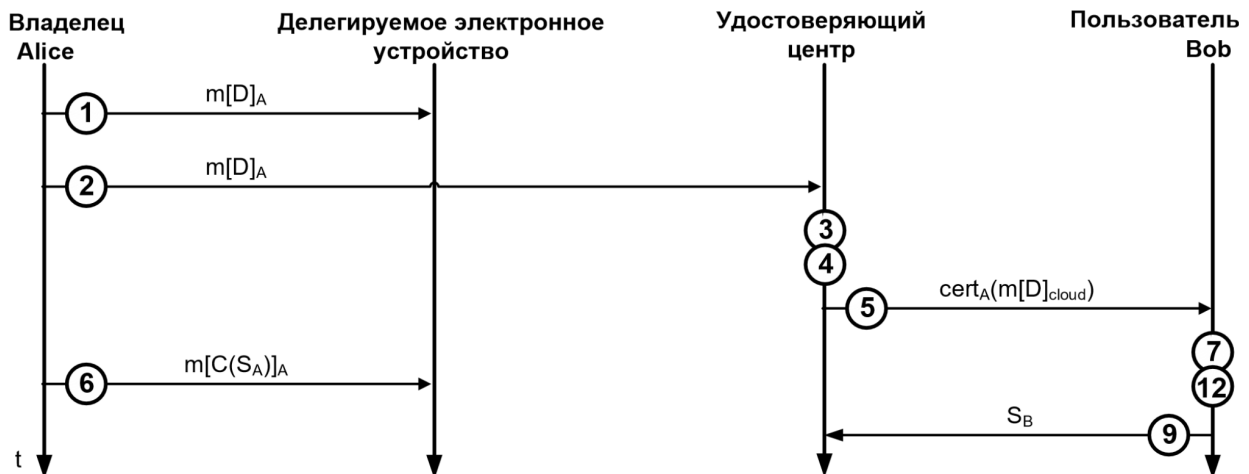


Рис. 2. Схема пошагового выполнения этапа делегирования ДЭУ при наличии стабильного соединения с УЦ

Fig. 2. Delegation phase execution: case of stable infrastructure connectivity



вает результат хеш-функции от текущего программного обеспечения ДЭУ с сохранённым в  $cert_{cloud}$ . Если они не совпадают, алгоритм прекращает исполнение. Таким образом пользователь Bob теряет возможность использовать устройство ДЭУ, т. к. предполагается, что ПО могло быть скомпрометировано владельцем. Важно отметить, что защищённые таймер и хранилище остаются неизменными даже при сбросе к заводским настройкам.

8. Если пользователь Bob **доверяет** Alice, программная компонента ДЭУ остается неизменной, и временный пользователь имеет возможность пользоваться программным обеспечением владельца устройства.

9. Пользователь Bob генерирует секретный ключ  $S_B$  для прямого взаимодействия с ДЭУ, согласно шагу 12 протокола.

10. Пользователь Bob передаёт  $S_B$  на ДЭУ по защищённому каналу, согласно шагу 9.

11. Для обеспечения целостности данных пользователь Bob рассчитывает новое значение  $sign_B(w_i, SW_i)$ .

12. В случае истечения таймера делегирования  $t_d$  на стороне ДЭУ, параметры устройства сбрасываются к заводским установкам с сохранением содержимого защищённого хранилища. Таймер может быть удалённо обновлён в случае одновременно наличия соединения с УЦ пользователя

и владельца при использовании служебного сообщения  $m[D]_A = sign_A(w_i, t_d, ID_A, ID_B, \{прочие условия делегации\})$ .

*Протокол 2B – отсутствие стабильного соединения хотя бы у одного из пользователей (рис. 3).*

1. Пользователь Alice устанавливает максимальное время делегирования  $t_d$  на ДЭУ  $w_i$ , используя служебное сообщение, подписанное  $SK_A$ , как  $m[D]_A = sign_A(w_i, t_d, ID_A, ID_B, \{прочие условия делегации\})$ , согласно шагу 1.

2. Пользователь Alice передаёт  $cert_A$  пользователю Bob по защищённому каналу, согласно шагу 2.

3. Bob проверяет подлинность  $cert_A$ , используя  $cert_{cloud}$ . В случае непрохождения проверки протокол прекращает работу, согласно шагу 3.

4. Пользователь Alice передаёт на ДЭУ служебное сообщение  $m[C(S_A)]_A$ , удаляющее секретный ключ  $S_A$  с ДЭУ, согласно шагу 4.

5. Если пользователь Bob **не доверяет** Alice, функционирование протокола выполняется в соответствии с шагом 5. ДЭУ сбрасывается к заводским установкам. Сброс происходит с сохранением  $m[D]_{cloud}$  и сертификата  $cert_{cloud}$  в защищённом хранилище. Данные значения были получены на шаге 6 протокола инициализации с целью сохранения целостности данных и под-

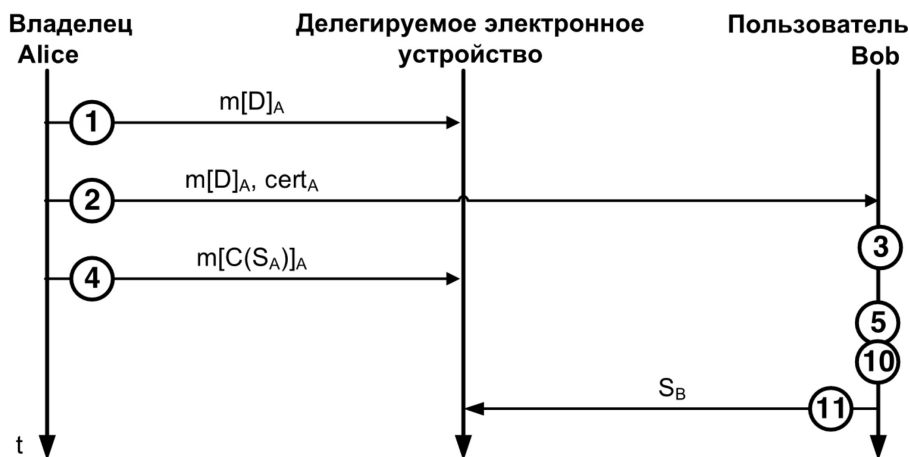


Рис. 3. Схема пошагового выполнения этапа делегирования ДЭУ при отсутствии стабильного соединения с УЦ

Fig. 3. Delegation phase execution: case of unreliable infrastructure connectivity

тверждения права собственности при отсутствии соединения с УЦ. АУ пользователя Bob сравнивает результат хеш-функции от текущего программного обеспечения ДЭУ с сохранённым в  $cert_{cloud}$ . Если они не совпадают, алгоритм прекращает исполнение. Таким образом, пользователь Bob теряет возможность использовать устройство ДЭУ, т. к. предполагается, что ПО могло быть скомпрометировано владельцем. Важно отметить, что защищённые таймер и хранилище остаются неизменными даже при сбросе к заводским настройкам.

6. Если пользователь Bob **доверяет** Alice, программная компонента ДЭУ остаётся неизменной, и временный пользователь имеет возможность пользоваться программным обеспечением владельца устройства.

7. Пользователь Bob генерирует секретный ключ  $S_B$  для прямого взаимодействия с ДЭУ, согласно шагу 10.

8. Пользователь Bob передаёт  $S_B$  на ДЭУ посредством защищённого канала, согласно шагу 11.

9. Для обеспечения целостности данных пользователь Bob рассчитывает новое значение  $sign_B(w_i, SW_i)$ .

10. В случае истечения времени делегирования  $t_d$  на стороне ДЭУ, параметры устройства сбрасываются к заводским установкам с сохранением содержимого защи-

щённого хранилища. Таймер может быть обновлён с использованием служебного сообщения  $m[D]_A = sign_A(w_i, t_d, ID_A, ID_B, \{\text{прочие условия делегирования}\})$  при наличии прямого соединения между пользователями.

11. Пользователь Bob передаёт  $S_B$  на ДЭУ по защищённому каналу, согласно шагу 9 на рис. 4.

12. Для обеспечения целостности данных пользователь Bob рассчитывает новое значение  $sign_B(w_i, SW_i)$ .

**Возврат устройства в пользование владельцу после временного использования ДЭУ другим пользователем.** Функционирование системы может происходить в двух режимах: при наличии соединения с УЦ и без такого соединения. Соответствующие шаги показаны на рис. 4 (протокол 3А) и 5 (протокол 3Б). В первом случае, аналогично делегированию, за аутентификацию полностью отвечает УЦ. В случае отсутствия стабильного соединения с УЦ алгоритм возврата устройства требует наличия прямых соединений между АУ пользователей и делегируемым ДЭУ.

*Протокол 3А – наличие стабильного соединения у обоих пользователей (рис. 4)*

1. Пользователь Bob генерирует служебное сообщение, подписанное  $SK_B$ , как  $m[R]_B = sign_A(w_i, R)$ , согласно шагу 1.

2. Пользователь Bob передаёт  $m[R]_B$  на

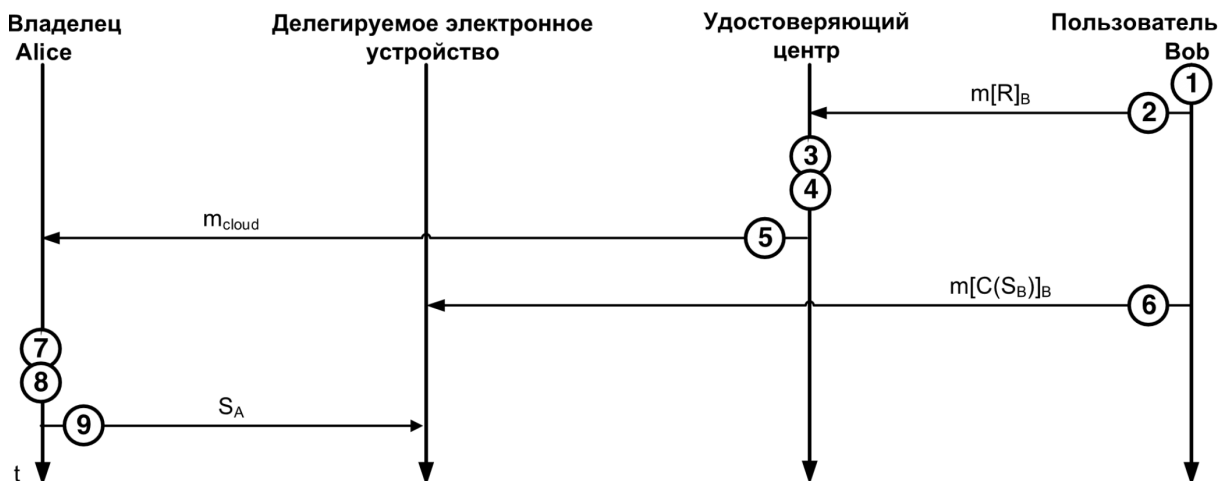


Рис. 4. Схема пошагового выполнения этапа возврата ДЭУ при наличии стабильного соединения с УЦ

Fig. 4. Reclaiming phase execution: case of stable infrastructure connectivity

УЦ, используя защищённый канал, согласно шагу 2.

3. УЦ проверяет подлинность сообщения от Bob с использованием  $PK_B$ . В случае непрохождения проверки протокол прекращает работу, согласно шагу 3.

4. УЦ подписывает сообщение возврата  $m[R]_{cloud} = sign_{cloud}(m[R]_B)$ , согласно шагу 4.

5. УЦ передаёт  $m[R]_{cloud}$  пользователю Alice, согласно шагу 5.

6. Пользователь Bob передаёт на ДЭУ служебное сообщение  $m[C(S_B)]_B$ , удаляющее секретный ключ  $S_B$  с ДЭУ, согласно шагу 6.

7. Если пользователь Alice **не доверяет** Bob, выполняется шаг 7. ДЭУ сбрасывается к заводским установкам с сохранением данных в защищённом хранилище. АУ пользователя Alice сравнивает результат хеш-функции от текущего программного обеспечения ДЭУ с сохранённым в  $cert_{cloud}$ . Если они не совпадают, протокол прекращает свою работу. Таким образом, пользователь Alice теряет возможность использовать устройство ДЭУ, т. к. предполагается, что ПО могло быть скомпрометировано владельцем. Важно отметить, что защищённые таймер и хранилище остаются неизменными даже при сбросе к заводским настройкам.

8. Если пользователь Alice **доверяет** Bob, программная компонента ДЭУ остаётся неизменной, а владелец устройства имеет возможность пользоваться программным обеспечением, установленным в процессе делегирования.

9. Пользователь Alice генерирует секретный ключ  $S_A$  для прямого взаимодействия с ДЭУ, согласно шагу 8.

10. Пользователь Alice передаёт  $S_A$  на ДЭУ, используя защищённый канал, согласно шагу 9.

11. Для обеспечения целостности данных пользователь Alice рассчитывает новое значение  $sign_A(w_i, SW_i)$ . В защищённом хранилище ДЭУ находятся только  $cert_A$  и  $cert_{cloud}$ .

*Протокол 3В – отсутствие стабильного соединения хотя бы у одного из пользователей (рис. 5)*

1. Пользователь Bob генерирует служебное сообщение, подписанное  $SK_B$ , как  $m[R]_B = sign_B(w_i, R)$ , согласно шагу 4.

2. Пользователь Bob передаёт  $m[R]_B$  пользователю Alice по защищённому каналу, согласно шагу 2.

3. Alice проверяет подлинность  $m[R]_B$ , используя  $cert_{cloud}$ . В случае непрохождения проверки протокол прекращает свою работу, согласно шагу 3.

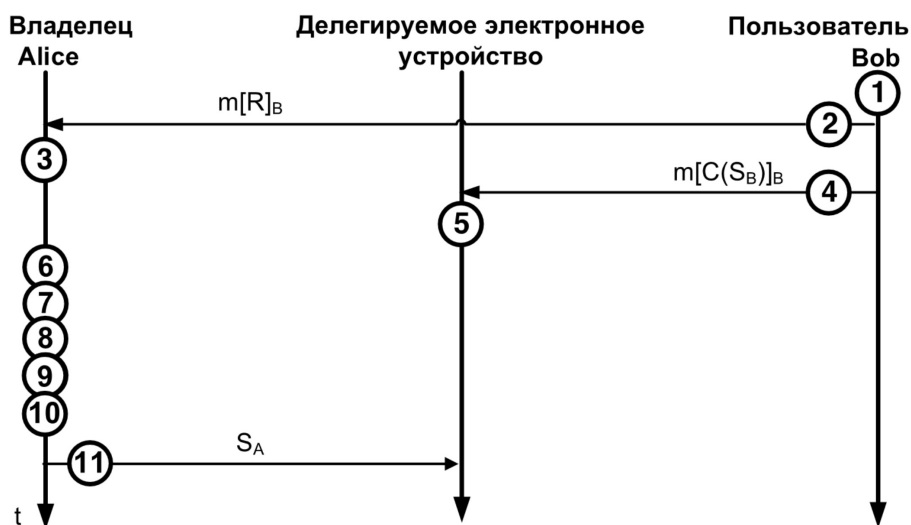


Рис. 5. Схема пошагового выполнения этапа возврата ДЭУ при отсутствии стабильного соединения с УЦ

Fig. 5. Reclaiming phase execution: case of unreliable infrastructure connectivity

4. Пользователь Bob передаёт на ДЭУ служебное сообщение  $m[C(S_B)]_B$ , удаляющее секретный ключ  $S_B$  с ДЭУ, согласно шагу 4.

5. Если пользователь Alice **не доверяет** Bob, функционирование протокола происходит согласно шагу 5. ДЭУ сбрасывается к заводским установкам с сохранением данных в защищённом хранилище. АУ пользователя Alice сравнивает результат хеш-функции от текущего программного обеспечения ДЭУ с сохранённым в  $cert_{cloud}$ . Если они не совпадают, протокол прекращает свою работу. Таким образом, пользователь Alice теряет возможность использовать устройство ДЭУ, т. к. предполагается, что ПО могло быть скомпрометировано владельцем. Важно отметить, что защищённые таймер и хранилище остаются неизменными даже при сбросе к заводским настройкам.

6. Если пользователь Alice **доверяет** Bob, программная компонента ДЭУ остаётся неизменной, а владелец имеет возможность пользоваться программным обеспечением временного пользователя устройства.

7. Пользователь Alice генерирует секретный ключ  $S_A$  для прямого взаимодействия с ДЭУ, согласно шагу 10.

8. Пользователь Alice передаёт  $S_A$  на ДЭУ по защищённому каналу, согласно шагу 11.

9. Для обеспечения целостности данных пользователь Alice рассчитывает новое значение  $sign_A(w_i, SW_i)$ . В защищённом хранилище ДЭУ находятся только  $cert_A$  и  $cert_{cloud}$ .

**Диссоциация устройства.** В данном случае возможны два варианта функционирования протокола: ручная и автоматическая диссоциация (сброс настроек устройства к заводским). При ручной диссоциации владелец и ДЭУ связываются по прямому каналу. АУ владельца передаёт управляющее сообщение, сбрасывающее ДЭУ к состоянию заводских настроек. Автоматическая диссоциация возможна по предустановленному таймеру, который производит аналогичную процедуру по истечению времени делегирования устройства в случае его невозврата.

*Протокол 4А – ручная диссоциация ДЭУ*

1. Владелец Alice генерирует служебное сообщение, подписанное  $SK_A$ , как  $m[F]_A = sign_A(w_i, F)$ .

2. Пользователь Alice передаёт  $m[R]_B$  на ДЭУ по защищённому каналу.

3. ДЭУ сбрасывается к заводским настройкам с очисткой защищённого хранилища.

4. Устройство ДЭУ может быть восстановлено только при использовании заводского ключа (например, PIN) и при наличии соединения с УЦ.

*Протокол 4В – автоматическая диссоциация ДЭУ*

Данный протокол выполняется в случае, когда устройство не было возвращено согласно условиям делегирования, утеряно или украдено. Все личные данные владельца и пользователя должны быть уничтожены с целью предотвращения потенциального вредоносного использования.

1. В процессе начальной ассоциации устройства владелец Alice опционально генерирует служебное сообщение, подписанное  $SK_A$ , как  $m[E]_A = sign_A(w_i, t_e)$ , где  $t_e$  является значением таймера обратного отсчёта автоматической диссоциации. ДЭУ обращается к таймеру в процессе эксплуатации с целью обнаружения потенциальной переработки.

2. В случае переработки ДЭУ сбрасывается к заводским настройкам, очищая защищённое хранилище.

3. Устройство ДЭУ может быть восстановлено только при использовании заводского ключа (например, PIN) и при наличии соединения с УЦ.

### Возможные атаки на протокол

После детального рассмотрения предложенного протокола важно отметить потенциальные атаки на его работу. Одной из таких атак является фишинг. В данном случае злоумышленник Eve может представиться доверенным пользователем Bob с  $ID_B$  в процессе делегации между Alice и Bob. Если Alice не может проверить подлинность источника запроса, атака считается успешной. Основная область применения атаки

находится в сфере носимой электроники, т. к. устройства данного типа часто не обладают дополнительным визуальным каналом (например, дисплеем) для дополнительного подтверждения процедуры делегации. В современном мире не существует решения, полностью защищающего от фишинг атак, однако методики многофакторной аутентификации позволяют понизить вероятность успеха атаки.

Другой немаловажной атакой на протокол является классическая атака посредника. В нашем случае злоумышленник Eve запрашивает  $m[D]_A$  для Bob ( $ID_B$ ) от Alice, представляясь Alice ( $ID_A$ ), и одновременно передаёт  $m[D]_A$  для пользователя Bob. В результате Eve не получает возможности использования устройства, однако, может следить за процессом делегирования.

Одной из интереснейших атак является атака с использованием заражённого устройства. Рассмотрим следующий пример. После перехвата злоумышленником корректного  $m[D]_A$  о делегировании устройства  $w_k$ , Eve создаёт заражённое устройство, которое представляется  $w_k$  и передаёт всегда корректное значение  $hash(SW_k)$ . Такое устройство позволяет, например, следить за активностью пользователя Bob.

В то же время протоколы, использующие цифровую подпись, особенно уязвимы с точки зрения конфиденциальности, поскольку они, как правило, вовлекают

однозначное подтверждение со стороны доверенной сущности. Иными словами, пользователь не может позднее отречься от факта делегирования. Протоколы, основанные на доказательствах с нулевым разглашением, полагаются на это свойство для повышения безопасности, однако, вызывают дополнительные осложнения в области конфиденциальности.

### Заключение

Представлен протокол аутентификации и контроля электронных устройств, разработанный для использования в условиях нестабильной связи с удостоверяющим центром. Алгоритмы, реализующие этапы функционирования предлагаемого протокола, могут быть реализованы как в виде программного обеспечения для универсального компьютера любой архитектуры, так и в виде аппаратного обеспечения для специализированного компьютера любой архитектуры. Протокол может использоваться в местах с отсутствием инфраструктуры, т. к. для реализации делегирования не требуется постоянного наличия соединения с УЦ.

Работа выполнена при поддержке Фонда содействия развитию малых форм предприятий в научно-технической сфере в рамках программы «УМНИК» по договору № 8268ГУ2015 от 02.12.2015 г.

### СПИСОК ЛИТЕРАТУРЫ

1. Kim H.J., Chang H.S., Suh J.J., Shon T.S. A study on device security in IoT convergence // Proc. of Internat. Conf. on Industrial Engineering, Management Science and Application. IEEE, 2016. Pp. 1–4.
2. Saadeh M., Sleit A., Qatawneh M., Almobaideen W. August. Authentication techniques for the Internet of Things: A survey // Proc. of Cybersecurity and Cyberforensics Conference. IEEE, 2016. Pp. 28–34.
3. Nix J.A. M2M and IoT technologies, Llc. Systems and methods for “machine-to-machine” (M2M) communications between modules, servers, and an application using Public Key Infrastructure (PKI) // U.S. Patent 9,276,740, 2016.
4. Sicari S., Rizzardi A., Grieco L.A., Coen-Porisini A. Security, privacy and trust in Internet of Things: The road ahead // Computer Networks. 2015. No. 76. Pp.146–164.
5. Andreev S., Galinina O., Pyattaev A., Gerasimenko M., Tirronen T., Torsner J., Sachs J., Dohler M., Koucheryavy Y. Understanding the IoT connectivity landscape: a contemporary M2M radio technology roadmap // IEEE Communications Magazine. 2015. Vol. 53. No. 9. Pp.32–40.
6. Arias O., Wurm J., Hoang K., Jin Y. Privacy and security in Internet of Things and wearable devices // IEEE Transactions on Multi-Scale Computing Systems. 2015. Vol. 1. No. 2. Pp. 99–109.
7. Wei J. How wearables intersect with the cloud and the Internet of Things: Considerations for the developers of wearables // IEEE Consumer Electronics Magazine. 2014. Vol. 3. No. 3. Pp. 53–56.



8. **Simon L., Anderson R.** Security analysis of android factory resets // *Proc. of 4th Mobile Security Technologies Workshop*. 2015.

9. **Abrardo A., Fodor G., Tola B.** Network coding schemes for device-to-device communications based relaying for cellular coverage extension // *Proc. of 16th Internat. Workshop on Signal Processing Advances in Wireless Communications*. IEEE, 2015. Pp. 670–674.

10. **Ometov A., Zhidanov K., Bezzateev S., Florea R., Andreev S., Koucheryavy Y.** Securing network-assisted direct communication: The case of unreliable cellular connectivity // *Proc. of Trustcom/BigDataSE/ISPA*. IEEE, 2015. Vol. 1. Pp. 826–833.

11. **Hummen R., Ziegeldorf J.H., Shafagh H., Raza S., Wehrle K.** Towards viable certificate-based authentication for the Internet of Things // *Proc. of the 2nd ACM Workshop on Hot Topics on Wireless Network Security and Privacy*. ACM, 2013. Pp. 37–42.

12. **Lee Y.K., Batina L., Verbaudhede I.** Untraceable RFID authentication protocols: Revision of EC-RAC // *Proc. of Internat. Conf. on RFID*. IEEE, 2007. Pp. 178–185.

13. **Aumasson J.P., Neves S., Wilcox-O’Hearn Z.,**

**Winnerlein C.** BLAKE2: simpler, smaller, fast as MD5 // *Proc. of Internat. Conf. on Applied Cryptography and Network Security*. Springer, 2013. Pp. 119–135.

14. **Wander A.S., Gura N., Eberle H., Gupta V., Shantz S.C.** Energy analysis of public-key cryptography for wireless sensor networks // *Proc. of 3rd IEEE Internat. Conf. on Pervasive Computing and Communications*. IEEE, 2015. Pp. 324–328.

15. **Elgamal T.** A public key cryptosystem and a signature scheme based on discrete logarithms // *IEEE Transactions on Information Theory*. 1985. Vol. 31. No. 4. Pp.469–472.

16. **He D., Zeadally S.** An analysis of rfid authentication schemes for Internet of Things in healthcare environment using elliptic curve cryptography // *IEEE Internet of Things Journal*. 2015. Vol. 2. No. 1. Pp.72–83.

17. **Diffie W., Hellman M.** New directions in cryptography // *IEEE Transactions on Information Theory*. 1976. Vol. 22. No. 6. Pp. 644–654.

18. **Joux A.** A one round protocol for tripartite Diffie–Hellman // *International Algorithmic Number Theory Symp*. Springer, 2000. Pp. 385–393.

*Статья поступила в редакцию 10.07.2017.*

## REFERENCES

1. **Kim H.J., Chang H.S., Suh J.J., Shon T.S.** A study on device security in IoT convergence. *Proceedings of International Conference on Industrial Engineering, Management Science and Application*, IEEE, 2016, Pp. 1–4.

2. **Saadeh M., Sleit A., Qatawneh M., Almobaideen W.** August. Authentication techniques for the Internet of Things: A survey. *Proceedings of Cybersecurity and Cyberforensics Conference*, IEEE, 2016, Pp. 28–34.

3. **Nix J.A.** M2M and IoT Technologies, Llc. Systems and methods for “machine-to-machine” (M2M) communications between modules, servers, and an application using Public Key Infrastructure (PKI). *U.S. Patent 9,276,740*, 2016.

4. **Sicari S., Rizzardi A., Grieco L.A., Coen-Porisini A.** Security, privacy and trust in Internet of Things: The road ahead. *Computer Networks*, 2015, No. 76, Pp.146–164.

5. **Andreev S., Galinina O., Pyattaev A., Gerasimenko M., Tirronen T., Torsner J., Sachs J., Dohler M., Koucheryavy Y.** Understanding the IoT connectivity landscape: a contemporary M2M radio technology roadmap. *IEEE Communications Magazine*, 2015, Vol. 53, No. 9, Pp. 32–40.

6. **Arias O., Wurm J., Hoang K., Jin Y.** Privacy

and security in Internet of Things and wearable devices. *IEEE Transactions on Multi-Scale Computing Systems*, 2015, Vol. 1, No. 2, Pp. 99–109.

7. **Wei J.** How wearables intersect with the cloud and the Internet of Things: Considerations for the developers of wearables. *IEEE Consumer Electronics Magazine*, 2014, Vol. 3, No. 3, Pp.5 3–56.

8. **Simon L., Anderson R.** Security analysis of android factory resets. *Proceedings of 4th Mobile Security Technologies Workshop*, 2015.

9. **Abrardo A., Fodor G., Tola B.** Network coding schemes for device-to-device communications based relaying for cellular coverage extension. *Proceedings of 16th International Workshop on Signal Processing Advances in Wireless Communications*, IEEE, 2015, Pp. 670–674.

10. **Ometov A., Zhidanov K., Bezzateev S., Florea R., Andreev S., Koucheryavy Y.** Securing network-assisted direct communication: The case of unreliable cellular connectivity. *Proceedings of Trustcom/BigDataSE/ISPA*, IEEE, 2015, Vol. 1, Pp. 826–833.

11. **Hummen R., Ziegeldorf J.H., Shafagh H., Raza S., Wehrle K.** Towards viable certificate-based authentication for the Internet of Things.

*Proceedings of the 2nd ACM Workshop on Hot Topics on Wireless Network Security and Privacy*, ACM, 2013, Pp. 37–42.

12. **Lee Y.K., Batina L., Verbauwhede I.** Untraceable RFID authentication protocols: Revision of EC-RAC. *Proceedings of International Conference on RFID*, IEEE, 2007, Pp. 178–185.

13. **Aumasson J.P., Neves S., Wilcox-O’Hearn Z., Winnerlein C.** BLAKE2: simpler, smaller, fast as MD5. *Proceedings of International Conference on Applied Cryptography and Network Security*, Springer, 2013, Pp. 119–135.

14. **Wander A.S., Gura N., Eberle H., Gupta V., Shantz S.C.** Energy analysis of public-key cryptography for wireless sensor networks. *Proceedings of 3rd IEEE International Conference on*

*Pervasive Computing and Communications*, IEEE, 2015, Pp. 324–328.

15. **Elgamal T.** A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Transactions on Information Theory*, 1985, Vol. 31, No. 4, Pp. 469–472.

16. **He D., Zeadally S.** An analysis of rfid authentication schemes for Internet of Things in healthcare environment using elliptic curve cryptography. *IEEE Internet of Things Journal*, 2015, Vol. 2, No. 1, Pp. 72–83.

17. **Diffie W., Hellman M.** New directions in cryptography. *IEEE Transactions on Information Theory*, 1976, Vol. 22, No. 6, Pp. 644–654.

18. **Joux A.** A one round protocol for tripartite Diffie–Hellman. *International Algorithmic Number Theory Symp.*, Springer, 2000, Pp. 385–393.

Received 10.07.2017.

#### СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

**ОМЁТОВ Александр Ярославич**

**OMETOV Aleksander Ya.**

E-mail: alexander.ometov@gmail.com

**БЕЗЗАТЕЕВ Сергей Валентинович**

**BEZZATEEV Sergey V.**

E-mail: bsv@aanet.ru

**КУЧЕРЯВЫЙ Евгений Андреевич**

**KOUCHERYAVY Yevgeni A.**

E-mail: ykoucheryavy@hse.ru

## **ОСОБЕННОСТИ РАЗРАБОТКИ СВЕРХШИРОКОПОЛОСНЫХ ПЕРЕСТРАИВАЕМЫХ ГЕНЕРАТОРОВ СВЧ ДИАПАЗОНА В ГИБРИДНОМ ИСПОЛНЕНИИ**

*А.Б. Никитин, Е.И. Хабитуева*

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

Рассмотрены особенности разработки перестраиваемых в широкой полосе частот (октава и более) транзисторных СВЧ-генераторов, управляемых напряжением (ГУН). В результате проведенных исследований сверхширокополосного ГУН с октавной перестройкой частоты в диапазоне от 6 до 12 ГГц, выполненного по гибридной технологии, определена базовая структура генератора и сформулированы требования к его управляющим элементам. Развита методика разработки сверхширокополосных ГУН в части создания специальных выходных согласующе-трансформирующих цепей, обеспечивающих непрерывную перестройку в рассматриваемом диапазоне частот. Разработан микрополосковый вариант схемы ГУН, выполненный в рамках гибридной технологии на базе стандартных промышленно выпускаемых электронных компонентов. Реализованная топология генератора по основным своим характеристикам не уступает устройствам зарубежного производства.

**Ключевые слова:** ГУН; варикап; широкополосная перестройка частоты; выходная цепь; микрополосковая схема.

**Ссылка при цитировании:** Никитин А.Б., Хабитуева Е.И. Особенности разработки сверхширокополосных перестраиваемых генераторов СВЧ диапазона в гибридном исполнении // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2017. Т. 10. № 4. С. 41–50. DOI: 10.18721/JCSTCS.10404

## **THE FEATURES OF DESIGN OF AN ULTRA-WIDEBAND MICROWAVE HYBRID VCO**

*A.B. Nikitin, E.I. Khabitueva*

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

Some features of designing an ultra-wideband microwave voltage-controlled oscillator (VCO) are reviewed in this article. The developed microstrip VCO is realized by means of hybrid technology. It has achieved an oscillation bandwidth of 6–12 GHz. As a result of studying the ultra-wideband VCO, we have determined the basic structure of the generator and formulated the requirements for its control elements.

We have developed a method for designing the microwave VCO's output-matching circuits. Using these output circuits allows achieving ultra-wideband characteristics. As result, a hybrid microstrip circuit of the ultra-wideband VCO has been designed. Standard SMD-components are used in the developed VCO circuit. The VCO's main parameters are not inferior to similar products of the leading manufacturers of microwave devices.

**Keywords:** VCO; varactor diode; ultra-wide band; output circuit; microstrip circuit.

**Citation:** Nikitin A.B., Khabitueva E.I. The features of design of an ultra-wideband microwave hybrid VCO. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2017, Vol. 10, No. 4, Pp. 41–50. DOI: 10.18721/JCSTCS.10404

## Введение

Одним из ключевых элементов синтезаторов частот (СЧ) с системой фазовой автоподстройки частоты (ФАПЧ) является генератор, управляемый напряжением (ГУН), параметры которого в существенной степени определяют характеристики всего устройства [1, 2]. Использование в качестве таких ГУН сверхширокополосных перестраиваемых СВЧ-генераторов (с полосой перестройки октава и более) дает возможность построения быстродействующих широкодиапазонных источников стабильных колебаний [3–5].

Разработкой подобных перестраиваемых генераторов в настоящее время занимаются несколько ведущих производителей СВЧ-устройств. Однако перестройку частоты в пределах октавы в высокочастотной части сантиметрового диапазона (например, в полосе от 6 до 12 ГГц) обеспечивают лишь единичные образцы ГУН, среди которых можно выделить интегральные микросхемы генераторов HMC732LC4B фирмы Analog Devices [6] и MAOC-409000 компании MACOM [7].

Примеры схем аналогичных ГУН в коротковолновой части сантиметрового диапазона волн, выполненных на основе более доступной гибридной технологии с использованием серийно выпускаемых электронных компонентов, в научно-технической литературе практически полностью отсутствуют. Из-за трудностей реализации на высоких частотах устройств в гибридном исполнении, обусловленных ограничениями технологии, налагаемыми на минимальные размеры используемых компонентов, примеры разработки сверхширокополосных ГУН СВЧ в имеющейся научно-технической

литературе ограничиваются лишь нижней частью сантиметрового и дециметровым диапазонами [8, 9]. Принимая во внимание данное обстоятельство, а также учитывая действующие ограничения на поставку в Россию радиоэлектронных устройств и систем двойного назначения [10], практический интерес представляет исследование путей построения сверхширокополосного СВЧ-генератора, обеспечивающего перестройку частоты в пределах октавы (и более) в коротковолновой части сантиметрового диапазона волн и выполненного в рамках доступной гибридной технологии.

Важнейшим этапом разработки подобных устройств является их компьютерное моделирование, необходимое для анализа рассматриваемых схем генератора и создания итоговой топологии устройства. При этом результативность проведенного проектирования в существенной степени определяется адекватностью моделей используемых электронных компонентов. Специфика исследуемого сантиметрового диапазона волн, ограниченные возможности гибридной технологии микросхем, а также сверхширокие (октавные) полосы перестройки выдвигают повышенные требования к точности моделей, используемых на этапе выбора и обоснования схемы ГУН, анализа его основных характеристик и обработки топологии.

Следует отметить, что для целого ряда типов и номиналов электронных компонентов предоставляемая производителями информация носит весьма ограниченный характер либо отсутствует вовсе. Вследствие этого процесс разработки микрополоскового ГУН осложняется еще и необходимостью проведения серии дополнительных экспериментальных исследований [11–14].

В данной статье рассматриваются особенности разработки и реализации сверхширокополосного микрополоскового ГУН на основе SiGe-транзистора, перестраиваемого в полосе частот от 6 до 12 ГГц.

### Выбор базовой схемы ГУН

Для выработки требований к количеству управляющих элементов, их параметрам и способам включения, а также для сравнения различных типов активных элементов (АЭ) генератора за основу при анализе была выбрана схема с последовательной обратной связью и нагрузкой в коллекторе [15]. Реактивные двухполюсники  $Z_B = jX_B$  и  $Z_E = jX_E$  соответственно в базовой и эмиттерной цепях транзистора обладают необходимыми для возникновения генерации сопротивлениями, обеспечивающими отрицательное сопротивление на выходном порту АЭ ( $\text{Re}(Z_{Ac}) < 0$ ), а двухполюсник  $Z_C$  в коллекторной цепи моделирует нагрузку с необходимыми согласующими устройствами (рис. 1 а). При этом  $Z_B$ ,  $Z_E$  и  $Z_C$  могут включать в себя как по отдельности, так и одновременно управляющие элементы — варикапы.

На первом этапе работы осуществлялось исследование АЭ автогенератора (АГ) в линейном режиме. Такой подход позволяет выбрать рациональную структуру АГ, найти начальные значения требуемых для возникновения автоколебаний реактивностей  $X_B$  и  $X_E$ , выработать требования к параметрам варикапов, а также в первом приближении оценить диапазон перестройки частоты. При этом рассматривались импедансные

характеристики со стороны коллекторных выводов транзистора.

Для обеспечения режима генерации необходимо, чтобы во всем диапазоне частот выполнялись следующие условия [15]:

$$\begin{cases} \text{Re}(Z_C) + \text{Re}(Z_{Ac}) = 0 \\ \text{Im}(Z_C) + \text{Im}(Z_{Ac}) = 0. \end{cases} \quad (1)$$

Полагая, что при использовании резистивной нагрузки (т. е. при условии  $\text{Im}(Z_C) = 0$ ) первое равенство системы (1) может быть выполнено соответствующим выбором  $\text{Re}(Z_C)$ , а именно при  $\text{Re}(Z_C) < |\text{Re}(Z_{Ac})|$ , можно сформулировать условия, позволяющие в линейном режиме приближенно оценить возможную частоту генерации  $f_0$ , а при изменении емкости варикапа — диапазон перестройки частоты автогенератора:

$$\begin{cases} \text{Re}(Z_{Ac}(f_0)) < 0 \\ \text{Im}(Z_{Ac}(f_0)) = 0, \end{cases} \quad (2)$$

$$\text{Im}(Z_{Ac}(f_0)) = 0, \quad (3)$$

где  $Z_{Ac}$  — импеданс АЭ автогенератора со стороны его коллекторных выводов (см. рис. 1 а). При этом на ожидаемой частоте автоколебаний  $f_0$  должно выполняться условие  $\partial \text{Im}(Z_{Ac}(f_0)) / \partial f > 0$ .

Использование соотношений (2), (3) позволяет в линейном режиме провести сравнение различных схем ГУН, выбрать исходные варианты построения управляющих цепей, способов их включения в схему генератора, а также выработать предварительные требования к варикапам. Проведенный анализ линейного режима помог определить возможные варианты базовых структур ГУН, характеристики которых в дальнейшем исследовались в стационарном режиме работы генератора. В результате было показано, что для достижения октавных полос перестройки в рассматриваемом частотном диапазоне целесообразно использовать схему ГУН с двумя управляющими элементами в базе и эмиттере транзистора. Данный вариант построения схемы генератора превосходит вариант с одним управляющим элементом по требуемому коэффициенту перекрытия по емкости варикапа  $k = C_{V\max}/C_{V\min}$ . Так, например, при моделировании варикапа линейной изменяемой емкостью необходимая величина  $k$  в двухдиодной схеме оказывается менее 4,8, а

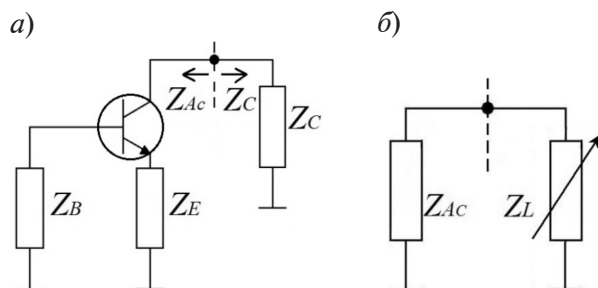


Рис. 1. Упрощенная эквивалентная схема ГУН

Fig. 1. Simplified equivalent circuit of VCO



в схеме с одним варикапом  $k \approx 5,5$ . В то же время выбранный вариант практически не уступает структуре с тремя варикапами [16], оказываясь существенно проще ее в практической реализации.

Моделирование автогенератора в стационарном режиме позволило определить необходимый тип варикапов, параметры которого дают возможность добиться требуемых перестроечных характеристик устройства. Среди весьма ограниченного набора СВЧ-диодов, обладающих достаточными значениями коэффициента перекрытия по емкости  $k$ , можно выделить варикапы серии МА46, выпускаемые компанией MACOM [18]. В ряду диодов этой серии имеются высокодобротные ( $Q = 3000$ ) GaAs-варикапы с т. н. «гиперрезким» переходом, выпускаемые в виде удобных для монтажа в гибридных микросхемах flip-chip компонентов и обладающие требуемыми значениями коэффициента перекрытия ( $k \approx 6...7$ ) и минимальной емкости ( $C_{\nu\min} < 0,15$  пФ).

Таким образом, на основе проведенных исследований в качестве базовой схемы ГУН была выбрана структура с нагрузкой в коллекторе и двумя перестраиваемыми цепями, построенными на основе варикапов серии МА46, включаемых в эмиттерную и базовую цепи транзистора (двухполюсники  $Z_B$  и  $Z_E$  на рис. 1 а).

#### Разработка микрополосковой схемы генератора

Результаты анализа стационарно-

го режима ГУН, проведенного на основе его упрощенной эквивалентной схемы (рис. 1 а), позволили приступить к разработке схемы генератора на базе модели, учитывающей влияние отрезков микрополосковой линии (МПЛ) и «паразитных» параметров используемых SMD-компонентов. Построение такой модели подразумевает использование достоверных экспериментальных данных о параметрах всех компонентов ГУН или их эквивалентных схем, адекватным образом описывающих характеристики этих компонентов в рассматриваемом частотном диапазоне. Однако вследствие того, что характеристики значительной части используемых электронных компонентов генератора отсутствуют в свободном доступе, а имеющиеся в наличии часто требуют своего экспериментального уточнения, для построения адекватной модели микрополоскового ГУН потребовалось проведение большого объема измерений в сантиметровом диапазоне волн частотных характеристик целого ряда SMD-компонентов: конденсаторов, резисторов, катушек индуктивностей, варикапов [13, 14].

На рис. 2 в качестве примера приведено семейство частотных зависимостей мнимой составляющей  $\text{Im}(Z_\nu)$  полного сопротивления варикапа МА46Н120 для нескольких значений управляющего напряжения  $U_\nu$ , полученное на основе измерений S-параметров диода, включенного в разрыв МПЛ.

На основе данного и аналогичных из-

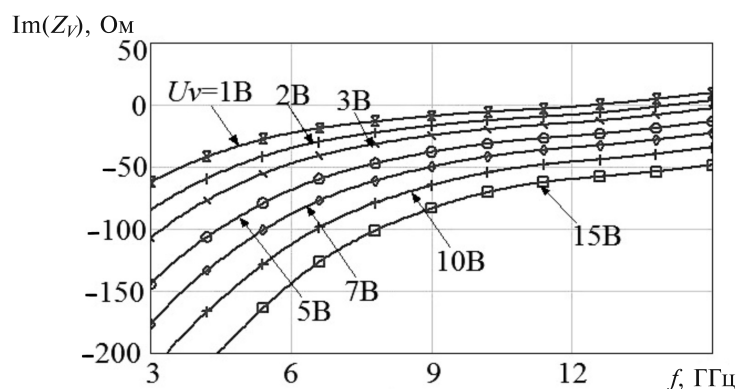


Рис. 2. Частотные зависимости мнимой составляющей сопротивления варикапа

Fig. 2. Imaginary part of varicap impedance vs frequency

мерений импедансных характеристик других элементов были внесены существенные коррективы в модели ряда компонентов ГУН, а также разработаны эквивалентные схемы для компонентов, их не имеющих.

Исследование стационарного режима ГУН на основе его модели, отражающей особенности реализуемой топологии микрополосковой схемы, а также учитывающей результаты экспериментальных исследований SMD-компонентов, показало, что внесенные в модель ГУН коррективы существенно затрудняют достижение необходимых характеристик устройства. Так, уже на этапе линейного анализа реализация условий (2) и (3) оказалась весьма затруднительна, поскольку указанное уточнение модели привело к существенному уменьшению модуля отрицательного сопротивления АЭ ГУН ( $|\operatorname{Re}(Z_{Ac})|$ ), а также к увеличению нелинейности частотной зависимости его импеданса. В результате получение отрицательного вещественного сопротивления  $\operatorname{Re}(Z_{Ac})$  при одновременном достижении условия (3) во всем требуемом частотном диапазоне оказалось невозможным для физически реализуемых параметров схемы генератора. Следствием этого стало нарушение непрерывности перестройки ГУН в рассматриваемом диапазоне частот.

Для решения задачи разработки микрополосковой схемы генератора, обладающего непрерывной перестройкой, проведено исследование влияния нагрузки на характеристики стационарного режима. Целью такого исследования являлось построение выходной согласующе-трансформирующей цепи (СТЦ), использование которой в схеме ГУН обеспечивает выполнение условий генерации (1) в требуемом частотном диапазоне.

На рис. 1 б приведена упрощенная эквивалентная схема генератора, на основе которой проводилось указанное выше исследование. В данной схеме символами  $Z_{Ac}$  и  $Z_L$  обозначены импедансы соответственно АЭ генератора со стороны коллекторных выводов и его нагрузки. При этом двухполюсник  $Z_{Ac}$  включает в себя полную эквивалентную схему ГУН, учитывающую

особенности его микрополосковой топологии, а двухполюсник  $Z_L$  моделирует варьруемую комплексную (в общем случае) нагрузку генератора, представляемую в виде идеализированных  $R$ -,  $RL$ - или  $RC$ -цепей, в которых можно произвольным образом изменять как вещественную составляющую полного сопротивления  $Z_L$ , так и величину и знак ее мнимой части с помощью реактивностей  $L$  и  $C$ .

Для определения требуемой частотной характеристики выходной цепи ГУН использовалась следующая методика компьютерного эксперимента. В каждой точке диапазона изменения управляющего напряжения  $U_{\nu}$ , подаваемого на варикапы, путем варьирования значений  $R$ -,  $L$ -,  $C$ -элементов двухполюсника  $Z_L$  (рис. 1 б) осуществлялся поиск параметров нагрузочной цепи ГУН, обеспечивающей решение уравнений автоколебаний (1). В результате многократного анализа стационарного режима работы схемы ГУН, включающей в себя идеализированную варьруемую нагрузку  $Z_L$  (см. рис. 1 б), формировались требуемые импедансные характеристики выходной цепи, обеспечивающей возможность непрерывной перестройки частоты в исследуемом диапазоне от 6 до 12 ГГц.

В результате это позволило оценить необходимые импедансные характеристики АЭ ГУН, обеспечивающие непрерывную перестройку частоты в диапазоне от 6 до 12 ГГц. Было установлено, что частотные зависимости импеданса выходной согласующей цепи ГУН должны обладать сложным и существенно немонотонным характером.

Так, например, при использовании чисто вещественной нагрузки получение удовлетворительных характеристик перестройки во всем диапазоне частот от 6 до 12 ГГц оказывается весьма затруднительно вследствие существенного (в несколько раз) изменения величины вещественной составляющей импеданса АЭ в рассматриваемом диапазоне частот. На рис. 3 приведены зависимости частоты  $f$  и мощности  $P$  автоколебаний ГУН от активной части нагрузки  $R$  ( $Z_L = R$ , рис. 1 б) для нескольких значений управляющего напряжения, подаваемого на

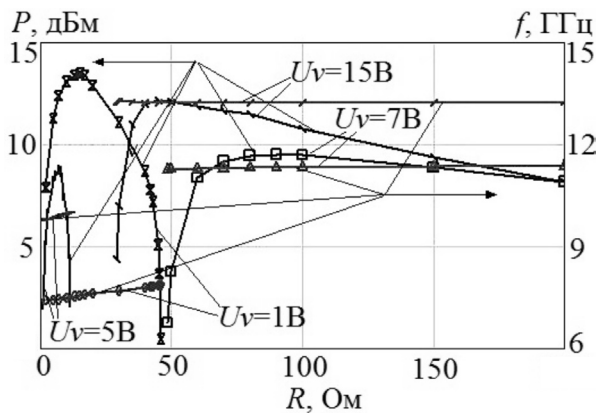


Рис. 3. Зависимости выходной мощности  $P$  и частоты колебаний  $f$  от величины нагрузки  $R$

Fig. 3. Power output  $P$  and oscillation frequency  $f$  vs load resistance  $R$

вариакпы:  $U_v = 1$  В, 5 В, 7 В, 15 В. Представленные на графике характеристики получены для случая отсутствия мнимой составляющей нагрузки:  $\text{Im}(Z_L) = 0$ .

Как видно из рис. 3, для обеспечения генерации во всем диапазоне изменения управляющего напряжения  $U_v$  необходима резистивная нагрузка с существенно нелинейной частотной характеристикой. Так, например, при больших значениях напряжения  $U_v = 7$ –15 В оптимальные величины нагрузки  $R$  оказываются большими, чем 40–50 Ом. При этом, например, для  $U_v = 15$  В возможно увеличение  $R$  до величин 150–200 Ом и даже более – в этом случае мощность выходного колебания снижается лишь на 3 дБ по сравнению с максимальным значением. Уменьшение же  $R$  менее 35 Ом приводит к срыву генерации. Для  $U_v = 7$  В минимально допустимые значения сопротивления нагрузки оказываются еще выше:  $R \geq 50$  Ом.

В то же время в нижней части частотного диапазона ( $U_v = 1$  В) требования к сопротивлению нагрузки оказываются диаметрально противоположными: оптимальный диапазон  $R$  составляет 7–20 Ом. При больших значениях сопротивления нагрузки ( $R > 45$  Ом) генерация невозможна. Данную ситуацию усугубляет тот факт, что в середине диапазона перестройки ( $U_v = 5$  В) требуются сверхмалые значения

нагрузочного сопротивления  $R = 5$ –7 Ом (см. рис. 3). При  $R > 11$ –12 Ом на этих частотах условия генерации не выполняются. Таким образом, для реализации непрерывной перестройки необходимо создание резистивной нагрузки, величина сопротивления которой изменялась бы более чем на порядок в рассматриваемом частотном диапазоне, что, по всей вероятности, не представляется практически осуществимым в диапазоне СВЧ.

Для поиска путей построения потенциально реализуемой нагрузочной цепи, обеспечивающей необходимые условия генерации во всей требуемой полосе частот, было проведено исследование влияния реактивной составляющей нагрузки генератора на его характеристики в каждой точке диапазона изменения управляющего напряжения, подаваемого на варикапы. В результате было установлено, что добавление реактивной компоненты нагрузочного сопротивления позволяет реализовать режим автоколебаний при менее жестких требованиях к вещественной составляющей нагрузки  $Z_L$ , что дает возможность искать пути построения микрополосковых цепей ГУН, обладающих соответствующими характеристиками.

На рис. 4 приведены зависимости частоты и мощности выходного колебания ГУН от величины мнимой составляющей полного сопротивления выходной согласующей цепи генератора для нескольких значений управляющего напряжения, подаваемого на варикапы ( $U_v = 1$  В, 7 В, 15 В). Приведенные характеристики получены при  $\text{Re}(Z_L) = 20$  Ом.

Следует отметить, что для некоторых вариантов схем использование произвольной чисто вещественной нагрузки и вовсе не позволяет обеспечить условия генерации в определенных точках рассматриваемого частотного диапазона.

Проведенные исследования показали, что на границах частотного диапазона в области малых ( $U_v = 1$ –2 В, рис. 4 а) и больших ( $U_v > 11$ –12 В, рис. 4 в) управляющих напряжений требуется комплексная нагрузка с реактивностью индуктивного характера ( $\text{Im}(Z_L) > 0$ ), а в

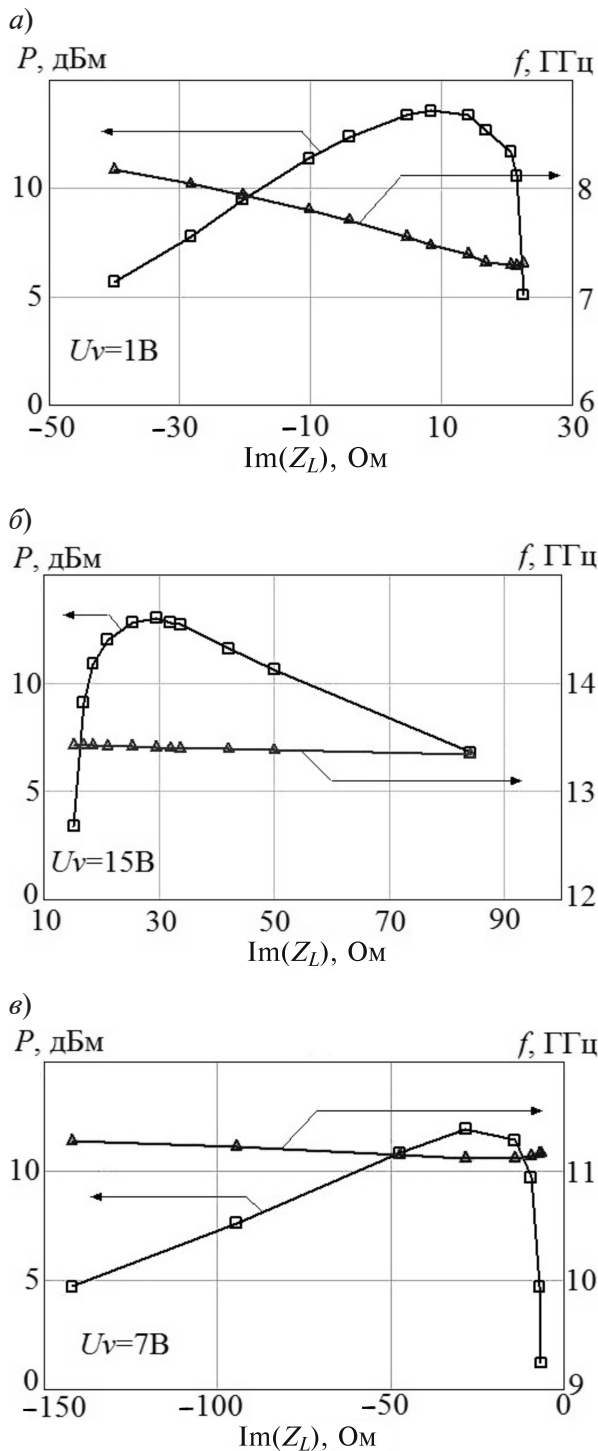


Рис. 4. Зависимости выходной мощности  $P$  и частоты колебаний  $f$  от величины мнимой составляющей нагрузки  $Z_L$

Fig. 4. Power output  $P$  and oscillation frequency  $f$  vs imaginary part of load impedance  $Z_L$

средней части рассматриваемого диапазона ( $U_v = 5-7$  В, рис. 4 б) необходимо использовать нагрузку с реактивностью емкостного типа ( $\text{Im}(Z_L) < 0$ ).

Таким образом, использование комплексной нагрузки ГУН со специальным законом изменения ее реактивной составляющей дает возможность удовлетворить условиям генерации (1) во всей требуемой полосе частот при реализуемых значениях ее вещественной компоненты (20–40 Ом).

Проведенное исследование влияния выходной цепи генератора на характеристики автоколебаний, выполненное на основе модели на рис. 1 б), позволило сформулировать требования к импедансным характеристикам нагрузки, необходимым для функционирования ГУН. На основе данных требований были определены ориентировочные границы допустимых областей изменения вещественной  $\text{Re}(Z_L)$  и мнимой  $\text{Im}(Z_L)$  составляющих полного сопротивления  $Z_L$  выходной цепи генератора, представленные на рис. 5. Цифрой 1 обозначены примерные границы диапазона изменения  $\text{Re}(Z_L)$ , а цифрой 2 –  $\text{Im}(Z_L)$ .

Полученные ограничения использовались при реализации процедуры синтеза согласующе-трансформирующих цепей, выполненных в рамках гибридной технологии. В дополнение к ограничениям на входной импеданс согласующей цепи  $Z_L$  (см. рис. 5) при их построении также учитывалось условие  $|S_{21}| < \delta$ , где  $S_{21}$  – элемент волновой матрицы рассеяния синтезируемой цепи, задающий коэффициент передачи СТЦ, а величина  $\delta$  определяет степень необходимой развязки генератора с нагрузкой ( $\delta = 0,2-0,4$ ).

Используя такой подход, удалось определить несколько вариантов построения «сложных» выходных СТЦ ГУН, применение которых в схеме генератора позволяет добиться непрерывной перестройки в рассматриваемом диапазоне частот. Импедансные характеристики некоторых таких цепей приведены на рис. 5. Разработанные СТЦ выполнены на основе отрезков МПЛ с использованием SMD-резисторов и конденсаторов.

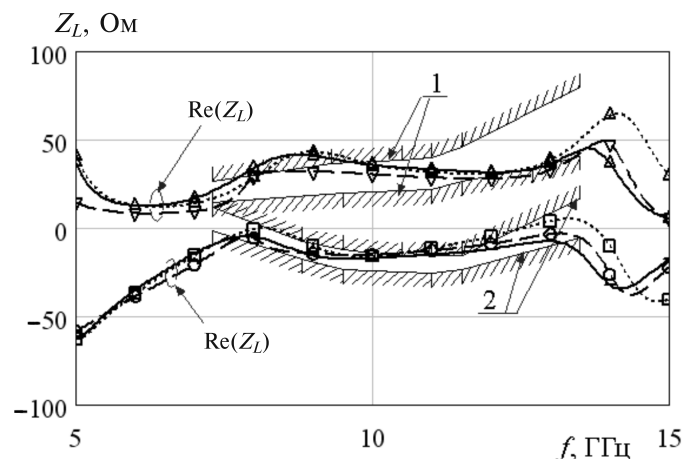


Рис. 5. Импедансные характеристики нагрузочной цепи ГУН

Fig. 5. Impedance characteristics of the VCO load circuit

### Результаты измерения характеристик ГУН

На основе разработанной схемы генератора, включающей в себя найденные структуры выходной СТЦ, было создано несколько вариантов топологии микрополоскового ГУН. Платы генератора реализованы на основе ламинированного СВЧ-материала серии RO4350B с применением дискретных компонентов типоразмеров 0201 и 0402. В качестве АЭ ГУН использован гетеропереходный биполярный SiGe-транзистор NESG3031M14 [18], а в качестве управля-

ющих элементов — варикапы серии MA46 компании MACOM [17].

Экспериментальные исследования позволили уточнить параметры элементов ГУН. На рис. 6 приведены характеристики одного из вариантов автогенератора.

Как видно из графика, приведенного на рисунке, разработанный ГУН обеспечивает перестройку частоты в пределах от 6 до 12,5 ГГц при изменении управляющего напряжения в диапазоне от 1 до 15 В со средним уровнем выходной мощности порядка 2 дБм.

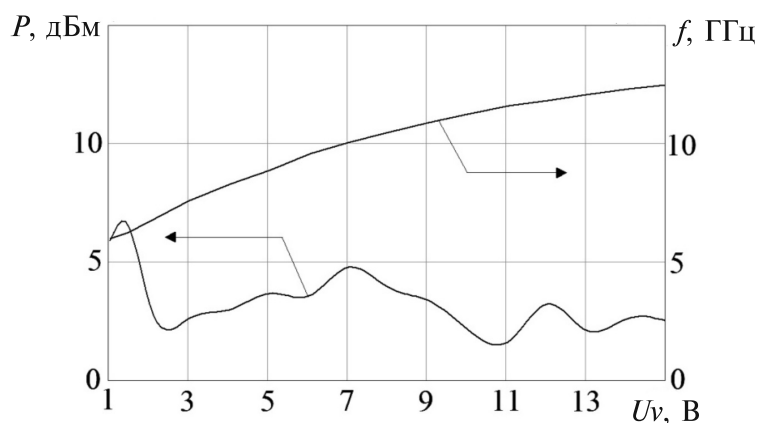


Рис. 6. Зависимости выходной мощности  $P$  и частоты колебаний  $f$  ГУН от управляющего напряжения  $U_v$

Fig. 6. Power output  $P$  and oscillation frequency  $f$  vs control voltage  $U_v$



## Заключение

В результате проведенных исследований сверхширокополосного ГУН с октавной перестройкой частоты в диапазоне от 6 до 12 ГГц, выполненного по гибридной технологии с использованием стандартных промышленно выпускаемых электронных компонентов, развита методика разработки генератора в части создания выходных согласующе-трансформирующих цепей, обеспечивающих непрерывную перестройку в рассматриваемом диапазоне частот. Показано, что для получения требуемых параметров ГУН требуются выходные цепи специального вида, формирующие в пло-

скости их подключения к генератору комплексное сопротивление нагрузки, у которого мнимая составляющая меняет свой знак в диапазоне перестройки.

Образец разработанного ГУН, реализованный на основе стандартных электронных компонентов, позволяет перестраивать частоту выходного колебания в пределах от 6 до 12 ГГц при изменении управляющего напряжения, подаваемого на варикапы, в диапазоне от 1 до 13 В включительно со средним уровнем выходной мощности порядка 2 дБм, что соответствует аналогичным параметрам интегральных схем зарубежного производства [6, 7].

## СПИСОК ЛИТЕРАТУРЫ

1. **Chenakin A.** Frequency synthesis: current status and future projections // *Microwave Journal*. 2017. Vol. 60. No. 4. Pp. 22–36.
2. **Chenakin A.** Frequency synthesizers: concept to product. Artech House, Inc., 2011. 254 p.
3. **Chenakin A.** Select a VCO or YIG for a PLL synthesizer // *Microwaves and RF*. 2016. Vol. 55. No. 2. Pp. 2–6.
4. **Khanna A.P.S.** State of the art in microwave VCOs // *Microwave Journal*. 2015. No. 5. Pp. 23–40.
5. **Горевой А.В.** Выбор генераторов для построения маломощных СВЧ-синтезаторов // *Компоненты и технологии*. 2012. № 6. С. 12–17.
6. **HMC732LC4B.** Wideband MMIC VCO with Buffer Amplifier // URL: <http://www.analog.com/media/en/technical-documentation/data-sheets/hmc732.pdf> (Дата обращения: 08.11.2017).
7. **MAOC-409000.** Broadband Voltage Controlled Oscillator 6-12 GHz // URL: [https://cdn.macom.com/datasheets/MAOC-409000\\_V3P.pdf](https://cdn.macom.com/datasheets/MAOC-409000_V3P.pdf) (Дата обращения: 08.11.2017).
8. **Naik M.R., Vithalani C.H.** Wideband local oscillator design for wideband transceivers // *Proc. of 2011 IEEE Internat. RF and Microwave Conf. Malaysia*, 2011. Pp. 33–36.
9. **Горевой А.** Генератор диапазона 1-2 ГГц с повышенной крутизной регулировочной характеристики // *Доклады ТУСУР*. 2011. № 1(23). С. 44–49.
10. **Кочемасов В., Строганова Е.** Электронные компоненты иностранного производства. Ограничение экспорта в Россию // *Электрони-*
- ка *НТБ*. 2013. № 1. С. 125–129.
11. **Stauffer G.H.** Finding the lumped element varactor diode model // *High Frequency Electronics*. 2003. Vol. 2. No. 6. Pp. 22–28.
12. **Sun W.** Accurate EM simulation of SMT components in RF designs // *Proc. of the 2017 IEEE Radio Frequency Integrated Circuits Symp. USA*, 2017. Pp.140–143.
13. **Мальшев В.М., Матвеев Ю.А., Никитин А.Б., Худяков А.В.** Выбор варактора для широкополосного перестраиваемого генератора // *Материалы междунар. науч.-техн. конф. Актуальные проблемы электронного приборостроения*. 2014. Т. 1. С. 34–41.
14. **Никитин А.Б., Хабитуева Е.И.** СВЧ-модель бескорпусного резистора // *Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление*. 2015. № 5 (229). С. 39–46. DOI: 10.5862/JCSTCS.229.4
15. **Grebennikov A.V.** RF and Microwave. Transistor Oscillator Design. John Wiley & Sons, Ltd., 2007. 441 p.
16. **Tsuru M., Kawakami K., Tajima K., Miyamoto K., Nakane M., Itoh K., Miyazaki M., Isota Y.** A Triple-Tuned Ultra-Wideband VCO // *IEEE Trans. on MTT*. 2008. Vol. 56. No. 2. Pp. 346–354.
17. **MA46 Series** // URL: <https://cdn.macom.com/datasheets/MA46%20Series.pdf> (Дата обращения: 15.11.2017).
18. **NPNSiliconGermaniumRFTtransistorNE SG3031M14** // URL: <http://www.cel.com/pdf/datasheets/nsg3031m14.pdf> (Дата обращения: 09.11.2017).

*Статья поступила в редакцию 04.11.2017.*

## REFERENCES

1. **Chenakin A.** Frequency synthesis: current status and future projections. *Microwave Journal*, 2017, Vol. 60, No. 4, Pp. 22–36.
2. **Chenakin A.** *Frequency synthesizers: concept to product*. Artech House, Inc., 2011, 254 p.
3. **Chenakin A.** Select a VCO or YIG for a PLL synthesizer. *Microwaves and RF*, 2016, Vol. 55, No. 2, Pp. 2–6.
4. **Khanna A.P.S.** State of the art in microwave VCOs. *Microwave Journal*, 2015, No. 5, Pp. 23–40.
5. **Gorevoy A.V.** Vybor generatorov dlya postroyeniya maloshumyashchikh SVCh-sintezatorov [Select an oscillator for a low noise microwave synthesizer]. *Komponenty i tekhnologii [Components & Technologies]*, 2012, No. 6, Pp. 12–17. (rus)
6. **HMC732LC4B. Wideband MMIC VCO with Buffer Amplifier.** Available: <http://www.analog.com/media/en/technical-documentation/data-sheets/hmc732.pdf> (Accessed: 08.11.2017).
7. **MAOC-409000. Broadband Voltage Controlled Oscillator 6-12 GHz.** Available: [https://cdn.macom.com/datasheets/MAOC-409000\\_V3P.pdf](https://cdn.macom.com/datasheets/MAOC-409000_V3P.pdf) (Accessed: 08.11.2017).
8. **Naik M.R., Vithalani C.H.** Wideband local oscillator design for wideband transceivers. *Proceedings of 2011 IEEE International RF and Microwave Conference*, Malaysia, 2011, Pp. 33–36.
9. **Gorevoy A.** Generator diapazona 1-2 GHz s povyshennoy krutiznoy regulirovochnoy kharakteristiki [VCO 1-2 GHz with increased tuning sensitivity]. *Doklady TUSUR [Proceedings of TUSUR Journal]*, 2011, No. 1(23), Pp. 44–49. (rus)
10. **Kochemasov V., Stroganova Ye.** Elektronnyye komponenty inostrannogo proizvodstva. Ogranicheniye eksporta v Rossiyu [Foreign-made electronic components. Russian exports restriction]. *Elektronika NTB [Electronics: STB]*, 2013, No. 1, Pp. 125–129. (rus)
11. **Stauffer G.H.** Finding the lumped element varactor diode model. *High Frequency Electronics*, 2003, Vol. 2, No. 6, Pp. 22–28.
12. **Sun W.** Accurate EM simulation of SMT components in RF designs. *Proceedings of the 2017 IEEE Radio Frequency Integrated Circuits Symposium*, USA, 2017, Pp. 140–143.
13. **Malyshev V.M., Matveyev Yu.A., Nikitin A.B., Khudyakov A.V.** Vybor varaktora dlya shirokopolosnogo perestraivayemogo generatora [A varactor diodes selection for the broadband microwave voltage controlled oscillator]. *Proceedings of 2014 International Conference on Actual Problems of Electron Devices Engineering*, Saratov, Russian Federation, 2014, Vol. 1, Pp. 34–41. (rus)
14. **Nikitin A.B., Khabituyeva Ye.I.** A Surface mount chip resistor model. *St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control System*, 2015, No. 5(229), Pp. 39–46. (rus) DOI: 10.5862/JCSTCS.229.4
15. **Grebennikov A.V.** *RF and Microwave Transistor Oscillator Design*. John Wiley & Sons, Ltd., 2007, 441 p.
16. **Tsuru M., Kawakami K., Tajima K., Miyamoto K., Nakane M., Itoh K., Miyazaki M., Isota Y.** A Triple-Tuned Ultra-Wideband VCO. *IEEE Trans. on MTT*, 2008, Vol. 56, No. 2, Pp. 346–354.
17. **MA46 Series.** Available: <https://cdn.macom.com/datasheets/MA46%20Series.pdf> (Accessed: 15.11.2017).
18. **NPN Silicon Germanium RF Transistor NESG3031M14.** Available: <http://www.cel.com/pdf/datasheets/nescg3031m14.pdf> (Accessed: 09.11.2017).

Received 04.11.2017.

## СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

**НИКИТИН Александр Борисович**

**NIKITIN Aleksandr B.**

E-mail: [nikitin@mail.spbstu.ru](mailto:nikitin@mail.spbstu.ru)

**ХАБИТУЕВА Екатерина Исаковна**

**KHABITUEVA Ekaterina I.**

E-mail: [basilliounderground@mail.ru](mailto:basilliounderground@mail.ru)

## О КОРРЕКТНОСТИ КОМПИЛЯЦИИ ПОДМНОЖЕСТВА ОБЕЩАЮЩЕЙ МОДЕЛИ ПАМЯТИ В АКСИОМАТИЧЕСКУЮ МОДЕЛЬ ARMv8.3

*А.В. Подкопаев<sup>1</sup>, О. Лахав<sup>2</sup>, В. Вафеядис<sup>3</sup>*

<sup>1</sup>Санкт-Петербургский государственный университет,  
Санкт-Петербург, Российская Федерация;

<sup>2</sup>Тель-Авивский университет, Тель-Авив, Израиль;

<sup>3</sup>Институт имени Макса Планка: Программные системы, Кайзерслаутерн, Германия

«Обещающая» модель памяти является перспективным решением проблемы задания семантики многопоточности в контексте императивных языков программирования, таких как C/C++ и Java. Естественным требованием, которое ставится перед моделью памяти языка программирования, является наличие эффективной и корректной схемы компиляции для распространенных процессорных архитектур. Ранее для обещающей модели была показана корректность компиляции в архитектуры x86, Power и для операционной модели памяти ARMv8 POP. В статье приведено доказательство корректности компиляции в аксиоматическую модель ARMv8.3. В доказательстве использован новый метод обхода исполнений аксиоматических моделей памяти. Этот метод является более общим, чем использованные ранее подходы, и может использоваться в последующих доказательствах корректности компиляции из обещающей модели памяти.

**Ключевые слова:** многопоточность; корректность компиляции; слабые модели памяти; ARM; семантика.

**Ссылка при цитировании:** Подкопаев А.В., Лахав О., Вафеядис В. О корректности компиляции подмножества обещающей модели памяти в аксиоматическую модель ARMv8.3 // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2017. Т. 10. № 4. С. 51–69. DOI: 10.18721/JCSTCS.10405

## ON COMPILATION CORRECTNESS FOR A SUBSET OF A PROMISING MEMORY MODEL TO THE ARMv8.3 MEMORY MODEL

*A.V. Podkopaev<sup>1</sup>, O. Lahav<sup>2</sup>, V. Vafeiadis<sup>3</sup>*

<sup>1</sup>St. Petersburg State University, St. Petersburg, Russian Federation;

<sup>2</sup>Tel Aviv University, Tel Aviv, Israel;

<sup>3</sup>Max Planck Institute for Software Systems, Kaiserslautern, Germany

A ‘promising’ memory model is an auspicious solution to the problem of defining semantics for an imperative language with concurrency, such as C/C++ or Java.

An essential requirement for such a memory model is the existence of an effective and correct compilation scheme from the language to its target platforms. There are compilation correctness proofs from the promising model to x86 and Power as well as to an operational model ARMv8 POP. This paper presents such proof for an axiomatic memory model for ARMv8.3. In the proof, we use a new method of execution traversal, which might be used in other compilation correctness proofs for the promising memory model.

**Keywords:** concurrency; compilation correctness; weak memory models; ARM; semantics.

**Citation:** Podkopaev A.V., Lahav O., Vafeiadis V. On compilation correctness for a subset of a Promising memory model to the ARMv8.3 memory model. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2017, Vol. 10, No. 4, Pp. 51–69. DOI: 10.18721/JCSTCS.10405

## 1. Введение

Параллельные вычисления давно являются естественной частью прикладного программирования. Несмотря на это, до сих пор ведется активная работа по разработке формальных семантик для языков программирования и процессоров с многопоточностью, или *моделей памяти* [4, 7, 9, 13, 17, 20]. Под моделью памяти можно понимать функцию, которая по программе выдает множество возможных *поведений*, т. е. состояний системы, исполняющей программу, после выполнения программы. Чаще всего речь идет об оперативной памяти и значении регистров.

Наиболее известная модель памяти для многопоточных программ — модель *последовательной консистентности* (sequential consistency — SC) [14]. Любое поведение программы в данной модели может быть получено как некоторое попеременное исполнение команд разных потоков на одноядерном процессоре. К сожалению, такая модель памяти не способна описывать поведения, которые наблюдаются у программ, оптимизированных компилятором и запущенных на современных процессорах. Так, аналог следующей программы, написанный на C++, обработанный компилятором gcc и запущенный на процессоре семейства x86, может завершиться с результатом  $a = b = 0^1$ :

$$\begin{array}{l} [x] := 1; \parallel [y] := 1; \\ a := [y]; \text{ // } 0 \parallel b := [x]; \text{ // } 0 \end{array} \quad (\text{SB})$$

Данное поведение, которое в литературе именуется буферизацией записи (store buffering — SB) [23], не может быть получено поочередным исполнением потоков, т. к. при таком исполнении одна из записей ( $[x] := 1$  и  $[y] := 1$ ) должна случиться раньше любого из чтений ( $a := [y]$  и  $b := [x]$ ). Поведение параллельной программы, которое выходит за рамки последовательной консистентности, называют слабым. В свою очередь, слабая модель памяти — это модель, допускающая слабые поведения.

Первая причина появления слабых поведения у многопоточных программ — компиляторные оптимизации. Например, компилятор gcc с включенным режимом оптимизации переставляет независимые обращения к памяти в рамках планирования кода. Соответствующая перестановка наблюдается для аналогов программы SB. Второй причиной является оптимизирующее поведение современных процессоров: процессоры могут исполнять инструкции не по порядку, а обращения к памяти могут быть переупорядочены за счет использования локальной памяти ядра.

Компиляторные и процессорные оптимизации стали широко применяться задолго до появления многопоточных процессоров; они дают существенный прирост производительности. Как следствие, большинство экспертов сходятся во мнении, что лучше разрешить слабые поведения программ, чем запретить оптимизации.

В последние годы были созданы сла-

<sup>1</sup> Здесь и во всех последующих примерах мы предполагаем, что  $x$ ,  $y$  являются различными адресами в памяти, в которые изначально записаны нули, а  $a$ ,  $b$  являются локальными переменными (регистрами).



бые модели памяти для самых распространенных архитектур: x86 [23], Power [3, 21] и ARM [9, 20]. Эти модели разработаны в сотрудничестве с компаниями, создающими данные архитектуры. Они описывают не только поведения, наблюдаемые с помощью тестирования существующих процессоров, но и оставляют пространство для оптимизаций, планируемых в новых версиях архитектур.

Существуют слабые модели и для пространственных языков программирования, таких как Java [17] и C/C++ [7]. Данные модели входят в соответствующие стандарты языков. Источником слабых поведений в контексте моделей памяти языков программирования являются оптимизации, которые создатели стандарта хотят разрешить для реализации в компиляторах, а также слабые поведения процессоров, которые являются целевыми для языков и их платформ.

У существующих моделей памяти Java и C/C++ имеются недостатки: разрешение «значений из воздуха» (out-of-thin-air values) [8], некорректность широко применяемых на практике оптимизаций [24] и слишком строгие требования к семантике процессоров в контексте используемых схем компиляции [13, 16]. Недавно была представлена новая модель памяти [10], т. н. «обещающая» (promising) модель памяти, призванная решить упомянутые проблемы.

Потенциально, обещающая модель может стать частью стандартов языков Java и C/C++; обсуждение этого уже начинается в соответствующих комитетах. Для того чтобы обещающая модель смогла стать частью стандартов Java и C/C++, должна быть доказана корректность существующих (эффективных) схем компиляции [2] из обещающей модели в модели памяти основных процессорных архитектур (x86, Power, ARM).

Под корректностью компиляции схемы *compile* из модели *A* в модель *B* мы подразумеваем следующее:

$$\forall Prog \in A. \llbracket compile(Prog) \rrbracket_B \subseteq \llbracket Prog \rrbracket_A,$$

а именно, что для любой программы *Prog* в языке определения модели *A* и ее ском-

пилированной версии *compile(Prog)* в языке определения модели *B* выполняется, что любое поведение *compile(Prog)* в *A* также является поведением *Prog* в *A*. Это естественное требование, позволяющее программисту быть уверенным в том, что написанная им программа работает только так, как описано в стандарте языка ее реализации, и никак иначе.

В работе [10] доказана корректность компиляции из обещающей модели в модели x86-TSO [23] и Power [3]; в [19] — корректность компиляции из подмножества обещающей модели в операционную модель памяти ARMv8 POP [9]. В апреле 2017 года была предложена новая модель памяти для архитектуры ARM, модель ARMv8.3 [1]. Как следствие, актуальна задача доказательства корректности компиляции в новую модель ARMv8.3. Решению этой задачи посвящена данная статья.

В статье показана корректность компиляции для подмножества обещающей модели без сертификации [10] в модель ARMv8.3. Рассматриваемое подмножество состоит из ослабленных (relaxed) обращений памяти (записей и чтений), высвобождающих (release) и приобретающих (acquire) барьеров памяти. Несмотря на то, что мы не доказываем корректность компиляции для всей модели, мы уверены, что доказательство может быть расширено, что является нашей дальнейшей работой. Но в связи с тем, что текущее доказательство достаточно объемное и сложное, а результат является базовым для дальнейшей работы, мы представляем доказательство корректности компиляции для подмножества модели.

Распространенным способом показать корректность компиляции из абстрактной машины *A* в машину *B* является доказательство симуляции между ними [18]. Для этого определяется отношение симуляции, связывающее состояния машин, и доказывается, что если текущие состояния машин *A* и *B* связаны отношением симуляции, и машина *B* делает шаг, то машина *A* может сделать ноль и более шагов так, чтобы новые состояния машин были также связаны отношением симуляции. Непосредственно данный подход не применим для



доказательства корректности компиляции из обещающей модели в модель ARMv8.3, т. к. модель памяти ARMv8.3 является аксиоматической, т. е. семантика конкретной программы в этой модели есть множество графов, каждый из которых представляет некоторый конкретный запуск программы. Это означает, что в модели ARMv8.3 запуск программы определяется сразу и целиком, а не по шагам, как это происходит в обещающей модели, заданной в виде абстрактной машины. Чтобы преодолеть данное ограничение, мы разработали способ обхода графов модели ARMv8.3, который представлен в виде операционной семантики. Этот обход и используется при доказательстве симуляции между обещающей и ARMv8.3 моделями. Данный метод может применяться для доказательства корректности компиляции из обещающей в аксиоматические модели памяти и является более общим, чем использованные ранее методы (см. раздел 9), базирующиеся на особых свойствах целевых аксиоматических моделей.

## 2. Обещающая модель на примерах

В этом разделе мы приведем неформальное описание обещающей модели памяти [10] на нескольких примерах.

**2.1. Базовые элементы обещающей модели.** Рассмотрим программу MP (передача сообщения, message passing):

$$\begin{array}{l} [x] := 1; \parallel a := [y]; //1 \\ [y] := 1; \parallel b := [x]; //0 \end{array} \quad (\text{MP})$$

Эта программа является упрощенным вариантом шаблона, используемого для передачи данных между потоками. Первый поток записывает данные в локацию  $x$  и потом выставляет флаг (локация  $y$ ), что данные подготовлены; в свою очередь, второй поток проверяет флаг, а потом читает данные. Модель последовательной консистентности гарантирует, что если второй поток увидел, что флаг выставлен ( $a = 1$ ), то он увидит и подготовленные данные ( $b = 1$ ). Данное заключение неверно для многих слабых моделей, в том числе для обещающей и ARMv8.3 моделей. Рассмотрим, как слабое поведение  $a = 1, b = 0$  достигается при исполнении MP в обещающей

модели памяти.

Обещающая модель памяти задана операционным способом. Это означает, что существует некоторая абстрактная машина, связанная с моделью, которая исполняет программы по шагам, и множество всех конечных состояний этой машины для некоторой программы является семантикой программы в рамках данной модели. Далее мы будем использовать термин *обещающая машина* для обозначения упомянутой абстрактной машины.

Состояние обещающей машины включает в себя множество сообщений, которое называется *памятью*. Мы считаем, что память перед исполнением программы содержит по сообщению со значением ноль для каждой локации. Так, для приведенной выше программы начальная память будет состоять из двух сообщений:

$$M_{init} = \{\langle x : 0 @ 0 \rangle, \langle y : 0 @ 0 \rangle\}.$$

Сообщение — это тройка из локации, значения и метки времени (timestamp, в приведенном примере — 0 после @). Метки времени используются для упорядочивания сообщений, связанных с одной и той же локацией.

Теперь покажем некоторое исполнение MP, которое приводит к  $a = 1, b = 0$ . Сначала левый поток программы MP выполняет обе записи, после чего память обещающей машины содержит четыре сообщения:

$$M = \{\langle x : 0 @ 0 \rangle, \langle y : 0 @ 0 \rangle, \langle x : 1 @ 5 \rangle, \langle y : 1 @ 3.5 \rangle\}.$$

Заметим, что метки времени новых сообщений должны быть больше, чем метки инициализирующих сообщений, но между собой они никак не связаны и могли бы совпасть — выбор меток времени в обещающей машине недетерминирован. Так, конкретное значение метки времени сообщения не имеет значения, важно лишь соотношение между метками сообщений, относящихся к той же локации.

Следующим шагом в выбранном запуске, приводящем к  $a = 1, b = 0$ , правый поток выполняет чтение из локации  $y$ . Обещающая машина разрешает потоку произвести чтение из любого сообще-

ния локации  $y$  в памяти,  $\langle y : 0 @ 0 \rangle$  или  $\langle y : 1 @ 3.5 \rangle$ . Поскольку мы хотим получить  $a = 1$ , правый поток производит чтение из сообщения с меткой времени 3.5. Далее, если бы поток еще раз читал из локации  $y$ , ему было бы запрещено производить чтение из сообщений с меткой времени меньше 3.5. Интуитивно, если поток увидел более новое значение, то он больше не должен видеть более старое значение.

Для того чтобы обеспечить данное свойство, у каждого потока обещающей машины есть фронт — взгляд (view, viewfront) на память. Это функция, которая по локации возвращает метку времени последнего сообщения, относящегося к этой локации, которое было увидено потоком. Так, перед тем как левый поток выполнил инструкции записи, фронты левого и правого потоков указывают на изначальные сообщения:

$$T1.view_{cur} = [x @ 0, y @ 0]$$

$$T2.view_{cur} = [x @ 0, y @ 0].$$

После того как левый поток выполнил обе инструкции записи, он больше не может видеть более старые значения. Это гарантируется тем, что его фронт по каждой локации увеличился и стал равным метки времени соответствующего сообщения:

$$T1.view_{cur} = [x @ 5, y @ 3.5]$$

$$T2.view_{cur} = [x @ 0, y @ 0].$$

После того как правый поток выполнил чтение из сообщения  $y$  с меткой времени 3.5, его фронт увеличивается по  $y$  (становится равным максимуму из старого значения по  $y$ , 0, и метки времени сообщения, 3.5):

$$T1.view_{cur} = [x @ 5, y @ 3.5]$$

$$T2.view_{cur} = [x @ 0, y @ 3.5].$$

В программе остается невыполненной только второе чтение из локации  $x$  в правом потоке. Это чтение правый поток может совершить из сообщения с меткой времени 0, так как  $T2.view_{cur}(x) \leq 0$ .

**2.2 Барьеры памяти.** Как же гарантировать передачу сообщения между потоками в программе МР, а именно запретить результат  $a = 1, b = 0$ ? Для этого нужно до-

бавить в программу МР несколько *барьеров памяти* (memory fences) — специальных инструкций, которые в реальных системах запрещают некоторые оптимизации со стороны компилятора и процессора. Например, *высвобождающий* (release, rel) барьер используется для того, чтобы запретить переупорядочивание инструкций записи, а *приобретающий* (acquire, acq) — инструкций чтения. Так, версия программы МР с высвобождающим и приобретающим барьерами в левом и правом потоках соответственно не может завершиться с результатом  $a = 1, b = 0$ :

$$\begin{array}{l} [x] := 1; \parallel a := [y]; //1 \\ \text{fence}(\text{rel}); \parallel \text{fence}(\text{acq}); \text{ (MP-rel-acq)} \\ [y] := 1; \parallel b := [x]; //0 \end{array}$$

Для поддержки барьеров памяти нужно внести некоторые изменения в память обещающей машины — у каждого сообщения появляется свой фронт. Интуитивно, фронт сообщения  $msg$  указывает на те сообщения в памяти, которые становятся видны потоку, который читает из сообщения  $msg$  и после этого выполняет приобретающий барьер. Так, после выполнения инструкции записи левым потоком, память машины выглядит следующим образом:

$$\begin{aligned} M = \{ & \langle x : 0 @ 0, [x @ 0] \rangle, \langle y : 0 @ 0, [y @ 0] \rangle, \\ & \langle x : 1 @ 5, [x @ 5, y @ 0] \rangle, \\ & \langle x : 1 @ 3.5, [x @ 5, y @ 3.5] \rangle, \end{aligned}$$

В данном случае интерес представляет фронт второго сообщения локации  $y$  —  $[x @ 5, y @ 3.5]$ . После того как левый поток выполнит чтение из этого сообщения и исполнит приобретающий барьер, его фронт будет равен  $[x @ 5, y @ 3.5]$ , что запретит ему чтение из сообщения  $\langle x : 0 @ 0, [x @ 0] \rangle$ .

Для поддержки барьеров нам также нужно добавить по два дополнительных фронта потокам — высвобождающий и приобретающий. Рассмотрим тот же запуск программы МР. **Перед исполнением программы** все три фронта левого потока равны  $[x @ 0, y @ 0]$ . После выполнения инструкции записи в локацию  $x$ , базовый и приобретающий фронты левого потока соответственным образом растут, тогда как высвобож-

дающий остается без изменений:

$$T1.view_{cur} = [x @ 5, y @ 0]$$

$$T1.view_{acq} = [x @ 5, y @ 0]$$

$$T1.view_{rel} = [x @ 0, y @ 0].$$

После выполнения барьера высвобождающий фронт становится равным базовому:

$$T1.view_{cur} = [x @ 5, y @ 0]$$

$$T1.view_{acq} = [x @ 5, y @ 0]$$

$$T1.view_{rel} = [x @ 5, y @ 0].$$

Теперь, когда левый поток добавляет новое сообщение локации  $y$  в память, фронт этого сообщения равен высвобождающему фронту потока, увеличенному на  $[y@3.5]$  — метку времени самого сообщения.

Перейдем к правому потоку. Его фронты также изначально равны  $[x@0, y@0]$ . После того как поток читает из сообщения  $\langle y : 1 @ 3.5, [x @ 5, y @ 3.5] \rangle$ , его базовый фронт увеличивается на метку времени самого сообщения,  $[y@3.5]$ , тогда как приобретающий — на фронт сообщения:

$$T2.view_{cur} = [x @ 0, y @ 3.5]$$

$$T2.view_{acq} = [x @ 5, y @ 3.5]$$

$$T2.view_{rel} = [x @ 0, y @ 0].$$

Выполнение барьера приравнивает базовый фронт к приобретающему:

$$T2.view_{cur} = [x @ 5, y @ 3.5]$$

$$T2.view_{acq} = [x @ 5, y @ 3.5]$$

$$T2.view_{rel} = [x @ 0, y @ 0].$$

После этого правый поток не может прочитать сообщение  $\langle x : 0 @ 0, [x @ 0] \rangle$ , так как  $T2.view_{cur} > 0$ .

**2.3 Механизм обещаний.** Как мы только что убедились, обещающая модель позволяет задавать поведения многопоточной программы, выходящие за пределы простого попеременного исполнения потоков над обычной памятью. Это достигается за счет устройства памяти обещающей машины, которая может хранить более одного сообщения для каждой локации. Тем не менее этого недостаточно для выражения всех поведения, наблюдаемых на современных архитектурах. Например, следующая програм-

ма LB (load buffering) может завершиться с результатом  $a = b = 1$ :

$$a := [x]; //1 \parallel b := [y]; //1 \quad (LB) \\ [y] := 1; \parallel [x] := 1$$

Такое поведение не может наблюдаться в версии обещающей машины, описанной выше, т. к. для получения  $a = b = 1$  первой инструкцией должна выполняться одна из инструкций записи. Архитектурные модели памяти, такие как ARMv8 POP [9], явным образом разрешают исполнение инструкций не по порядку (out of order execution). В обещающей модели нет такой возможности. Зато любой поток в любой момент исполнения программы может пообещать сделать запись некоторого сообщения в память в будущем. Например, первым шагом обещающей модели для программы LB может быть обещание левым потоком сделать запись в локацию  $y$ . После этого в памяти машины появляется соответствующее сообщение:

$$M = \{ \langle x : 0 @ 0, [x @ 0] \rangle, \langle y : 0 @ 0, [y @ 0] \rangle, \\ \langle y : 1 @ 2, [x @ 0, y @ 2] \rangle \}.$$

Теперь правый поток может выполнить свои инструкции: сначала прочитать из нового сообщения, а потом выполнить инструкцию записи обычным способом, без обещаний. После этого память будет содержать четыре сообщения:

$$M = \{ \langle x : 0 @ 0, [x @ 0] \rangle, \langle y : 0 @ 0, [y @ 0] \rangle, \\ \langle y : 0 @ 0, [y @ 0] \rangle, \langle x : 1 @ 2, [x @ 2, y @ 0] \rangle \},$$

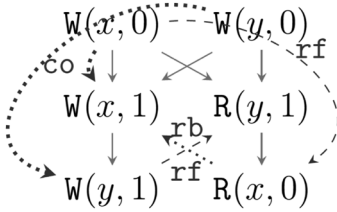
и левый поток может прочитать из сообщения  $\langle x : 1 @ 2 \rangle$  и после этого выполнить обещание записать сообщение  $\langle y : 1 @ 2, [y@2] \rangle$  в память.

Как чтение из памяти потоком ограничено его базовым фронтом, так и для обещаний существует ограничение: поток не может читать из сообщений, которые он пообещал, но еще не выполнил. Для поддержания этого инварианта у каждого потока имеется компонента состояния, которая хранит множество еще не выполненных обещаний.

### 3. Модель ARMv8.3 на примерах

Модель памяти ARMv8.3 представляет семантику программы как множество гра-

фов, каждый из которых соответствует некоторому конкретному запуску программы [20]. Так, например, для программы МР, которую мы уже рассматривали в разделе 2, исполнение программы с результатом  $a = 1$ ,  $b = 0$  выглядит следующим образом:



В графовом представлении каждый узел представляет некоторое событие (event) — операцию над памятью. Событие может быть записью (например,  $W(x, 1)$ ), чтением (например,  $R(y, 1)$ ) или барьером памяти (например,  $F(acq)$ ). В представленном выше графе  $W(x, 0)$  и  $W(y, 0)$  обозначают инициализацию, а остальные вершины — операции левого и правого потоков. Далее в статье мы будем использовать  $W$ ,  $R$  и  $F$  как для обозначения меток конкретных событий, так и для обозначения подмножества событий соответствующего типа.

Ребра в графе используются для обозначения различных отношений между событиями. Стрелки без подписи обозначают  $po$  — отношение программного порядка (program order). Программный порядок связывает инициализирующие записи со всеми остальными событиями в графе, а также является полным порядком на событиях одного и того же потока<sup>2</sup>.

События записи в одну и ту же локацию тотально упорядочены с помощью  $co$  — отношения согласованности (coherence order). Это отношение выполняет ту же функцию, что и метки времени в рамках обещающей машины.

Отношение  $rf$  («читает из», reads from) связывает события записи с событиями чтения, причем у каждого чтения должна существовать одна и только одна запись, с

которой оно (чтение) связано данным отношением, и целевые локации и значения событий должны совпадать.

Отношение  $rb$  («читает ранее», reads before) является производным от  $rf$  и  $co$ :

$$rb \triangleq rf^{-1}; co,$$

и связывает чтение  $a$  с записью, которая со-следует за записью, из которой читает  $a$ . Здесь и далее композиция отношений ';' задается так:

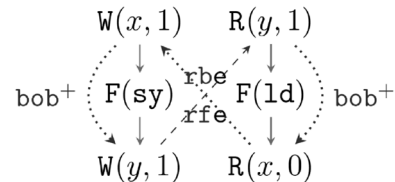
$$A; B \triangleq \{ \langle a, b \rangle \mid \exists c. \langle a, c \rangle \in A \wedge \langle c, b \rangle \in B \}.$$

Как и в случае обещающей семантики, добавление в программу МР барьеров памяти запрещает результат  $a = 1$  и  $b = 0$ :

$$\begin{array}{l} [x] := 1; \parallel a := [y]; //1 \\ fence(sy); \parallel fence(ld); \\ [y] := 1; \parallel b := [x]; //0 \end{array} \quad (\text{MP-sy-ld})$$

Данная программа не соответствует один в один программе МР-rel-acq из раздела 2, но является результатом компиляции МР-rel-acq в ассемблер ARMv8.3.

Каким образом модель ARMv8.3 запрещает поведение  $a = 1$  и  $b = 0$  для программы МР-sy-ld? Рассмотрим граф, который соответствует данному поведению:



В этом графе мы опустили инициализирующие записи и добавили отношение барьеров  $bob$  (barrier-ordered-before), заданное так:

$$\begin{aligned} bob &\triangleq po; [F(sy)] \cup [F(sy)]; po \cup [R]; \\ &po; [F(ld)] \cup [F(ld)]; po, \end{aligned}$$

где  $[A] \triangleq \{ \langle a, a \rangle \mid a \in A \}$ . Кроме того, мы заменили ребра отношений  $rf$  и  $rb$  ребрами отношений  $rfe$  и  $rbe$ , где  $e$  (external) означает, что соответствующее отношение связывает только события разных потоков.

В приведенном графе есть цикл из ребер  $bob$ ,  $rfe$  и  $rbe$ , что противоречит одной из аксиом, которая используется в модели ARMv8.3 для определения ARM-согласованного исполнения (подробнее в

<sup>2</sup> Программный порядок транзитивен, но в приведенном графе мы оставили только непосредственные ребра.



разделе 6). Таким образом граф не является ARM-согласованным, и поведение  $a = 1$  и  $b = 0$  запрещено для программы MP-sy-ld в рамках модели.

#### 4. Структура доказательства корректности компиляции

Центральным результатом данной работы является доказательство следующей теоремы.

**Теорема 4.1.** Для любых программы  $Prog$ , результата ее компиляции  $Prog_{ARM}$  и ARM-согласованного исполнения  $G$  программы  $Prog_{ARM}$  существует исполнение  $Prog$  обещающей машиной, т. ч. финальное состояние памяти машины совпадает с состоянием памяти  $G$ .

Здесь под «финальным состоянием памяти» мы понимаем значение последних записей в локации, т. е. записей с наибольшими метками времени, в случае обещающей семантики, и событий-максимумов по отношению частичного порядка  $G.co$ , в случае ARMv8.3. Обещающая и ARMv8.3 модели памяти заданы в существенно разных стилях и имеют множество концептуальных отличий. Для доказательства теоремы 4.1 необходимо преодолеть данные различия.

Главным отличием является то, что обещающая модель памяти представлена операционно, в терминах шагов некоторой абстрактной машины, тогда как аксиоматическая модель памяти ARMv8.3 задает семантику конкретного исполнения программы в виде графа, для которого выполняются определенные свойства. Для решения данной проблемы мы вводим операционную семантику обхода графа ARM-согласованного исполнения, которому обещающая машина может следовать (раздел 7).

Другим существенным отличием является то, как модели запрещают «плохие» поведения программ: например, чтение потоком старой записи в  $x$ , если поток уже читал из более новой записи в  $x$ . Для этого модель памяти ARMv8.3 использует аксиомы, которые проверяют отсутствие циклов в графе, тогда как в обещающей модели для этого используются фронты потоков и сообщений. Для того чтобы обойти

описанное различие между моделями, мы вводим дополнительные отношения на вершинах ARM согласованного исполнения, которые тесно связаны с фронтами обещающей модели (раздел 8). Используя описанные выше идеи, мы доказываем корректность компиляции между обещающей и ARMv8.3 моделями с помощью симуляции (раздел 8).

#### 5. Формальное определение обещающей модели

В этом разделе мы формально описываем подмножество обещающей модели памяти [10].

Состояние обещающей машины  $State_{Promise}$  является парой  $\langle TS, M \rangle$ . При этом  $M \subset Msg$  — это память машины, являющаяся множеством сообщений вида  $\langle \ell : val @ \tau, view \rangle$ :  $Msg$ , состоящих из целевой локации,  $\ell : Loc$ , значения,  $val : Val$ , метки времени,  $\tau : Time = Q$ , и фронта сообщения,  $view : V = Loc \rightarrow Time$ .  $TS : Tid \rightarrow TS$  — это функция, которая по идентификатору потока возвращает его состояние. Состояние потока,  $TS : TS$ , является тройкой  $\langle \sigma, V, P \rangle$ ;  $\sigma$  — локальное состояние потока в рамках описанной ниже помеченной системы переходов (labeled transition system — LTS);  $V = \langle view_{cur}, view_{acq}, view_{rel} \rangle : V \times V \times V$  — базовый, приобретающий и высвобождающий фронты потока;  $P \subset Msg$  — множество сообщений, которые были обещаны потоком, но еще не выполнены.

Как было отмечено выше, обещающая модель памяти задана на помеченной системе переходов, а не на программах непосредственно. Сами метки в системе переходов могут быть: (1) операцией чтения из локации  $\ell$  значения  $v(R(\ell, v))$ ; (2) операцией записи в локацию  $\ell$  значения  $v(W(\ell, v))$ ; (3) барьером памяти с модификатором  $fmod$  ( $F(fmod)$ ) или (4) внутреннему переходу ( $\epsilon$ ). Последний тип описывает действия потока, которые не затрагивают память, например, присваивание в локальную переменную и исполнение оператора условного перехода.

Перед исполнением любой программы память обещающей машины состоит из записей во все локации,  $M^{init} \triangleq \{ \langle \ell : 0 @ 0, view^{init} \mid \ell \in Loc \rangle \}$ , где



$$\begin{array}{c}
\text{(READ-HELPER)} \\
\frac{\text{cur}(\ell) \leq t \quad \text{cur}' = \text{cur} \sqcup [\ell@t] \quad \text{acq}' = \text{acq} \sqcup \text{cur}'}{\langle \text{cur}, \text{acq}, \text{rel} \rangle \xrightarrow{\text{R}(\ell, \tau, R)} \langle \text{cur}', \text{acq}', \text{rel} \rangle} \\
\\
\text{(READ)} \\
\frac{\langle \ell : v@t, R \rangle \in M \quad \sigma \xrightarrow{\text{R}(\ell, v)} \sigma' \quad \mathcal{V} \xrightarrow{\text{R}(\ell, \tau, R)} \mathcal{V}'}{\langle \langle \sigma, \mathcal{V}, P \rangle, M \rangle \xrightarrow{\text{Promise } tid} \langle \langle \sigma', \mathcal{V}', P \rangle, M \rangle} \\
\\
\text{(ACQ-FENCE)} \\
\frac{\sigma \xrightarrow{\text{F}(\text{acq})} \sigma'}{\langle \langle \sigma, \langle \text{cur}, \text{acq}, \text{rel} \rangle, P \rangle, M \rangle \xrightarrow{\text{Promise } tid} \langle \langle \sigma', \langle \text{acq}, \text{acq}, \text{rel} \rangle, P \rangle, M \rangle} \\
\\
\text{(SILENT)} \\
\frac{\sigma \xrightarrow{\epsilon} \sigma'}{\langle \langle \sigma, \mathcal{V}, P \rangle, M \rangle \xrightarrow{\text{Promise } tid} \langle \langle \sigma', \mathcal{V}, P \rangle, M \rangle} \\
\\
\text{(GLOBAL)} \\
\frac{\langle TS, M \rangle \xrightarrow{\text{Promise } tid} \langle TS', M' \rangle}{\langle \mathcal{TS}[tid \mapsto TS], M \rangle \xrightarrow{\text{Promise}} \langle \mathcal{TS}[tid \mapsto TS'], M' \rangle}
\end{array}$$

$$\begin{array}{c}
\text{(WRITE-HELPER)} \\
\frac{\text{cur}(\ell) < t \quad R = \text{rel} \sqcup [\ell@t] \quad \text{cur}' = \text{cur} \sqcup [\ell@t] \quad \text{acq}' = \text{acq} \sqcup \text{cur}'}{\langle \text{cur}, \text{acq}, \text{rel} \rangle \xrightarrow{\text{W}(\ell, \tau, R)} \langle \text{cur}', \text{acq}', \text{rel} \rangle} \\
\\
\text{(WRITE)} \\
\frac{\sigma \xrightarrow{\text{W}(\ell, v)} \sigma' \quad \mathcal{V} \xrightarrow{\text{W}(\ell, \tau, R)} \mathcal{V}' \quad m = \langle \ell : v@t, R \rangle \quad \forall v', \text{view}'. \langle \ell : v'@t, \text{view}' \rangle \notin M \quad M' = M \cup \{m\} \quad P' = P \setminus \{m\}}{\langle \langle \sigma, \mathcal{V}, P \rangle, M \rangle \xrightarrow{\text{Promise } tid} \langle \langle \sigma', \mathcal{V}', P' \rangle, M' \rangle} \\
\\
\text{(REL-FENCE)} \\
\frac{\sigma \xrightarrow{\text{F}(\text{rel})} \sigma'}{\langle \langle \sigma, \langle \text{cur}, \text{acq}, \text{rel} \rangle, \emptyset \rangle, M \rangle \xrightarrow{\text{Promise } tid} \langle \langle \sigma', \langle \text{cur}, \text{acq}, \text{cur} \rangle, \emptyset \rangle, M \rangle} \\
\\
\text{(PROMISE)} \\
\frac{\forall v, \text{view}. \langle m.\ell : v@m.\tau, \text{view} \rangle \notin M \quad P' = P \cup \{m\} \quad M' = M \cup \{m\} \quad \forall \ell. \exists v, \text{view}. \langle \ell : v@m.\text{view}(\ell), \text{view} \rangle \in M'}{\langle \langle \sigma, \mathcal{V}, P \rangle, M \rangle \xrightarrow{\text{Promise } tid} \langle \langle \sigma, \mathcal{V}, P' \rangle, M' \rangle}
\end{array}$$

#### Переходы обещающей машины

$\text{view}^{\text{init}} \triangleq \lambda \ell. 0$ . В целом, изначальное состояние машины выглядит следующим образом:

$$\begin{aligned}
\mathbf{p}^{\text{init}} &\triangleq \langle \lambda tid. \langle \sigma = \sigma^{\text{init}}, \\
V &= \langle \text{view}^{\text{init}}, \text{view}^{\text{init}}, \text{view}^{\text{init}} \rangle, P = \emptyset \rangle, M^{\text{init}} \rangle.
\end{aligned}$$

Перейдем к описанию шагов исполнения в обещающей машине (см. рисунок). Единственным правилом, оперирующим на всем состоянии машины, является правило (GLOBAL). В версии обещающей машины без сертификации [10] оно не несет смысловой нагрузки, а только позволяет записать остальные правила локально для потоков (см. описание ниже).

*Выполнение приобретающего барьера (ACQ-FENCE)* делает базовый фронт пото-

ка  $\text{view}^{\text{cur}}$  равным приобретающему фронту потока  $\text{view}^{\text{acq}}$ , где приобретающий фронт — объединение фронтов сообщений<sup>3</sup>, прочитанных потоком к этому моменту, и меток времени записей, произведенных потоком.

*Выполнение высвобождающего барьера (REL-FENCE)* делает высвобождающий фронт потока  $\text{view}^{\text{rel}}$  равным базовому фронту потока  $\text{view}^{\text{cur}}$ . Как результат, фронты сообщений, которые поток добавит в память после выполнения высвобождающего барьера, будут содержать информацию о записях, произведенных потоком

<sup>3</sup> Под объединением фронтов  $A$  и  $B$  мы понимаем фронт  $C \triangleq \lambda \ell. \max(A(\ell), B(\ell))$ .

до выполнения высвобождающего барьера. Кроме того, переход, соответствующий высвобождающему барьеру, имеет дополнительное ограничение: в момент его выполнения у потока не должно быть невыполненных обещаний, т. е.  $P$  должен быть равен пустому множеству.

**Чтение из локации  $\ell$  (*READ*)** поток выполняет следующим образом. Поток выбирает некоторое сообщение  $\langle \ell : val @ \tau, view \rangle$  из памяти машины. При этом метка времени записи  $\tau$  должна быть больше, чем значение базового фронта потока  $view^{cur}(\ell)$ , а само сообщение не должно быть среди обещанных, но не выполненных потоком сообщений  $P$ . Это правило увеличивает базовый фронт потока на  $[\ell @ \tau]$ , а приобретающий — на  $view$ .

**Обещание записи  $\langle \ell : val @ \tau, view \rangle$  (*PROMISE*)** добавляет сообщение в память машины и во множество  $P$ . При этом целевая локация  $\ell$  и значение  $val$  могут быть произвольными, т. е. они не зависят от локального состояния потока  $\sigma$ . Метка времени должна быть уникальной среди сообщений локации  $\ell$ , уже находящихся в памяти. Этот переход не обновляет фронты потока. Также легко заметить, что этот переход существенно недетерминирован, однако он ограничен сертификацией.

**Выполнение обещания  $\langle \ell : val @ \tau, view \rangle$  (*WRITE*)** удаляет сообщение из множества невыполненных обещаний  $P$ . При этом данный переход должен быть возможен в рамках помеченной системы переходов локальных состояний, т. е. должно существовать состояние  $\sigma'$ , такое что  $\sigma$  связан с  $\sigma'$  переходом  $W(\ell, val)$ . При этом метка времени  $\tau$  должна быть больше значения базового фронта  $view^{cur}(\ell)$ , и фронт сообщения  $view$  должен быть равен композиции высвобождающего фронта  $view^{rel}$  и  $[\ell @ \tau]$ . Переход также увеличивает базовый и приобретающий фронты потока на  $[\ell @ \tau]$ .

**Внутренний переход (*SILENT*)** соответствует шагу исполнения потока, который не взаимодействует с памятью — присваивание в локальную переменную, исполнение условного перехода или пустой операции. Данный переход меняет только локальное состояние потока.

## 6. Формальное определение модели ARMv8.3

В этом разделе модель ARMv8.3 описывается формально, в соответствии с [20]. Мы не будем явным образом использовать язык ARM-ассемблера и будем считать, что он совпадает с исходным языком с точностью до модификаторов барьера памяти:

$$fmod_{ARM} ::= sy \mid ld.$$

Также мы будем считать, что результат компиляции любой программы *Prog* на исходном языке является той же программой, в которой модификаторы *rel* заменены на *su*, а *acq* — на *ld*.

**Определение.** Исполнение  $G$  — это граф, который состоит из следующих компонент.

1. Конечное множество событий  $E \subseteq \mathbb{Z}$ , которое включает выделенное множество начальных событий  $E_0 = \{a_0^x \mid x \in Loc\}$ . Мы используем  $a$ ,  $b$ , ... как переменные для обозначения событий.

2. Функция  $tid$ , которая по событию из  $E$  возвращает номер потока, породившего событие. При этом начальные события  $a_0^x \in E_0$  мы считаем относящимися к потоку с номером 0,  $tid(a_0^x) = 0$ .

3. Функция  $lab$ , которая присваивает метку каждому событию из  $E$ . Метки могут обозначать следующие операции:

- чтение,  $R(x, v)$ , где  $x \in Loc$  — локация, из которой событие читает значение  $v \in Val$ ;
- запись,  $W(x, v)$ , где  $x \in Loc$  — локация, в которую событие записывает значение  $v \in Val$ ;
- барьер памяти,  $F(o)$ , где  $o \in \{ld, sy\}$  — тип барьера, причем *sy*-барьеры строже, чем *ld*,  $ld \sqsubseteq sy$ .

Каждое начальное событие — это инициализирующая запись в некоторую локацию  $\forall a_0^x \in E_0. lab(a_0^x) = W(x, 0)$ . Функция  $lab$  естественным образом определяет следующие частично определенные функции, которые по событиям возвращают:

- $typ$  — тип события (*R*, *W* или *F*);
- $mod$  — тип барьера;
- $loc$  — целевую локацию;
- $val_r$  — прочитанное значение;

- $\text{val}_w$  — записанное значение.

Далее мы будем использовать  $R$ ,  $W$  и  $F$  также и для обозначения соответствующих множеств событий (например,  $R$  для обозначения  $\{e \in E \mid \text{typ}(e) = R\}$ ).

4. Строгий частичный порядок на событиях  $\text{po} \subseteq E \times E$ , называемый программным порядком, который является объединением непересекающихся множеств  $\{\text{po}_i\}_{i \in \{0\} \cup \text{Tid}}$ , где  $\text{po}_0 = E_0 \times (E \setminus E_0)$  и для каждого потока  $i \in \text{Tid}$  отношение  $\text{po}_i$  является полным порядком на множестве событий этого потока  $\{a \in E \mid \text{tid}(a) = i\}$ .

5. Отношения  $\text{data}, \text{addr}, \text{ctrl} \subseteq \text{po} \setminus \text{po}_0$ , которые удовлетворяют следующим ограничениям:

- $\text{data} \subseteq R \times W$ ;
- $\text{addr} \subseteq R \times (R \cup W)$ ;
- $\text{ctrl}; \text{po} \subseteq \text{ctrl}$ .

Они представляют зависимости по данным, по целевому адресу инструкции и по потоку управления соответственно.

6. Отношение  $\text{rf} \subseteq [W];_{\text{loc}};[R]$ , которое

$$\begin{aligned} \text{obs} &\triangleq \text{rfe} \cup \text{rbe} \cup \text{coe} && (\text{observed-by}) \\ &(\text{addr} \cup \text{data}); \text{rfi}^? \cup \\ \text{dob} &\triangleq (\text{ctrl} \cup \text{data}); [W]; \text{coi}^? \cup && (\text{dependency-ordered-before}) \\ &\text{addr}; \text{po}; [W] \\ \text{bob} &\triangleq \text{po}; [F(\text{sy})] \cup [F(\text{sy})]; \text{po} \cup && (\text{barrier-ordered-before}) \\ &[R]; \text{po}; [F(\text{ld})] \cup [F(\text{ld})]; \text{po} \end{aligned}$$

**Определение.** Исполнение  $G$  является **ARM-согласованным**, если выполняется следующее:

- $R = \text{codom}(\text{rf})$ ; (COMPLETENESS)
- отношение  $\text{po}|_{\text{loc}} \cup \text{rb} \cup \text{co} \cup \text{rf}$  не имеет циклов; (INTERNAL)
- отношение  $\text{obs} \cup \text{dob} \cup \text{bob}$  не имеет циклов. (EXTERNAL)

## 7. Обход ARM-согласованных исполнений

В разделе 4 мы обсудили, что наше доказательство корректности компиляции представляет собой симуляцию обещающей машиной некоторого обхода ARM-согласованных исполнений. В этом разделе мы введем данный обход. Для этого мы сначала определим несколько вспомогательных понятий, потом формально введем шаг обхода с требуемыми ограничениями,

удовлетворяет следующим ограничениям:

- $\text{val}_w(a) = \text{val}_r(b)$  для любой пары  $\langle a, b \rangle \in \text{rf}$ ;

- $a_1 = a_2$ , если  $\langle a_1, b \rangle, \langle a_2, b \rangle \in \text{rf}$ .

Мы также вводим функцию  $\text{rf}^{-1}$ , которая определена на области отображения  $\text{rf}$ ,  $\text{codom}(\text{rf})$ , и для любого  $a$  из  $\text{codom}(\text{rf})$  верно, что  $\langle \text{rf}^{-1}(a), a \rangle \in \text{rf}$ .

7. Строгий частичный порядок  $\text{co}$  на элементах  $W$ , который является объединением отношений  $\{\text{co}_x\}_{x \in \text{Loc}}$ , где  $\text{co}_x$  — полный порядок на элементах  $W_x = \{e \in W \mid \text{loc}(e) = x\}$ . Мы также используем некоторые производные отношения. Так,  $\text{rb} \triangleq \text{rf}^{-1}$ ;  $\text{co}$  — это отношение связывает событие чтения с событием записи, которое  $\text{co}$ -следует за прочитанным событием записи. С помощью суффиксов  $i$  и  $e$  обозначаются подмножества отношений, которые связывают события одного и разных потоков соответственно. Например,  $\text{coi} = \{\langle w, w' \rangle \in \text{co} \mid \text{tid}(w) = \text{tid}(w')\}$  и  $\text{rfe} = \{\langle w, r \rangle \in \text{rf} \mid \text{tid}(w) \neq \text{tid}(r)\}$ .

а потом докажем существование последовательности шагов, которые полностью обходят любое ARM-согласованное исполнение.

**Определение.** Конфигурацией обхода для ARM-согласованного исполнения  $G$  называется пара множеств  $\langle C, I \rangle$ , таких что

- $C \subseteq G.E$ ;
- $\text{dom}(G.\text{po};[C]) \subseteq C$ ;
- $C \cap G.W \subseteq I \subseteq W$ .

Элементы  $C$  мы будем называть покрытыми (covered), а  $I$  — выпущенными (issued).

Интуитивно, в рамках симуляции покрытые события будут соответствовать префиксу программы, который исполнен обещающей семантикой, тогда как выпущенные — записями, находящимися в памяти обещающей машины в соответствующий момент исполнения.

**Определение.** Множеством следующих событий  $\text{Next}(G, C)$  для исполнения  $G$  и его покрытия  $C \subseteq G.E$  называется множество, состоящее из событий, все ро-предшествующие события которых уже покрыты:

$$\begin{aligned} \text{Next}(G, C) &\triangleq \\ &\triangleq \{b \in G.E \mid \text{dom}(G.\text{po}; [b]) \subseteq C\} \setminus C. \end{aligned}$$

В рамках симуляции данное множество будет содержать по одному событию для каждого еще не заверщенного потока в обещающей семантике, причем события будут соответствовать следующему переходу в потоке, который не является обещанием. Так, если бы обещающая семантика была определена поверх некоторого синтаксиса, то множество следующих событий соответствовало инструкциям, на которые указывают счетчики команд (**program counter**) потоков в обещающей модели.

**Определение.** Событие  $w$  является *покрываемым* для исполнения  $G$  в конфигурации обхода  $\langle C, I \rangle$ ,  $a \in \text{Coverable}(G, C, I)$ ,

- $w$  является событием записи ( $w \in G.W$ );
- все ро-предшествующие барьеры покрыты ( $\text{dom}([F]; G.\text{po}; [w]) \subseteq C$ );
- все записи других потоков, от которых зависит  $w$ , выпущены ( $\text{dom}(G.\text{rfe}; G.\text{dob}^+; [w]) \subseteq I$ ).

Шаг обхода, который «выпускает» событие записи, соответствует в симуляции обещанию, которое делает обещающая семантика. Ограничение **WRITE-BOB** является более строгим, чем ограничение из обещающей семантики: обещание может быть сделано «через» приобретающий барьер, который соответствует  $\text{fence}(\text{ld})$  в

$$\frac{a \in \text{Next}(G, C) \cap \text{Coverable}(G, C, I)}{G \vdash \langle C, I \rangle \rightarrow_{\text{TC}} \langle C \cup \{a\}, I \rangle}$$

Здесь левое правило покрывает событие, если событие принадлежит множеству следующих событий и при этом покрываемо в данной конфигурации. Правое правило выпускает событие записи, если событие выпускаемо и при этом еще не выпущено.

Поскольку каждое конкретное ARM-согласованное исполнение является некоторым конечным графом, а приведенные

если  $a$  является:

- событием чтения и связанное событие записи является выпущенным ( $a \in G.R$  и  $\text{rf}^{-1}(a) \in I$ ),
- выпущенным событием ( $a \in I$ ) или
- барьером памяти ( $a \in G.F$ ).

В нашем обходе, заданном как операционная семантика, есть шаг, который «покрывает» событие, т. е. добавляет его во множество покрытых. Событие, покрываемое данным правилом, должно соответствовать приведенным выше ограничениям, которые, в свою очередь, соответствуют ограничениям обещающей семантики. Например, обещающая семантика может исполнить инструкцию чтения и прочитать из некоторого конкретного сообщения только в том случае, если данное сообщение есть в памяти, т. е. соответствующее ему событие  $w$  уже выпущено,  $w \in I$ .

**Определение.** Событие  $w$  называется *выпускаемым* для исполнения  $G$  в конфигурации обхода  $\langle C, I \rangle$ ,  $w \in \text{Issuable}(G, C, I)$ , если выполняется следующее:

ARMv8.3, однако более строгое ограничение позволяет упростить симуляцию. Ограничение **WRITE-DOB** нужно для того, чтобы обещающая семантика могла сертифицировать обещание, которое она делает.

Шаги обхода задаются следующим образом:

$$\frac{w \in \text{Issuable}(G, C, I) \setminus I}{G \vdash \langle C, I \rangle \rightarrow_{\text{TC}} \langle C, I \cup \{w\} \rangle}$$

выше шаги обхода наращивают конфигурации, то для доказательства наличия полного обхода графа достаточно показать, что для каждой неконечной конфигурации существует шаг к новой конфигурации.

**Теорема 7.1.** Пусть  $\langle C, I \rangle$  является конфигурацией ARM-согласованного исполнения  $G$ , при этом конфигурация достижима из начальной ( $G \vdash \langle \emptyset, \emptyset \rangle \xrightarrow{*}_{\text{TC}} \langle C, I \rangle$ )



и  $C \neq G.E$ . Тогда существуют  $C'$  и  $I'$ , т. ч.  $G \vdash \langle C, I \rangle \rightarrow_{TC} \langle C', I' \rangle$ .

Доказательство теоремы основано на следующих вспомогательных леммах:

Лемма 7.2. Пусть  $G \vdash \langle \emptyset, \emptyset \rangle \rightarrow_{TC}^* \langle C, I \rangle$ . Тогда  $C \subseteq \text{Coverable}(G, C, I)$  и  $I \subseteq \text{Issuable}(G, C, I)$ .

Доказательство. Следует из определений  $\text{Coverable}$  и  $\text{Issuable}$  и того, что множества покрытых и выпущенных событий растут в течение обхода  $G$ .

Лемма 7.3. Пусть  $G \vdash \langle \emptyset, \emptyset \rangle \rightarrow_{TC}^* \langle C, I \rangle$ . Тогда  $W \cap \text{Next}(G, C) \subseteq \text{Issuable}(G, C, I)$ .

Доказательство. Зафиксируем  $w \in W \cap \text{Next}(G, C)$  и покажем, что  $w \in \text{Issuable}(G, C, I)$ . Так как  $w \in \text{Next}(G, C)$ , то все  $\text{po}$ -предшествующие события являются покрытыми (т. е. находятся в  $C$ ), из чего напрямую следует, что выполняется  $\text{WRITE-BOB}$ . Кроме того, из леммы 7.1 следует, что все покрытые события являются покрываемыми, из чего следует  $\text{WRITE-DOB}$ .

Доказательство теоремы 7.1. Нам известно, что  $\text{Next}(G, C) \neq \emptyset$ , так как  $C \neq G.E$ . В доказательстве мы рассматриваем два варианта. Первым мы рассматриваем вариант, когда существует элемент  $\text{Next}(G, C)$ , который покрываем, или сначала выпускаем, а потом покрываем. Вторым мы рассматриваем вариант, когда все элементы в  $\text{Next}(G, C)$  не покрываемы и не выпускаемы. Для этой ситуации мы показываем, что в  $G$  существует событие записи, которое выпускаемо в данной конфигурации.

Первый вариант. Зафиксируем элемент  $a \in \text{Next}(G, C)$ . Если  $a \in R$  и  $\text{rf}^{-1}(a) \in I$ , или  $a \in F$ , или  $a \in W \cap I$ , тогда  $a$  покрываем по определению. Если  $a \in W \setminus I$ , тогда  $a$  — выпускаемо по лемме 7.3.

Второй вариант. В этом случае мы предполагаем, что нет события из  $\text{Next}(G, C)$ , которое покрываемо или выпускаемо. Таким образом,  $\text{Next}(G, C) \subseteq R$  является следствием леммы 7.2 и определения покрываемого события. Кроме того, для любого чтения события  $r$  из  $\text{Next}(G, C)$  мы знаем, что  $\text{rf}^{-1}(r) \notin I$ . Далее мы покажем, что существует событие записи, которое выпускаемо в данной конфигурации. Для этого введем вспомогательное отно-

шение  $\text{eord} \triangleq (\text{obs} \cup \text{dob} \cup \text{bob})^+$ , которое антирефлексивно по определению  $\text{ARM}$ -согласованности  $\text{EXTERNAL}$ .

Мы знаем, что есть как минимум одно событие (чтения) из  $\text{Next}(G, C)$ , которое при этом не покрываемо. Это означает, что существует событие записи, которое еще не выпущено. Выберем событие записи  $w \in W \setminus I$ , которое является минимальным по отношению  $\text{eord}$  среди записей, которые еще не выпущены, то есть  $\forall w' \in W \setminus I. \text{eord}(w', w)$ . Осталось показать, что событие  $w$  выпускаемо.

Так как  $w \notin \text{Next}(G, C)$ , то существует событие чтения  $r \in \text{Next}(G, C)$ , такое что  $\text{po}(r, w)$  и  $\text{rf}^{-1}(r) \notin I$ . При этом  $\text{rf}^{-1}(r) = \text{rfe}^{-1}(r)$ , так как  $C \cap W \subseteq I$  и  $\forall e \in C.\text{dom}(\text{po};[e]) \in C$ . Для доказательства того, что  $w$  выпускаемо, нужно показать, что два утверждения выполняются.

**WRITE-BOB:** Пусть  $f \in F$ , такое что  $\text{po}(f, w)$ . Предположим, что  $f \notin C$ . Тогда  $\text{po}(r, f)$  и  $\langle \text{rfe}^{-1}(r), w \rangle \in \text{obs}; \text{bob}^+ \subseteq \text{eord}$ . Так как  $\text{rfe}^{-1}(r) \notin I$ , то существует  $\text{eord}$ -предшествующее  $w$  событие записи, которое не выпущено. Это противоречит выбору  $w$ .

**WRITE-DOB:** Пусть существует событие чтения  $r'$ , такое что  $\text{dob}^+(r', w)$ . Если  $\text{rfe}^{-1}(r') \neq \perp$ , то  $\langle \text{rfe}^{-1}(r'), w \rangle \in \text{rfe}; \text{dob}^+ \subseteq \text{obs}; \text{dob}^+ \subseteq \text{eord}$ . По определению  $w$  это означает, что  $\text{rfe}^{-1}(r') \in I$ . ■

## 8. Симуляция обхода обещающей машиной

В этом разделе мы приводим доказательство основной теоремы 4.1.

Теорема 4.1. Для любых программы  $\text{Prog}$ , результата ее компиляции  $\text{Prog}_{\text{ARM}}$  и  $\text{ARM}$ -согласованного исполнения  $G$  программы  $\text{Prog}_{\text{ARM}}$  существует исполнение  $\text{Prog}$  обещающей машиной, т. ч. финальное состояние памяти машины совпадает с состоянием памяти  $G$ .

Доказательство. Зафиксируем  $\text{ARM}$ -согласованное исполнение  $G$ . Далее мы покажем, что обещающая семантика может симулировать некоторый обход исполнения  $G$ , который существует согласно теореме 7.1.

Почему нам достаточно рассмотреть некоторый обход исполнения  $G$ ? Это так, по-



сколько сам обход задает нам лишь схему индукции по исполнению  $G$  и не влияет на финальное состояние памяти  $G$ , поскольку оно полностью определяется отношением  $G.co$ .

Для того чтобы показать симуляцию между обещающей семантикой и обходом, мы введем отношение симуляции  $\mathcal{I}$ , которое будет связывать конфигурацию обхода  $G$  с состоянием обещающей машины. Формально утверждение о наличии симуляции выглядит следующим образом.

**Лемма 8.1.** Существуют такие  $\mathcal{TS}$  и  $M$ , что  $\mathcal{I}(G.E, G.W, \mathcal{TS}, M)$  и  $\langle \mathcal{TS}_{init}, M_{init} \rangle \rightarrow_{\text{Promise}}^* \langle \mathcal{TS}, M \rangle$ .

При этом мы определим отношение симуляции  $\mathcal{I}$  так, чтобы из него было очевидно, что финальное состояние памяти  $M$  совпадает с состоянием памяти исполнения  $G$ , а также, что  $\mathcal{I}(\emptyset, \emptyset, \mathcal{TS}_{init}, M_{init})$  выполняется. Тогда доказательство теоремы будет непосредственно следовать из доказательства леммы 8.1. Сама же лемма доказывается индукцией с использованием леммы 8.2, которая показывает, что для любой неполной конфигурации обхода существует шаг, который обещающая машина может симулировать.

**Лемма 8.2.** Пусть для некоторых конфигураций обхода  $\langle C, I \rangle$  и  $\langle C', I' \rangle$  исполнения  $G$ , а также некоторого состояния обещающей машины  $\langle \mathcal{TS}, M \rangle$  выполняется  $G \vdash \langle C, I \rangle \rightarrow_{\text{TC}} \langle C', I' \rangle$  и  $\mathcal{I}(C, I, \mathcal{TS}, M)$ . Тогда существуют  $\mathcal{TS}'$ ,  $M'$ , такие что  $\langle \mathcal{TS}, M \rangle \rightarrow_{\text{Promise}}^+ \langle \mathcal{TS}', M' \rangle$  и  $\mathcal{I}(C', I', \mathcal{TS}', M')$ .

Далее в этом разделе мы опишем вспомогательные отношения на графах исполнения, которые будут нужны для выражения связи с фронтами обещающей модели в отношении симуляции, и само отношение симуляции. Как и в доказательстве теоремы 4.1, мы зафиксируем ARM-согласованное исполнение  $G$  и будем вводить все определения в контексте  $G$ .

**8.1 Аналогии фронтов для ARM-согласованных исполнений.** Введем вспомогательную функцию  $T : \text{TimeMap} = W \rightarrow \mathbb{N}$ , сопоставляющую событиям записи в  $G$  их порядковые номера в отношении  $co$  так, чтобы  $\text{correct-tmap}(G, T)$  выполнялось:

$$\begin{aligned} \text{correct-tmap}(G, T) &\triangleq \forall w, w'. \\ &(\langle w, w' \rangle \in co \Rightarrow T(w) < T(w')) \wedge \\ &(w \notin \text{codom}(co) \Rightarrow T(w) = 0). \end{aligned}$$

Эта функция фактически вводит метки времени на событиях записи  $G$ . Далее введем вспомогательные отношения  $sw$  (**s**yn-**c**hronized-with) и  $hb$  (**h**appens-before):

$$\begin{aligned} sw &\triangleq [F(\text{sy})]; po; rf; po; [F(\text{ld})] \\ hb &\triangleq (sw \cup po)^+. \end{aligned}$$

Здесь отношение  $sw$  представляет пути в графе, соответствующие передачи информации о событиях записи с помощью ARM-аналогов высвобождающего ( $F(\text{sy})$ ) и приобретающего ( $F(\text{ld})$ ) барьеров памяти. Отношение  $hb$  мы далее используем, чтобы определить отношения  $sig\text{-rel}$ ,  $acq\text{-rel}$  и  $rel\text{-rel}$ . Они в паре с функцией  $T$  будут использованы в симуляции как ограничения для базового, приобретающего и высвобождающего фронтов потока в обещающей модели памяти:

$$\begin{aligned} \text{cur-rel} &\triangleq rf^?; hb \\ \text{acq-rel} &\triangleq rf^?; hb; ([F(\text{sy})]; po; rf; po)^? \\ \text{rel-rel} &\triangleq rf^?; hb; [F(\text{sy})]; po. \end{aligned}$$

Аналогичным образом отношение  $\text{msg-rel} \subseteq W \times W$  представляет ограничение на фронт сообщения из обещающей модели. Так, если  $\langle w, w' \rangle \in \text{msg-rel}$  и события  $w$  и  $w'$  соответствуют сообщениям  $m$  и  $m'$  в памяти обещающей семантики, то  $m.\tau = T(w)$ ,  $m'.\tau = T(w')$  и  $m'.\text{view}(\text{loc}(w)) \geq m.\tau$

$$\text{msg-rel} \triangleq rf^?; hb; [F(\text{sy})]; po \cup [W].$$

**8.2 Отношение симуляции  $\mathcal{I}$ .** Отношение симуляции мы определяем следующим образом:

$$\begin{aligned} \mathcal{I}(C, I, \mathcal{TS}, M) &\triangleq \mathcal{I}_{\text{mem1}}(C, I, \mathcal{TS}, M) \wedge \\ &\wedge \mathcal{I}_{\text{mem2}}(C, I, M) \wedge \mathcal{I}_{\text{view}}(C, \mathcal{TS}) \wedge \\ &\wedge \mathcal{I}_{\text{view-rel}}(\mathcal{TS}) \wedge \mathcal{I}_{\text{state}}(C, \mathcal{TS}). \end{aligned}$$

$\mathcal{I}_{\text{mem1}}(C, I, \mathcal{TS}, M)$  утверждает, что все выпущенные события  $I$  имеют аналоги в памяти обещающей машины, причем выполненные обещания соответствуют по-

крытым событиям записи:

$$\mathcal{I}_{\text{mem1}}(C, I, TS, M) \triangleq \forall w \in I. \exists \text{view} \leq \text{dom-view}(\text{msg-rel}; [w]).$$

$$\text{let } \text{msg} \triangleq \langle \text{loc}(w) : \text{val}_w(w) @ T(w), \text{view} \rangle \text{ in}$$

$$\text{let } P \triangleq TS(\text{tid}(w)).P \text{ in}$$

$$(w \in C \Rightarrow \text{msg} \in M \setminus P) \wedge (w \notin C \Rightarrow \text{msg} \in P).$$

Здесь  $\text{dom-view}$  является вспомогательной функцией, которая по множеству записей строит соответствующий множеству фронт:

$$\text{set-view}(S) \triangleq \lambda \ell. \max\{T(w) \mid w \in S, \text{loc}(w) = \ell\};$$

$$\text{dom-view}(R) \triangleq \text{set-view}(\text{dom}(R)).$$

$\mathcal{I}_{\text{mem2}}(C, I, M)$  обозначает обратную связь: для каждого сообщения из  $M$  существует соответствующее ему выпущенное событие в  $I$ :

$$\mathcal{I}_{\text{mem2}}(C, I, M) \triangleq \forall \langle \ell : v @ \tau, \text{view} \rangle \in M. \tau \neq 0 \Rightarrow \exists w \in I.$$

$$\ell = \text{loc}(w) \wedge v = \text{val}_w(w) \wedge \tau = T(w) \wedge \text{view} \leq \text{dom-view}(\text{msg-rel}; [w]).$$

$\mathcal{I}_{\text{view}}(C, TS)$  утверждает, что для любого элемента множества следующих событий ( $\text{Next}(G, C)$ ) фронты, представляющие связанные с ним отношения  $\text{sig-rel}$ ,  $\text{acq-rel}$  и  $\text{rel-rel}$  события записи, больше, чем базовый, приобретающий и высвобождающие фронты соответствующего потока:

$$\mathcal{I}_{\text{view}}(C, TS) \triangleq \forall e \in \text{Next}(G, C).$$

$$\text{let } \langle \text{cur}, \text{acq}, \text{rel} \rangle \triangleq TS(\text{tid}(e)).V \text{ in}$$

$$\text{cur} \leq \text{dom-view}(\text{cur-rel}; [e]) \wedge$$

$$\text{acq} \leq \text{dom-view}(\text{acq-rel}; [e]) \wedge$$

$$\text{rel} \leq \text{dom-view}(\text{rel-rel}; [e]).$$

$\mathcal{I}_{\text{view-rel}}(TS)$  показывает, что все еще не выполненные обещания имеют специальную форму их фронтов — значение высвобождающего фронта соответствующего потока плюс метка времени самого обещания:

$$\mathcal{I}_{\text{view-rel}}(TS) \triangleq \forall \text{tid}, \langle \ell : \_ @ \tau, \text{view} \rangle \in TS(\text{tid}).P.$$

$$\text{view} = [\ell @ \tau] \sqcup TS(\text{tid}).V.\text{rel}.$$

$\mathcal{I}_{\text{state}}(C, TS)$  утверждает, что для каждого потока  $\text{tid}$  существует список переходов  $\{t_i\}_{i \in [1..k]}$ , который соответствует событиям в

$\text{tid}$ -подграфе исполнения  $G$ , и текущее состояние потока  $TS(\text{tid}).\sigma$  получено с помощью  $p$  переходов из списка, где  $p$  — количество покрытых событий,  $|E_{\text{tid}} \cap C|$ .

$$\mathcal{I}_{\text{state}}(C, TS) \triangleq \forall \text{tid},$$

$$k = |E_{\text{tid}}| \cdot \exists \{\sigma_i\}_{i \in [0..k]}, \{t_i\}_{i \in [1..k]}.$$

$$\text{let } p \triangleq |E_{\text{tid}} \cap C| \text{ in}$$

$$TS(\text{tid}).\sigma =$$

$$= \sigma_p \wedge (\forall j \in [0..k-1]. \sigma_j \xrightarrow{\varepsilon^*} \sigma_{j+1}) \wedge$$

$$\forall n \in [1..k], a \in \text{nth}([E_{\text{tid}}]; p);$$

$$[E_{\text{tid}}](n-1).t_{n+1} \approx \text{lab}(a).$$

Здесь функция  $\text{nth } p \text{ order } n$  возвращает элементы с номером  $n$  из отношения частичного порядка  $p \text{ order}$ , а предикат  $t \approx \text{lab}(a)$  выполняется тогда и только тогда, когда метка перехода  $t$  соответствует метке события  $a$ .

## 9. Связанные работы

Корректность компиляции в общем смысле является большой, проработанной областью, которая продолжает развиваться. Так, недавно были представлены верифицированные компиляторы для языков C, CompCert [15] и ML, CakeML [11]. Эти компиляторы для однопоточных программ, но существуют и компиляторы для многопоточных программ в архитектуры со слабыми моделями памяти, в частности из языка C в архитектуру x86-TSO [22]. В работах [5, 6] приведены доказательства корректности компиляции из аксиоматической модели C/C++11 [7] в модели архитектур x86-TSO и Power.

Наиболее близкими работами к нашей являются [10] и [19]. В [10] приведены доказательства корректности компиляции из обещающей модели в модели памяти x86-TSO [23] и Power [3], которые, так же как и [20], являются аксиоматическими. Доказательства для моделей x86-TSO и Power имеют следующую структуру: (1) обе модели памяти, x86-TSO и Power, представляются как некоторые более строгие модели и набор трансформаций над программами; (2) приводится доказательство того, что трансформации корректны в рамках

обещающей модели памяти, т. е. поведения трансформированной программы являются поведениями изначальной программы в обещающей модели памяти; (3) показывается корректность компиляции для более строгих версий моделей.

Первый пункт этого доказательства приведен в [12]. Так, каждое поведение некоторой программы  $A$  в рамках модели x86-TSO является поведением программы  $B$  в рамках модели последовательной консистентности, где  $B$  может быть получена из  $A$  путем (многократного) применения двух трансформаций — переупорядочивание инструкции записи с непосредственно следующей операцией чтения из другой локации и удаления повторного чтения:

$$\begin{array}{ll} [x] := 1; & \rightarrow \quad a := [y]; \\ a := [y]; & [x] := 1; \\ \\ a := [y]; & \rightarrow \quad a := [y]; \\ b := [y]; & b := a \end{array}$$

Для модели Power приведен аналогичный результат: каждое поведение программы  $A$  в рамках модели Power является поведением программы  $B$  в рамках более строгой модели StrongPower, запрещающей часть поведения, наблюдаемых в модели Power. Здесь  $B$  может быть получена из  $A$  путем (многократного) применения трансформации, которая переупорядочивает произвольные независимые инструкции чтения или записи, следующие друг за другом.

Мы пытались применить аналогичный подход для доказательства корректности компиляции для модели ARMv8.3 [20], но отказались от него ввиду того, что существует программа, одно из поведения которой разрешено моделью ARMv8.3 и не выразимо с помощью ранее использованных трансформаций кода над более строгой моделью памяти. Упомянутая программа выглядит следующим образом:

$$\begin{array}{l} a := [x]; //1 \\ \text{if } a = 1 \text{ then} \\ b := [y]; //0 \end{array} \parallel \begin{array}{l} [y] := 1; \\ \text{fence(sy);} \\ [x] := 1 \end{array}$$

В левом потоке между операциями чтения есть зависимость по управлению, а в

правом потоке используется барьер памяти, который запрещает исполнение записей не по порядку. Модель памяти ARMv8.3 разрешает программе завершиться с  $a = 1$  и  $b = 0$ , при этом инструкции в обоих потоках не могут быть переставлены ввиду наличия зависимости и барьера.

В работе [19] приведено доказательство корректности компиляции из обещающей модели в модель ARM POP [9]. Несмотря на то, что этот результат выглядит очень близким к тому, что предлагается в данной статье, существует важное различие: модель ARM POP представлена в операционном стиле. Последнее позволило авторам [19] провести доказательство через непосредственную симуляцию модели ARM POP обещающей моделью. При этом модель ARM POP обладает более широкими возможностями по исполнению инструкций не по порядку по сравнению с обещающей моделью. Для решения этой проблемы в [19] используется т. н. *запаздывающая симуляция*, которая позволяет обещающей семантике в рамках симуляции повторять действия модели ARM POP в возможном для обещающей семантики порядке. Наша операционная семантика обхода графа исполнения использует похожую идею, но в приложении к аксиоматической модели ARMv8.3.

## 10. Заключение

В данной статье приведено доказательство корректности компиляции для подмножества обещающей модели памяти [10], состоящего из ослабленных операций чтения и записи, приобретающих и высвобождающих барьеров памяти, в модель памяти ARMv8.3 [20]. Доказательство базируется на новой идее обхода исполнений в аксиоматических семантиках, который может симулировать обещающая модель памяти. Мы убеждены в том, что данная идея может использоваться для прямых доказательств корректности компиляции в другие модели памяти, а подобные задачи регулярно появляются ввиду бурного развития области.

Несмотря на то, что подмножество обещающей модели, рассмотренное в статье, достаточно ограничено, доказательство кор-

ректности компиляции для него достаточно сложно и объемно. В наши дальнейшие планы входит расширение доказательства до полной обещающей модели памяти. Для этого нужно будет поддержать операции приобретающего чтения (**read acquire**) и высвобождающей записи (**write release**), атомарные инструкции чтения и записи (**read-modify-write**), частным случаем которых является сравнение с обменом (**compare-and-set** – **CAS**), а также полные барьеры памяти

(**SC fences**). Поддержка данных инструкций потребует небольшого усложнения обхода ARM-согласованных исполнений, а также существенного усложнения отношения симуляции ввиду большого количества мелких деталей, связанных с упомянутыми инструкциями в рамках обещающей модели памяти.

Работа выполнена при поддержке компании JetBrains (<http://jetbrains.com>).

### СПИСОК ЛИТЕРАТУРЫ

1. ARM architecture reference manual: ARMv8, for ARMv8-A architecture profile // URL: <https://developer.arm.com/docs/ddi0487/latest/arm-architecture-reference-manual-armv8-for-armv8-a-architecture-profile> (Дата обращения: 16.05.2017).
2. C/C++11 mappings to processors // URL: <https://www.cl.cam.ac.uk/~pes20/cpp/cpp0xmappings.html>. (Дата обращения: 16.05.2017).
3. Alglave J., Maranget L., Tautschnig M. Herding cats: Modelling, simulation, testing, and data mining for weak memory // ACM Trans. Program. Lang. Syst. 2014. Vol. 36(2). Pp. 7:1–7:74. DOI: 10.1145/2627752
4. Batty M., Memarian K., Nienhuis K., Pichon-Pharabod J., Sewell P. The problem of programming language concurrency semantics // ESOP. Springer, 2015. Vol. 9032 of LNCS. Pp. 283–307. DOI: 10.1007/978-3-662-46669-8\_12
5. Batty M., Memarian K., Owens Sc., Sarkar S., Sewell P. Clarifying and compiling C/C++ concurrency: From C++11 to POWER // POPL 2012. ACM, 2012. DOI: 10.1145/2103621.2103717.
6. Batty M., Owens Sc., Sarkar S., Sewell P., Weber T. Mathematizing C++ concurrency: The post-Rapperswil model. technical report n3132, iso iec jtc1/sc22/wg21 // URL: <http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2010/n3132.pdf>.
7. Batty M., Owens Sc., Sarkar S., Sewell P., Weber T. Mathematizing C++ concurrency // POPL 2011. ACM, 2011. Pp. 55–66. DOI: 10.1145/1925844.1926394
8. Boehm H.-J., Demsky B. Outlawing ghosts: Avoiding out-of-thin-air results // MSPC 2014. ACM, 2014. Pp. 7:1–7:6. DOI: 10.1145/2618128.2618134
9. Flur Sh., Gray K.E., Pulte Ch., Sarkar S., Sezgin A., Maranget L., Deacon W., Sewell P. Modelling the ARMv8 architecture, operationally: Concurrency and ISA // POPL 2016. ACM, 2016. Pp. 608–621. DOI: 10.1145/2837614.2837615
10. Jecheon Kang, Chung-Kil Hur, Lahav O., Vafeiadis V., Dreyer D. A promising semantics for relaxed-memory concurrency // POPL 2017. ACM, 2017. DOI: 10.1145/3009837. 3009850
11. Kumar R., Myreen M.O., Norrish M., Owens Sc. CakeML: A verified implementation of ML // POPL 2014: Proc. of the 41st ACM SIGPLAN-SIGACT Symp. on Principles of Programming Languages. ACM Press, 2014. Pp. 179–191. DOI: 10.1145/2535838.2535841
12. Lahav O., Vafeiadis V. Explaining relaxed memory models with program transformations // FM 2016. Springer, 2016. DOI: 10.1007/978-3-319-48989-6\_29
13. Lahav O., Vafeiadis V., Jecheon Kang, Chung-Kil Hur, Dreyer D. Repairing sequential consistency in C/C++11 // In PLDI 2017. ACM, 2017.
14. Lamport L. How to make a multiprocessor computer that correctly executes multiprocess programs // IEEE Trans. Computers. 1979. Vol. 28(9). Pp. 690–691. DOI: 10.1109/TC.1979.1675439
15. Leroy X. A formally verified compiler back-end // J. Autom. Reasoning. 2009. Vol. 43(4). Pp. 363–446. URL: <https://doi.org/10.1007/s10817-009-9155-4>. DOI: 10.1007/s10817-009-9155-4
16. Manerkar Ya.A., Trippel C., Lustig D., Pellauer M., Martonosi M. Counterexamples and proof loophole for the C/C++ to POWER and ARMv7 trailing-sync compiler mappings. CoRR, abs/1611.01507, 2016 // URL: <http://arxiv.org/abs/1611.01507>.
17. Manson J., Pugh W., Adve S.V. The Java memory model // POPL 2005. ACM, 2005. Pp. 378–391. DOI: 10.1145/1040305.1040336.
18. Milner R. Communication and concurrency // PHI Series in Computer Science. Prentice Hall, 1989.
19. Podkopaev A., Lahav O., Vafeiadis V. Promising compilation to ARMv8 POP // ECOOP 2017. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2017.



20. Pulte Ch., Flur Sh., Deacon W., French J., Sarkar S., Sewell P. Simplifying ARM concurrency: Multicopy-atomic axiomatic and operational models for ARMv8 // Accepted to POPL'18.

21. Sarkar S., Sewell P., Alglave J., Maranget L., Williams D. Understanding POWER multiprocessors // PLDI 2011. ACM, 2011. Pp. 175–186. DOI: 10.1145/1993498.1993520

22. Sevcik J., Vafeiadis V., Nardelli F.Z., Jagannathan S., Sewell P. Relaxed-memory concurrency and verified compilation // Proc. of the 38th ACM SIGPLAN-SIGACT Symp. on Principles of Programming Languages. POPL

2011. Austin, TX, USA, 2011. Pp. 43–54. URL: <http://doi.acm.org/10.1145/1926385.1926393>. DOI: 10.1145/1926385.1926393

23. Sewell P., Sarkar S., Owens Sc., Nardelli F.Z., Myreen M.O. x86-TSO: A rigorous and usable programmer's model for x86 multiprocessors // Commun. ACM, 2010. Vol. 53(7). Pp. 89–97. DOI: 10.1145/1785414.1785443

24. Vafeiadis V., Balabonski Th., Chakraborty S., Morisset R., Nardelli F.Z. Common compiler optimisations are invalid in the C11 memory model and what we can do about it // POPL 2015. ACM, 2015. Pp. 209–220.

Статья поступила в редакцию 27.09.2017.

## REFERENCES

1. ARM architecture reference manual: ARMv8, for ARMv8-A architecture profile. Available: <https://developer.arm.com/docs/ddi0487/latest/arm-architecture-reference-manual-armv8-for-armv8-a-architecture-profile> (Accessed: 16.05.2017).

2. C/C++11 mappings to processors. Available: <https://www.cl.cam.ac.uk/~pes20/cpp/cpp0xmappings.html>. (Accessed: 16.05.2017).

3. Alglave J., Maranget L., Tautschnig M. Herding cats: Modelling, simulation, testing, and data mining for weak memory. *ACM Trans. Program. Lang. Syst.*, 2014, Vol. 36(2), Pp. 7:1–7:74. DOI: 10.1145/2627752

4. Batty M., Memarian K., Nienhuis K., Pichon-Pharabod J., Sewell P. The problem of programming language concurrency semantics. In *ESOP*. Springer, 2015, Vol. 9032 of LNCS, Pp. 283–307. DOI: 10.1007/978-3-662-46669-8\_12

5. Batty M., Memarian K., Owens Sc., Sarkar S., Sewell P. Clarifying and compiling C/C++ concurrency: From C++11 to POWER. In *POPL 2012*, ACM, 2012. DOI: 10.1145/2103621.2103717

6. Batty M., Owens Sc., Sarkar S., Sewell P., Weber T. *Mathematizing C++ concurrency: The post-Rapperswil model. technical report n3132, iso iec jtc1/sc22/wg21*. Available: <http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2010/n3132.pdf>.

7. Batty M., Owens Sc., Sarkar S., Sewell P., Weber T. Mathematizing C++ concurrency. In *POPL 2011*, ACM, 2011, Pp. 55–66. DOI: 10.1145/1925844.1926394

8. Boehm H.-J., Demsky B. Outlawing ghosts: Avoiding out-of-thin-air results. In *MSPC 2014*, ACM, 2014, Pp. 7:1–7:6. DOI: 10.1145/2618128.2618134

9. Flur Sh., Gray K.E., Pulte Ch., Sarkar S., Sezgin A., Maranget L., Deacon W., Sewell P. Modelling the ARMv8 architecture, operationally: Concurrency and ISA. In *POPL 2016*, ACM, 2016,

Pp. 608–621. DOI: 10.1145/2837614.2837615

10. Jeheon Kang, Chung-Kil Hur, Lahav O., Vafeiadis V., Dreyer D. A promising semantics for relaxed-memory concurrency. In *POPL 2017*. ACM, 2017. DOI: 10.1145/3009837.3009850

11. Kumar R., Myreen M.O., Norrish M., Owens Sc. CakeML: A verified implementation of ML. In *POPL '14: Proceedings of the 41st ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, ACM Press, 2014, Pp. 179–191. DOI: 10.1145/2535838.2535841

12. Lahav O., Vafeiadis V. Explaining relaxed memory models with program transformations. In *FM 2016*. Springer, 2016. DOI: 10.1007/978-3-319-48989-6\_29

13. Lahav O., Vafeiadis V., Jeheon Kang, Chung-Kil Hur, Dreyer D. Repairing sequential consistency in C/C++11. In *PLDI 2017*. ACM, 2017.

14. Lamport L. How to make a multiprocessor computer that correctly executes multiprocess programs. *IEEE Trans. Computers*, 1979, Vol. 28(9), Pp. 690–691. DOI: 10.1109/TC.1979.1675439

15. Leroy X. A formally verified compiler back-end. *J. Autom. Reasoning*, 2009, Vol. 43(4), Pp. 363–446. Available: <https://doi.org/10.1007/s10817-009-9155-4>. DOI: 10.1007/s10817-009-9155-4

16. Manerker Ya.A., Trippel C., Lustig D., Pellauer M., Martonosi M. *Counterexamples and proof loophole for the C/C++ to POWER and ARMv7 trailing-sync compiler mappings*. *CoRR*, abs/1611.01507, 2016. Available: <http://arxiv.org/abs/1611.01507>.

17. Manson J., Pugh W., Adve S.V. The Java memory model. In *POPL 2005*, ACM, 2005, Pp. 378–391, DOI: 10.1145/1040305.1040336.

18. Milner R. Communication and concurrency. *PHI Series in Computer Science*. Prentice Hall, 1989.



19. **Podkopaev A., Lahav O., Vafeiadis V.** Promising compilation to ARMv8 POP. In *ECOOP 2017*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2017.

20. **Pulte Ch., Flur Sh., Deacon W., French J., Sarkar S., Sewell P.** Simplifying ARM concurrency: Multicopy-atomic axiomatic and operational models for ARMv8. *Accepted to POPL'18*.

21. **Sarkar S., Sewell P., Alglave J., Maranget L., Williams D.** Understanding POWER multiprocessors. In *PLDI 2011*, ACM, 2011, Pp. 175–186. DOI: 10.1145/1993498.1993520

22. **Sevcik J., Vafeiadis V., Nardelli F.Z., Jagannathan S., Sewell P.** Relaxed-memory concurrency and verified compilation. *Proceedings*

*of the 38th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2011*, Austin, TX, USA, 2011, Pp. 43–54. Available: <http://doi.acm.org/10.1145/1926385.1926393>. DOI: 10.1145/1926385.1926393

23. **Sewell P., Sarkar S., Owens Sc., Nardelli F.Z., Myreen M.O.** x86-TSO: A rigorous and usable programmer's model for x86 multiprocessors. *Commun. ACM*, 2010, Vol. 53(7), Pp. 89–97. DOI: 10.1145/1785414.1785443

24. **Vafeiadis V., Balabonski Th., Chakraborty S., Morisset R., Nardelli F.Z.** Common compiler optimisations are invalid in the C11 memory model and what we can do about it. In *POPL 2015*, ACM, 2015, Pp. 209–220.

*Received 27.09.2017.*

#### СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

**ПОДКОПАЕВ Антон Викторович**

**PODKOPAEV Anton V.**

E-mail: a.podkopaev@2009.spbu.ru

**ЛАХАВ Ори**

**LAHAV Ori**

E-mail: orilahav@tau.ac.il

**ВАФЕЯДИС Виктор**

**VAFEIADIS Viktor**

E-mail: viktor@mpi-sws.org

DOI: 10.18721/JCSTCS.10406

УДК 004.81

## ОПИСАНИЕ ПРОЦЕССОВ РАЗРАБОТКИ ПРОГРАММ С ПОМОЩЬЮ СКРЫТЫХ МАРКОВСКИХ МОДЕЛЕЙ

*Д.А. Тимофеев, А.В. Самочадин*

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

Рассмотрены процессы работы программиста над индивидуальными задачами в рамках программного проекта и предложен способ моделирования таких процессов на основе скрытых марковских моделей. Модель процесса может использоваться для решения трех задач: анализа и сравнения процессов, повышения эффективности работы и снижения утомляемости работника, обучения студентов технологическим процессам опытных разработчиков. Две последние задачи предложено решать с помощью введения обратной связи в виде подсказок, формируемых на основе анализа текущего вида деятельности программиста и предыдущих состояний процесса разработки. Идентификация вида деятельности программиста осуществляется путем восстановления наиболее правдоподобной последовательности состояний модели на основе анализа наблюдаемых действий программиста в среде разработки и других приложениях.

**Ключевые слова:** разработка программного обеспечения; моделирование процессов; скрытые марковские модели; обратная связь; производительность труда.

**Ссылка при цитировании:** Тимофеев Д.А., Самочадин А.В. Описание процессов разработки программ с помощью скрытых марковских моделей // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2017. Т. 10. № 4. С. 70–77. DOI: 10.18721/JCSTCS.10406

## MODELING OF SOFTWARE DEVELOPMENT PROCESSES WITH HIDDEN MARKOV MODELS

*D.A. Timofeev, A.V. Samochadin*

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

This paper describes the individual process of programming as a specific part of a more general software development process. We discuss the task of programming process modeling and propose a new approach based on hidden Markov models. The model may be used to solve three kinds of problems: the analysis and comparison of processes, making the process more efficient and less exhausting and transferring the process knowledge from experienced developers to students. The latter two problems are solved by detecting the kind of programmer's activity. We do it by inferring the most likely sequence of process model states based on observed actions the developer performs using the integrated development environment and other relevant applications.

**Keywords:** software development; process modeling; hidden Markov model; feedback; productivity.

**Citation:** Timofeev D.A., Samochadin A.V. Modeling of software development processes with hidden Markov models. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2017, Vol. 10, No. 4, Pp. 70–77. DOI: 10.18721/JCSTCS.10406

## Введение

Процесс разработки программного обеспечения включает в себя планирование работ и выполнение их конкретными исполнителями. Вопрос планирования работ изучен достаточно хорошо. Все распространенные методологии задают критерии разбиения задач на подзадачи, способы оценки их трудоемкости и порядок контроля выполнения работ. В это же время вопрос о том, как именно разработчик работает над конкретной задачей, исследован значительно меньше.

Одной из причин этого является природа самого процесса. Работа программиста рассматривается (и прежде всего самими разработчиками) как творческий акт.

В качестве иллюстрации можно привести заключение тьюринговской лекции Д. Кнута: «Итак, мы видели, что программирование — это искусство, поскольку оно является приложением накопленных знаний для практических целей, поскольку оно требует умения и мастерства, и в особенности потому, что продукты программирования могут представлять эстетическую ценность. Программист, который бессознательно ощущает себя художником, получает удовольствие от своей работы и справляется с ней лучше» [1].

Одновременно процесс программирования предполагает использование определенной технологии<sup>1</sup>. Она определяет правила декомпозиции задачи (структурное, объектно-ориентированное, компонентное и другие виды программирования), использование стандартных приемов организации программы (например, шаблонов проектирования [2]), применение тех или иных инструментов (языков программирования, сред разработки, систем контроля версий). Принятая в рамках проекта технология в существенной степени определяет способ организации программы, подход к ее проектированию и реализации. Тем не менее

даже в рамках одной технологии производительность труда программистов может существенно отличаться друг от друга из-за использования ими различных процессов работы<sup>2</sup>.

Данная статья посвящена одному подходу к моделированию процесса работы программиста на уровне отдельных задач. Целью создания такой модели является решение следующих задач:

- экспериментальное исследование и сравнение процессов, характерных для программистов с различным опытом;
- выработка рекомендаций по организации процесса работы и формированию обратной связи в процессе работы, позволяющих увеличить производительность труда и снизить утомляемость работника;
- обучение начинающих программистов приемам и процессам работы опытных программистов путем формирования подсказок в ходе выполнения заданий.

## Моделирование процесса программирования

Процесс программирования относительно редко рассматривается с точки зрения технологии. В работе [3] исследован вопрос различий между способами решения задачи, характерными для опытных и для начинающих программистов. Для выявления этих различий использовался метод, при котором участник эксперимента рассуждает вслух и комментирует свои действия. Так как предметом исследования в данной работе являлся сам способ рассуждения о задаче, формальное описание процесса работы над задачей не выполнялось. Аналогичный подход использован в работе [4], где видеозапись процесса разработки в сочетании с проговариванием и объяснением всех выполняемых действий применялась для обучения начинающих студентов-программистов.

Построение формальной модели про-

<sup>1</sup> Гагарина Л.Г., Кокорева Е.В., Виснадул Б.Д. Технология разработки программного обеспечения: учебное пособие. М.: ИД «ФОРУМ»; ИНФРА-М, 2008. 400 с.

<sup>2</sup> Kamma D. Study of Task Processes for Improving Programmer Productivity. PhD thesis. Indraprastha Institute of Information Technology, Delhi, 2007 // URL: <https://repository.iiitd.edu.in/jspui/bitstream/handle/123456789/513/PhD1004.pdf>.

цесса разработки было предпринято в диссертации<sup>2</sup>. В ходе исследования шесть разработчиков были разделены на две группы на основании своего опыта. Каждый из них решал типовую задачу разработки модульных тестов, комментируя выполняемые действия, а процесс разработки записывался на видео. Модели процессов строились на основе полученных видеозаписей и комментариев с использованием аппарата марковских цепей. Полученные модели затем использовались для определения степени сходства между процессами различных разработчиков и анализа их эффективности.

Существенно большее количество работ посвящено моделированию других выполняемых людьми процессов. Основным формальным аппаратом при этом являются скрытые марковские модели (СММ) [5] и их модификации. Одно из направлений исследований связано с задачами анализа, индексирования и поиска видеозаписей. Например, в [6] анализируется структура теннисного матча. Трансляция разбивается на базовые фрагменты (ошибка при первой подаче, розыгрыш, повтор момента и перерыв в трансляции), из которых затем составляется иерархическая структура матча (сеты, геймы, перерывы) и трансляции. На основе вручную размеченных записей строится СММ, которая затем служит для классификации фрагментов новых трансляций. Аналогичный подход использовался для анализа структуры трансляции матчей по футболу и бейсболу [7, 8].

Другой важной областью применения формальных моделей поведения человека является здравоохранение. В ряде работ анализ повседневной деятельности людей, особенно пожилых, используется для принятия решений об оказании помощи [9] или для ранней диагностики деменции и других состояний, приводящих к изменению поведения пациента [10]. В работе [9] модель строится на основании размеченных и неразмеченных данных, получаемых с датчиков. Для классификации неразмеченных элементов пространства признаков использовалась их близость к известным точкам этого пространства с учетом локальной структуры многообразия. Ре-

зультат классификации наблюдений используется для построения СММ. Авторы работы [10] применили аппарат двухуровневых иерархических СММ [11], в которых каждому состоянию соответствует своя СММ, а в качестве наблюдаемых величин использовали признаки, извлекаемые из видеоизображения. Если целью работы является не анализ процесса, а только идентификация отдельных состояний, то могут применяться и более простые методы на основе стандартных алгоритмов классификации [12, 13].

В данной статье рассматриваются новые возможности использования формальной модели процесса программирования, основанные на сопоставлении модели с фактическим процессом работы над программой и формировании на этой основе обратной связи. Анализ текущего состояния и истории процесса разработки позволит идентифицировать состояния, в которых процесс целесообразно прервать или изменить, либо, напротив, следует максимально ограничить воздействие отвлекающих факторов для поддержания максимальной эффективности и удовлетворения от работы (состояние «потока» [14]).

Процесс работы программиста над задачей включает несколько видов деятельности, среди которых можно выделить следующие:

- обсуждение задач, требований и ограничений, путей решения;
- получение или предоставление информации о работе компонентов системы, справочной информации;
- изучение документации (знакомство с предметной областью, изучение документации по используемым инструментам, по внешним компонентам, поиск справочных материалов);
- изучение программного кода (чтение и анализ кода внешних компонентов, чтение и анализ существующего кода, инспекция кода других разработчиков);
- планирование и проектирование (планирование работы, архитектурное проектирование, разработка структур данных и алгоритмов, разработка тестов, планирование рефакторинга);

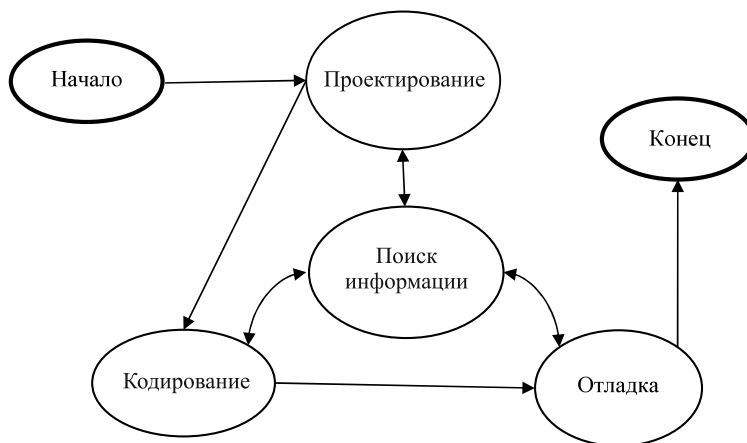
- кодирование (написание кода, тестов, редактирование ранее написанного кода);
- отладка (проверка работоспособности написанного или отредактированного кода, поиск и диагностика ошибок или проблем производительности);
- тестирование (модульное, интеграционное, нагрузочное);
- сборка и развертывание (подготовка установочных образов, развертывание системы на сервере, настройка взаимодействия компонентов распределенной системы);
- документирование (комментирование кода, разработка внутренней и внешней документации, разработка учебных и демонстрационных материалов);
- обеспечение процесса разработки (настройка сред разработки, редакторов, сред выполнения программы, управление версиями кода и данных, обновление и анализ информации в системе управления задачами).

Перечень задач определяется ролью конкретного разработчика в проекте. В крупной компании задачи тестирования, сборки и развертывания, документирования, а также в определенной степени настройки среды разработки могут быть делегированы выделенным сотрудникам, в то время как в малых коллективах к набору функций разработчика могут быть также добавлены такие задачи, как анализ требований или поддержка и обучение пользователей.

Перечисленные виды деятельности с точки зрения управления процессом могут как входить в одну задачу, так и составлять отдельные задачи. Обычно вместе объединяются задачи планирования и проектирования (особенно на уровне структур данных и алгоритмов), кодирования и отладки, к которым по мере необходимости добавляется поиск справочной информации. Такой процесс в идеализированном виде представлен на рисунке.

На практике такой почти линейный процесс реализуется не всегда. Одной из причин нарушения линейности является обнаружение разработчиком ошибок, допущенных на одном из предшествующих этапов. В этом случае ошибка исправляется, затрагивая все последующие этапы. Почти всегда происходит возврат из состояния «отладка» к состоянию «кодирование», но возможен и возврат к этапу «проектирование». Вторая причина, по которой процесс может выполняться нелинейно — декомпозиция задачи (в частности, программирование «сверху вниз», при котором вместо отсутствующих функций используются «заглушки», и «снизу вверх», когда сначала разрабатываются и отлаживаются отдельные базовые функции, объединяемые затем в более сложные алгоритмы). В этом случае процесс становится циклическим, а в каждом состоянии работа ведется только над частью задачи.

Исходя из этих наблюдений, можно,



Базовый процесс работы над задачей  
The basic process of work on the task



следуя описанному в диссертации<sup>2</sup> подходу, моделировать процесс работы над задачей с помощью марковских цепей. Марковская цепь задается множеством состояний  $S = \{s_1, s_2, \dots, s_n\}$  и матрицей переходов  $A$ , каждый элемент которой  $a_{ij}$  соответствует вероятности перехода цепи из состояния  $s_i$  в состояние  $s_j$ . Вероятность перехода в каждое из состояний зависит только от текущего состояния вне зависимости от того, какими были предшествующие состояния.

В реальном процессе разработки переходы между состояниями не случайны, но они зависят от большого количества факторов, в том числе недоступных или только частично доступных для наблюдения (от опыта и знаний разработчика, его психофизиологического состояния, плана работы, особенностей решаемой задачи, присутствия других людей и т. д.), что позволяет считать использование статистического моделирования допустимым упрощением.

Чтобы построить модель конкретного процесса, необходимо зафиксировать множество состояний  $S$  и найти матрицу  $A$ . Для поиска матрицы переходов необходимо идентифицировать последовательность состояний процесса. При наличии явного комментария, включающего переходы между состояниями, для построения элемента  $a_{ij}$  матрицы  $A$  достаточно вычислить количество переходов из состояния  $s_i$  в состояние  $s_j$  к общему количеству переходов из состояния  $s_i$ :

$$a_{ij} = \frac{|s_i \rightarrow s_j|}{|s_i \rightarrow s_k, 1 \leq k \leq n|}. \quad (1)$$

Моделирование процесса программирования с помощью марковских цепей в принципе позволяет решить первую из трех сформулированных во введении задач: анализ и сравнение процессов, характерных для программистов с различным опытом работы. Тем не менее для решения двух других задач этой модели недостаточно. Формирование обратной связи в процессе работы требует идентификации текущего состояния. Эта задача тривиальна при наличии эталонной разметки — комментария, описывающего все изменения состояния.

Однако в повседневной деятельности подобные комментарии оказываются недоступными по двум причинам. Во-первых, комментирование всех действий неудобно для пользователя. Во-вторых, автоматический анализ комментариев в свободной форме на естественном языке представляет собой трудную задачу, которая в настоящее время не может быть решена с достаточной точностью без существенного ограничения словаря. Таким образом, идентификация текущего состояния процесса должна выполняться неявно на основе косвенных признаков.

### Моделирование процесса с помощью скрытых марковских моделей

Скрытые марковские модели [5, 15] представляют собой модификацию марковских цепей. Скрытая марковская модель также имеет  $n$  состояний  $S = \{s_1, s_2, \dots, s_n\}$ , переходы между которыми определяются матрицей вероятностей перехода  $A$ , но теперь наблюдатель не может непосредственно определить, в каком состоянии находится модель. Вместо этого модель определяет  $m$  наблюдаемых величин  $V = \{v_1, v_2, \dots, v_m\}$ . На каждом шаге  $t$  модель находится в состоянии  $q_t \in S$  и формирует случайное значение наблюдаемой величины  $o_t \in V$ . Распределение наблюдаемой величины в состоянии  $s_i$  задается вектором  $b_i$ : вероятность в состоянии  $s_i$  наблюдать величину  $v_k$  равна  $b_i(k)$ . Состояние  $s_i$  выбирается в качестве начального состояния  $q_0$  с вероятностью  $\pi(i)$ . Так как процесс работы над задачей конечен, среди состояний модели выбирается одно состояние  $q_F$ , являющееся заключительным.

При моделировании процесса работы программиста скрытые состояния  $S$  соответствуют состояниям процесса разработки (видам деятельности программиста). Наблюдаемые величины соответствуют действиям, которые разработчик выполняет в каждом из состояний.

В разрабатываемой в настоящее время версии системы сбора данных и анализа процессов разработки используются следующие виды наблюдаемых величин:

- низкоуровневые события, связанные с взаимодействием пользователя с операционной системой: нажатия клавиш, перемещение и нажатие кнопок мыши, переключение между окнами приложений, запуск и завершение процессов;

- высокоуровневые события, связанные с взаимодействием пользователя с прикладными программами: действия в среде разработки (начало и завершение сеанса работы, открытие и закрытие проектов и файлов, запуск и завершение отладки) и браузере (открытие и закрытие вкладок, переходы по ссылкам).

Дополнительно предусмотрено получение физиологических характеристик состояния разработчика. В настоящее время поддерживается измерение частоты сердечных сокращений с помощью фитнес-браслета MioLink. Тем не менее эти данные пока не используются для построения моделей.

Анализ событий, возникающих в ходе программирования, производится на компьютере с установленными приложениями для перехвата событий, включая модули расширения для среды разработки Microsoft Visual Studio 2017 и браузера Google Chrome. Данные о каждом событии сохраняются в центральной базе данных с указанием типа события, его параметров и отметки времени. Использование центральной базы данных позволяет формировать и анализировать набор данных о действиях многих пользователей. Такой способ работы с данными ориентирован в первую очередь на использование в лабораторном окружении. Одновременно с этим в системе предусмотрена возможность ограничения передачи данных за пределы локального компьютера, что позволяет пользователям проводить локальный анализ данных о своей повседневной деятельности без риска утечки приватной информации. Управление сбором и передачей данных о каждом из видов событий осуществляет сам пользователь.

Построение скрытой марковской модели может быть выполнено двумя методами. Первый способ заключается в итеративном восстановлении матрицы вероятностей переходов  $A$ , набора распределений  $b_i$  (который можно задать одной матрицей  $B$ ) и

распределения  $\pi$  путем последовательных приближений на основе анализа последовательностей наблюдаемых величин. Для скрытых марковских моделей эта задача может быть решена с помощью алгоритма Баума-Велша [15]. Достоинством данного подхода является отсутствие необходимости знать истинную последовательность скрытых состояний процесса. Вместо этого алгоритм восстанавливает параметры модели в соответствии с принципом максимального правдоподобия. Использование данного метода целесообразно для выявления закономерностей в действиях многих разработчиков, когда идентификация «истинных» состояний (например, в ходе явного проговаривания переходов между видами деятельности) невозможна или слишком дорога. В этом случае достаточно организовать сбор событий о наблюдаемых действиях пользователей, не требуя дополнительных аннотаций или участия экспериментатора.

Тем не менее данный подход требует большого количества наблюдений, а для быстрой сходимости нужно хорошее начальное приближение для матриц  $A$  и  $B$ . Предварительные эксперименты также показали, что для использования описанного подхода целесообразно проводить предварительную обработку данных о событиях. В частности, вместо использования элементарных событий типа «нажатие клавиши» следует анализировать последовательности нажатых клавиш и формировать события более высокого уровня, позволяющие идентифицировать вид вводимого текста. Такой анализ выполняется на основе языковых моделей для естественных языков и языков программирования. Использование событий более высокого уровня позволяет существенно снизить размерность задачи.

Второй подход возможен при наличии явного знания о последовательности состояний. Хотя в общем случае последовательность состояний скрытого марковского процесса недоступна для наблюдения, при построении модели процесса разработки эти данные могут быть получены для некоторого ограниченного количества экспериментов, в которых действия разработчика протоколируются. В таком случае исполь-

зуется следующий алгоритм оценки параметров модели.

1. С помощью формулы (1) на основе собранных данных вычисляются элементы матрицы  $A$ .

2. Для каждого состояния  $s_j$ , для каждого наблюдаемого действия  $v_i$  вычисляется

$$b_i(j) = \frac{|o = v_j, s = s_i|}{|s = s_i|} \quad (\text{отношение количе-}$$

ства наблюдений действия  $v_j$ , выполняемого в состоянии  $s_j$ , к общему числу действий в этом состоянии).

Несмотря на то что при использовании этого подхода модель строится для относительно небольшого числа экспериментов, ее можно использовать либо непосредственно в качестве эталонной модели (для простых моделей), либо в качестве начального приближения для алгоритма Баума-Велша. При этом также целесообразно снижать размерность пространства наблюдаемых величин путем построения более высокоуровневых признаков вида «ввод текста программы на языке Java» или «ввод текста на русском языке».

### Заключение

В данной статье рассмотрен подход к построению формальной модели процессов, выполняемых разработчиками программного обеспечения в ходе работы над отдельными задачами. Такая модель может

применяться для анализа и повышения эффективности используемых процессов, более точной оценки производительности труда разработчика при выполнении определенного класса задач, а также для обучения студентов-программистов и начинающих разработчиков эффективным процессам работы над задачами.

Если в качестве наблюдаемых величин использовать также показатели психофизиологического состояния (частота сердечных сокращений, пульсовая волна, электрическая активность кожи и др.), измеряемые с помощью фитнес-браслетов или других доступных приборов, аналогичные модели можно также применять для оценки и прогнозирования психофизиологического состояния разработчика в течение рабочего дня или на больших интервалах времени.

В настоящее время реализована первая версия системы, осуществляющей сбор и хранение данных о выполняемых разработчиками действиях и о значениях физиологических показателей. С ее помощью ведется формирование корпуса данных о действиях разработчиков программного обеспечения.

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

### СПИСОК ЛИТЕРАТУРЫ

1. Эшенхерст Р. (ред). Лекции лауреатов премии Тьюринга. Пер. с англ. М.: Мир, 1993. 560 с.
2. Гамма Э., Хелм Р., Джонсон Р., Влассидес Д. Приемы объектно-ориентированного проектирования. Паттерны проектирования. Пер. с англ. СПб.: Питер, 2014. 366 с.
3. Jeffries R., et al. The processes involved in designing software // Cognitive Skills and their Acquisition. 1981. Pp. 255–283.
4. Bennedsen J., Caspersen M.E. Revealing the programming process // Proc. of SIGCSE'05. St. Louis, Missouri, 2005.
5. Baum L.E., Petrie T. Statistical inference for probabilistic functions of finite state Markov chains // The Annals of Mathematical Statistics. 1966. Vol. 37 (6). Pp. 1554–1563.
6. Kijak E., et al. Audiovisual integration for tennis broadcast structuring // Multimedia Tools and Applications. 2006. Vol. 30. No. 3. Pp. 289–311.
7. Xie L., et al. Structure analysis of soccer video with Hidden Markov Models // Proc. of the IEEE Internat. Conf. on Acoustics, Speech, and Signal Processing. IEEE, 2002. Vol. 4. Pp. IV-4096–IV-4099.
8. Chang P., Han M., Gong Y. Extract highlights from baseball game video with Hidden Markov Models // Proc. of the 2002 Internat. Conf. on. IEEE, 2002. Vol. 1. Pp. I-I.
9. Ghazvininejad M., et al. HMM based semi-supervised learning for activity recognition // Proc. of the 2011 Internat. Workshop on Situation Activity & Goal Awareness. ACM, 2011. Pp. 95–100.
10. Karaman S., et al. Hierarchical Hidden Markov Model in detecting activities of daily



living in wearable videos for studies of dementia // *Multimedia Tools and Applications*. 2014. Vol. 69. No. 3. Pp. 743–771.

11. **Fine S., Singer Y., Tishby N.** The hierarchical Hidden Markov Model: Analysis and applications // *Machine Learning*. 1998. Vol. 32. No. 1. Pp. 41–62.

12. **Stikic M., Schiele B.** Activity recognition from sparsely labeled data using multi-instance learning // *Location and Context Awareness*. 2009. Pp. 156–173.

13. **Zhou W., Vellaikal A., Kuo C.C.** Rule-based video classification system for basketball video indexing // *Proc. of the 2000 ACM Workshops on Multimedia*. ACM, 2000. Pp. 213–216.

14. **Чиксентмихайи М.** Поток: Психология оптимального переживания. Пер. с англ. 7-е изд. М.: Смысл; Альпина нон-фикшн, 2017.

15. **Rabiner L.R.** A tutorial on Hidden Markov Models and selected applications in speech recognition // *Proc. of the IEEE*. 1989. Vol. 77. Issue 2. Pp. 257–286.

Статья поступила в редакцию 11.11.2017.

## REFERENCES

1. **Eshenherst R. (ed).** *ACM turing award lectures*. Moscow: Mir Publ., 1993, 560 p. (rus)

2. **Gamma E., Helm R., Johnson R., Vlissides D.** *Design patterns: elements of reusable object-oriented software*. St. Petersburg: Piter Publ., 2014, 366 p. (rus)

3. **Jeffries R., et al.** The processes involved in designing software. *Cognitive Skills and their Acquisition*, 1981, Pp. 255–283.

4. **Bennedsen J., Caspersen M.E.** Revealing the programming process. *Proceedings of SIGCSE'05*, St. Louis, Missouri, Febr. 23–27, 2005.

5. **Baum L.E., Petrie T.** Statistical inference for probabilistic functions of finite state Markov chains. *The Annals of Mathematical Statistics*, 1966, Vol. 37 (6), Pp. 1554–1563.

6. **Kijak E., et al.** Audiovisual integration for tennis broadcast structuring. *Multimedia Tools and Applications*, 2006, Vol. 30, No. 3, Pp. 289–311.

7. **Xie L., et al.** Structure analysis of soccer video with Hidden Markov Models. *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, IEEE, 2002, Vol. 4, Pp. IV-4096–IV-4099.

8. **Chang P., Han M., Gong Y.** Extract highlights from baseball game video with Hidden Markov Models. *Proceedings of the 2002 International Conference on*, IEEE, 2002, Vol. 1, Pp. I-I.

9. **Ghazvininejad M., et al.** HMM based

semi-supervised learning for activity recognition. *Proceedings of the 2011 International Workshop on Situation Activity & Goal Awareness*, ACM, 2011, Pp. 95–100.

10. **Karaman S., et al.** Hierarchical Hidden Markov Model in detecting activities of daily living in wearable videos for studies of dementia. *Multimedia Tools and Applications*, 2014, Vol. 69, No. 3, Pp. 743–771.

11. **Fine S., Singer Y., Tishby N.** The hierarchical Hidden Markov Model: Analysis and applications. *Machine Learning*, 1998, Vol. 32, No. 1, Pp. 41–62.

12. **Stikic M., Schiele B.** Activity recognition from sparsely labeled data using multi-instance learning. *Location and Context Awareness*, 2009, Pp. 156–173.

13. **Zhou W., Vellaikal A., Kuo C.C.** Rule-based video classification system for basketball video indexing. *Proceedings of the 2000 ACM Workshops on Multimedia*, ACM, 2000, Pp. 213–216.

14. **Csikszentmihalyi M.** *Flow: The psychology of optimal experience*. Moscow: Smysl; Alpina non-fiction publ., 2017. (rus)

15. **Rabiner L.R.** A tutorial on Hidden Markov Models and selected applications in speech recognition. *Proceedings of the IEEE*, 1989, Vol. 77, Issue 2, Pp. 257–286.

Received 11.11.2017.

## СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

**ТИМОФЕЕВ Дмитрий Андреевич**

**TIMOFEEV Dmitrii A.**

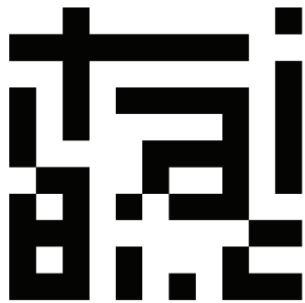
E-mail: dtim@dcn.icc.spbstu.ru

**САМОЧАДИН Александр Викторович**

**SAMOCHADIN Alexander V.**

E-mail: samochadin@gmail.com





# ICAIT 2018 AIZU

## 3<sup>rd</sup> International Conference on Applications in Information Technology

November 1 - 3, 2018, Aizu-Wakamatsu

<http://ica-it-aizu.org>

The 3rd International Conference on Applications in Information Technology in Aizu-Wakamatsu (Japan) is the meeting point for researchers, scholars and IT-industry professionals interested in advancing computer science and technology towards serving society and expectations of the day in the digitally transformed world. Our objective is to foster rich creativity in students' research works and to encourage young scientists to participate actively in open discussions with their colleagues.



**CALL FOR PAPERS** The 3rd International Conference on Applications in Information Technology invites original contributions presenting new advances and research results in a rich domain of information technology applications.

In 2018 we particularly welcome high quality contributions in the areas which include but are not limited to:

- Computer Systems and Architectures
- Software Design and Engineering
- Web Engineering
- Control Systems
- Intelligent and Embedded Systems
- Cloud Computing and Networking
- Human-Centric Computing
- Information Retrieval and Data Science
- Information Management
- Mobile and Cross Platform Applications
- Knowledge Engineering
- Machine Learning
- Information Technology for Learning and Education
- Machine Vision and Pattern Recognition
- Modeling and Simulation
- Game Development
- 3D Modeling, Virtualization and Visualization
- Internet of Things
- Programming Robots
- Information Technology in Economy

**IMPORTANT DATES** May 31, 2018

September 2, 2018

September 23, 2018

November 1 – 3, 2018

**Paper submission**

**Acceptance notification**

**Camera-ready paper submission**

**Conference**

Accepted papers will be published in the Conference ISBN labeled proceedings. Selected papers, after further revisions, will be recommended for publication in the conference partner journals.

**VENUE**

**The University of Aizu**

Tsuruga, Ikki-machi,  
Aizu-Wakamatsu, 9658580,  
Japan





DOI: 10.18721/JCSTCS.10407

УДК 004.021 004.046

## РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТАЛЬНЫХ ИССЛЕДОВАНИЙ ТОЧНОСТИ ПОЗИЦИОНИРОВАНИЯ ПРИ ИСПОЛЬЗОВАНИИ РАЗЛИЧНЫХ СИСТЕМ СПУТНИКОВОЙ НАВИГАЦИИ

*Л.М. Курочкин, М.А. Курочкин, С.Г. Попов, М.В. Попов*

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

Система навигации автономного мобильного объекта и точность навигационных данных играют ключевую роль при решении задач автономного управления мобильными объектами: планирование маршрута; управление параметрами движения; организация группового взаимодействия. Решение указанных задач требует высокой точности навигационных данных. Точность навигационных данных, получаемых от различных спутниковых систем, постоянно изменяется. Использование дифференциальных поправок для повышения точности навигационных данных в произвольной географической точке возможно не всегда из-за удалённости опорных станций и отсутствия надёжных каналов передачи дифференциальных поправок. Оперативная обработка/фильтрация навигационных данных на стороне автономного устройства часто невозможна ввиду ограничений доступных вычислительных мощностей. Оперативный выбор системы спутникового позиционирования или их комбинации, обеспечивающей наиболее точное позиционирование, является актуальной задачей. В статье описана методика, позволяющая осуществить оперативный выбор наиболее точной системы спутникового позиционирования или их комбинаций. Выбор осуществляется на основе статистического анализа данных, получаемых от систем спутникового позиционирования.

**Ключевые слова:** навигация автономных мобильных объектов; статистическая оценка навигационных данных; выбор системы спутникового позиционирования; точность навигационных данных; автономное управление.

**Ссылка при цитировании:** Курочкин Л.М., Курочкин М.А., Попов С.Г., Попов М.В. Результаты экспериментальных исследований точности позиционирования при использовании различных систем спутниковой навигации // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2017. Т. 10. № 4. С. 79–88. DOI: 10.18721/JCSTCS.10407

## RESULTS OF EXPERIMENTAL STUDIES OF POSITIONING ACCURACY USING VARIOUS SATELLITE NAVIGATION SYSTEMS

*L.M. Kurochkin, M.A. Kurochkin, S.G. Popov, M.V. Popov*

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

There is a constantly growing interest in tasks involving groups of autonomous mobile objects: environmental monitoring, elimination of consequences of technological disasters. A navigation system of an autonomous mobile object and the accuracy of navigation data play a key role in tasks of autonomous control of mobile objects. These tasks include: route planning, control over driving parameters of an autonomous mobile object, arrangement of interaction within a group of autonomous mobile objects. The accuracy of navigation data from satellite navigation systems essentially depends on such factors as: the number of visible satellites for any available navigation system, receiver characteristics, terrain features, environment properties. It is an important task to make a rapid choice from the available satellite navigation systems or their combination that provides the most accurate position when differential corrections are not available. This paper covers the technique that allows quick choosing of the most precise satellite navigation system or their combination. The choice is based on a statistical analysis of data received from satellite navigation systems.

**Keywords:** autonomous mobile objects navigation; navigational data statistical estimation; choice of satellite positioning system; navigation data precision; autonomous control.

**Citation:** Kurochkin L.M., Kurochkin M.A., Popov S.G., Popov M.V. Results of experimental studies of positioning accuracy using various satellite navigation systems. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2017, Vol. 10, No. 4, Pp. 79–88. DOI: 10.18721/JCSTCS.10407

## Введение

В настоящее время эксплуатируются несколько систем спутникового позиционирования: GPS, GLONASS, GALILEO, BeiDou. В силу влияния различных факторов точное позиционирование не всегда возможно. Точность позиционирования зависит от расположения спутников на орбите, погодных условий, особенностей рельефа, ионосферной и тропосферной рефракции [9, 10]. Результаты анализа указанных параметров предоставляются глобальной службой навигационных спутниковых систем [5].

Для снижения влияния случайной погрешности на точность определения координат используются следующие методы фильтрации: фильтр Кальмана; расширенный фильтр Кальмана, что позволяет сократить погрешность определения координат на порядок [1, 8]. Использование заранее сформированных полей точности конкретной системы навигации также позволяет значительно сократить погрешность определения координат [7].

Разработке и исследованию методов высокоточного навигационного обеспечения, учитывающего погрешности, вызванные ошибками эфемеридно-временной информации, влияния релятивистских, гравитационных и приливных эффектов,

ошибки многолучевости посвящена работа [2]. Приведённый метод позволяет достичь точности, сравнимой с точностью позиционирования, полученной с учётом RTK поправок.

Анализ навигационных данных, полученных от комбинации навигационных систем, позволяет в частных случаях получить погрешность не более 10 см, сократить время конвергенции до 70 % [3]. Анализ доступности различных спутниковых навигационных систем и точности предоставляемых ими данных в зависимости от периода наблюдения также вызывает интерес у исследователей [4].

Основной целью данного исследования является разработка методики, позволяющей осуществить оперативный выбор наиболее точной системы спутникового позиционирования или их комбинаций на основе статистического анализа данных, получаемых от этих систем.

## Критерий выбора наиболее точной системы спутникового позиционирования

При формировании управляющего воздействия, корректирующего траекторию движения мобильного объекта, необходимо использовать наиболее точные координаты объекта. Задача оптимального управления формулируется следующим образом: в фазо-

вом пространстве объекта  $X$  даны две точки  $x_n$  и  $x_k$ . Среди всех допустимых управлений  $u(t)$ , переводящих объект из точки  $x_n$  в точку  $x_k$ , найти такое, для которого критерий оптимальности  $Q(x, u) = \int_{t_0}^{t_1} G(x(t), u(t)) dt$  принимает наименьшее значение;  $x(t)$  – решение системы дифференциальных уравнений, описывающих движение управляемого объекта уравнения (с начальными условиями  $x(t_0) = x_n$ ) [7].

Для определения координат объекта (при отсутствии возможностей получения дифференциальных поправок и фильтрации полученных данных) необходимо выбрать наиболее точную систему спутникового позиционирования или их ком-

бинацию. Мобильный объект, оснащённый бюджетным одноканальным приёмником навигационных сигналов, способен самостоятельно выбрать наиболее точную систему спутникового позиционирования или комбинации таких систем из набора доступных в данной точке Земли, проанализировав данные, получаемые от систем спутникового позиционирования или их комбинаций.

Критериями выбора системы спутникового позиционирования или их комбинации являются:

- минимум среднего расстояния до «эталонного» центра (вычисленного по данным конкретной системы спутникового позиционирования или их комбинации):

$$\min_{s1..k} \left( \sum_1^{ns} \sqrt{((x_{\text{центр}-s} - x_{ns})^2 + (y_{\text{центр}-s} - y_{ns})^2) / n_s} \right),$$

где  $s$  – система спутникового позиционирования или их комбинация;  $n_s$  – количество точек в выборке;  $x_{ns}$ ,  $y_{ns}$  – координаты точки из выборки, полученной для данной системы позиционирования или их

комбинации;  $x_{\text{центр}-s}$ ,  $y_{\text{центр}-s}$  – координаты «эталонного» центра (для данного набора точек).

- минимум дисперсии расстояний до «эталонного» центра:

$$\min_{s1..k} \left( \sum_1^n (d_{\text{среднее}-s} - d_i)^2 / (n_s - 1) \right), 1 \leq i \leq n_s,$$

где  $d_i$  – расстояние от точки, принадлежащей выборке, до вычисленного «эталонного» центра;  $d_{\text{среднее}-s}$  – среднее расстояние до вычисленного «эталонного» центра, полученное для данной системы позиционирования или их комбинации;  $n_s$  – количество точек в выборке.

### Выбор системы спутникового позиционирования. Эксперимент

Выбор системы спутникового позиционирования проводится по следующему алгоритму:

1. Объект прекращает движение.
2. Выбирается система спутникового позиционирования или комбинация систем.

$$d_{\text{среднее}} = \sum_1^n \sqrt{((x_{\text{центр}} - x_n)^2 + (y_{\text{центр}} - y_n)^2) / n},$$

где  $n$  – количество точек в выборке;  $x_n$ ,  $y_n$  – координаты точки из выборки, и 2) диспер-

сия расстояний до «эталонного» центра для полученного в п. 3 набора точек:

3. По полученным данным определяется собственное местоположение: в течение времени  $T$  формируется набор  $n$  точек – собственных координат:  $x_i, y_i, 1 \leq i \leq n$ .

4. Для полученного набора точек вычисляется положение «эталонного» центра – точки с координатами, соответствующими среднему значению координат в выборке:

$$x_{\text{центр}} = \sum_1^n x_i / n; y_{\text{центр}} = \sum_1^n y_i / n,$$

где  $n$  – количество точек в выборке.

Вычисляются: 1) среднее евклидово расстояние от всех точек набора до вычисленного «эталонного» центра:

$$D = \sum_{i=1}^n (d_{\text{среднее}-s} - d_i)^2 / (n-1), 1 \leq i \leq n,$$

где  $d_i$  – расстояние от точки, принадлежащей выборке, до вычисленного «эталонного» центра.

5. Повторение п. 2 до тех пор, пока не будут перебраны все доступные системы и их комбинации

6. Рекомендуемой для использования системой спутникового позиционирования или комбинацией таких систем принимается, показавшая в п. 4 наименьшие среднее расстояние до центра и дисперсию.

Ключевым моментом в предложенном алгоритме становится выбор длительности интервала времени  $T$ , достаточного для оценки точности данных, получаемых от спутниковой системы или их комбинации. Выбор малого значения  $T$  может привести к некорректному выбору системы спутникового позиционирования или их комбинации; выбор излишне большого значения  $T$  может привести к устареванию полученных в п. 4. оценок, вызванного изменением положений группировок спутников, изменению состояния окружающей среды.

Для оценки длительности временного интервала, достаточного для выбора наилучшей системы спутникового позиционирования или их комбинации была проведена серия экспериментов. В каждом эксперименте использовался приёмник EVK-7 Multi-GNSS evaluation kit. Приёмник устанавливался стационарно, после чего осуществлялся выбор системы спутникового позиционирования или их комбинации, затем осуществлялось протоколирование полученных координат, в течение 15 мин. Частота получения данных от приёмника – 1 с. В экспериментах участвовали системы спутникового позиционирования GPS, Glonass, BeiDou, Galileo, QZSS, SBAS и их парные комбинации. Эксперименты проводились в Санкт-Петербурге, Мадриде (Испания), п. Рудный (Казахстан), г. Конаклы (Турция). Целесообразность территориального «разброса» мест проведения экспериментов вызвана неоднородностью покрытия Земли сигналом спутниковых навигационных

систем. Выбор интервала, превышающего 15 мин, нецелесообразен ввиду того, что оценка точности данных только шести систем спутникового позиционирования, без учёта их комбинаций, займёт 1,5 ч, что не приемлемо для задач оперативного управления, например, для управления группировкой автономных роботов.

Полученные данные обрабатывались согласно алгоритму, приведённому выше: вычислялся центр, дисперсия. Затем проводилась оценка отклонений среднего расстояния, полученных до центра, сформированного по полной пятнадцатиминутной выборке, используя наборы точек от первого измерения до  $k$ -го, где  $k$  менялось, начиная с 30, с шагом 30. То есть проводилось сравнение «эталонного» центра, полученного по пятнадцатиминутной выборке, и значений центров, полученных по выборкам длительностью 30, 60, 90, ..., 300 с; также проводилось сравнение дисперсий.

#### Анализ экспериментальных данных

Значение среднего расстояния для наборов измерений, полученных при использовании различных систем спутниковой навигации и их комбинаций, может значительно отличаться в рамках проведения одного эксперимента (серия пятнадцатиминутных измерений). Например, для измерений, проведённых в Мадриде, значение среднего расстояния до «эталонного» центра отличаются более чем в 7,5 раз (рис. 1).

Дисперсия соответствующих измерений (Galileo и GPS) отличается более чем в 97 раз (рис. 2).

Рассмотрим зависимость отклонения значений дисперсии измерений для различных выборок, размер которых изменяется с шагом 30 с (30, 60, 90, ..., 900 с). На рис. 3 представлены зависимости для систем спутникового позиционирования, показавшие наименьшие значения среднего расстояния до «эталонного» центра: GPS, GPS+QZZ, GPS+BeiDou.

Представленная диаграмма показывает, что для выборок, соответствующих временным интервалам 150 и более секунд, значе-

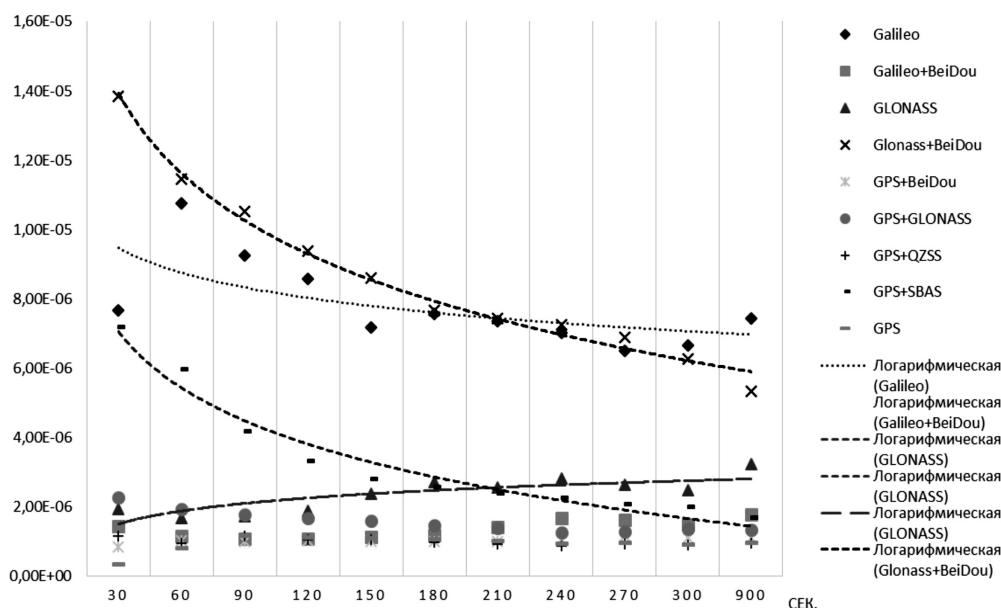


Рис. 1. Среднее расстояние до «эталонного» центра (Мадрид)

Fig. 1. The average distance to the «reference» center (Madrid)

ние вычисленной дисперсии отличается от «эталонного» не более, чем на 0,01 %.

Диаграмма, представленная на рис. 4, показывает, что для выборки, соответствующей временному интервалу 150 с и более,

значение среднего расстояния до «эталонного» центра, полученного для конкретной выборки, отличается не более, чем на 0,001 % от значения, полученного для полной выборки.

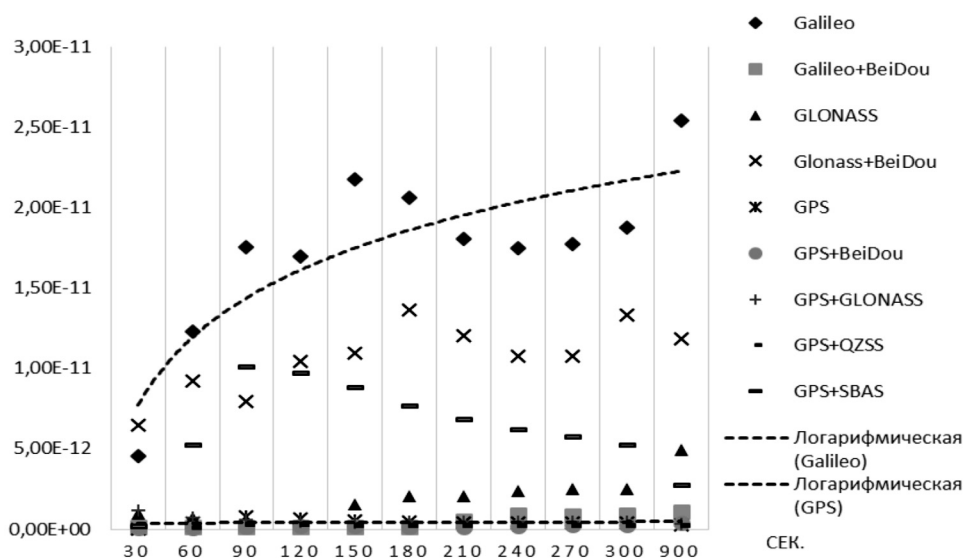


Рис. 2. Зависимость дисперсии среднего расстояния до «эталонного» центра (Мадрид) от размера выборки

Fig. 2. Dependence of the variance of the average distance to the «reference» center (Madrid) on the sample size



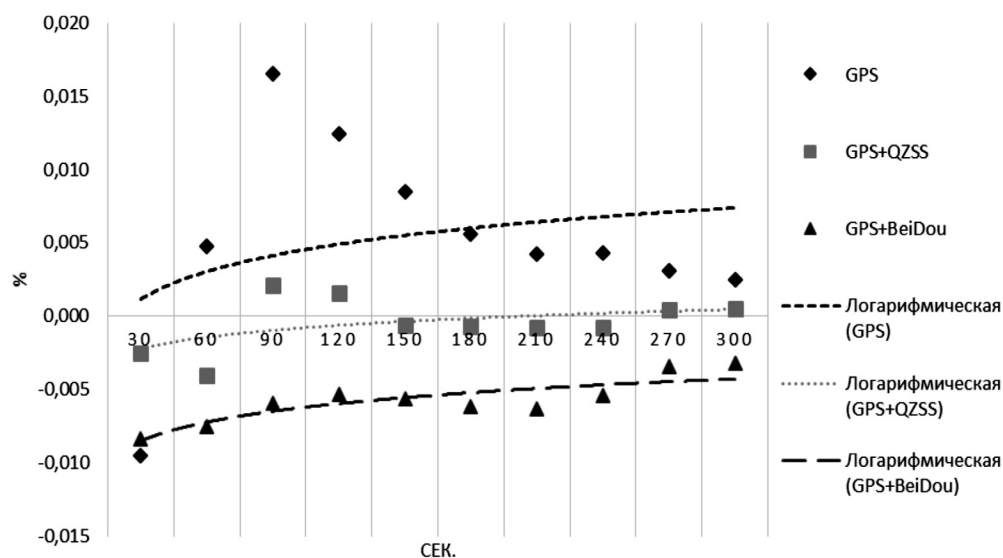


Рис. 3. Зависимость отклонения значений дисперсии (%) от результирующего значения по полной выборке

Fig. 3. Dependence of the deviation of the variance (%) from the resulting value by the full sample

Вычисленные значения дисперсий для указанных выборок принимают значения, представленные на рис. 5.

Диаграмма показывает, что две комбинации спутниковых систем (GPS+BeiDou и

GPS+QZSS) дают наименьшую дисперсию на рассматриваемых выборках. Зависимости отношения дисперсии к среднему значению (математическое ожидание) расстояния до «эталонного» центра составляют величину

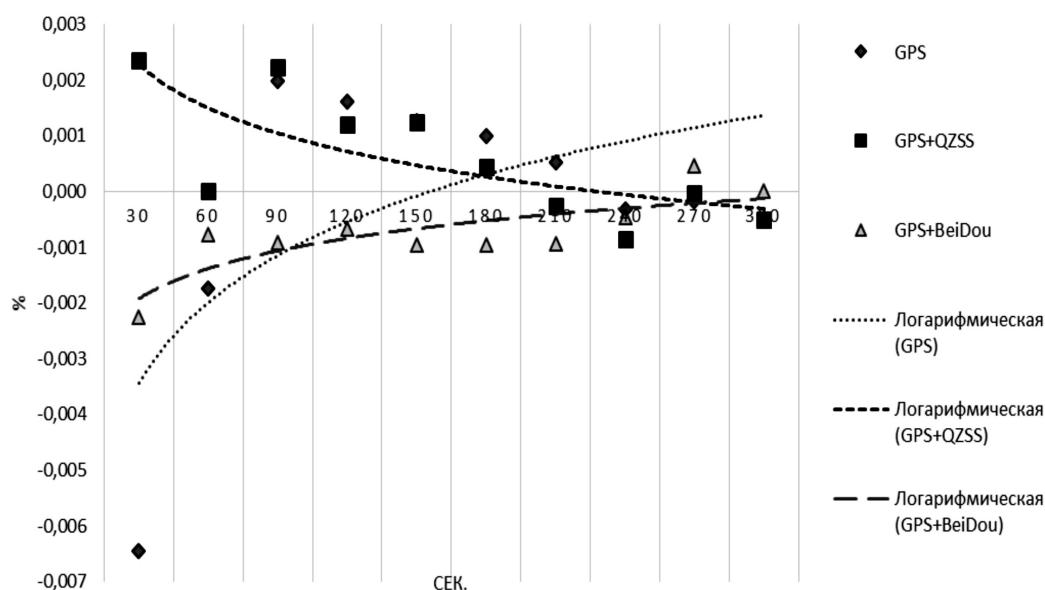


Рис. 4. Зависимость значения среднего расстояния до «эталонного» центра от размера выборки

Fig. 4. Dependence of the mean distance to the «reference» center on the sample size

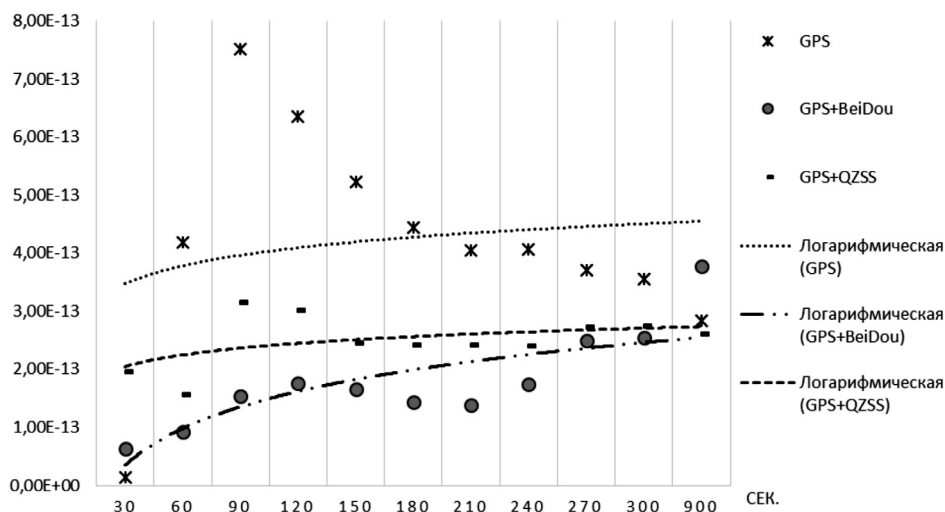


Рис. 5. Зависимость значения среднего расстояния до «эталонного» центра от размера выборки. GPS, GPS+BeiDou, GPS+QZSS

Fig. 5. Dependence of the mean distance to the «reference» center on the sample size. GPS, GPS+BeiDou, GPS+QZSS

порядка  $5 \cdot 10^{-7}$ , что свидетельствует о том, что процессы сходятся. Разность с «эталонным» значением составляет величину порядка  $10^{-7}$ .

По результатам экспериментов, проведённых в п. Рудный (Казахстан) установлено, что объём выборки, необходимой для

адекватной оценки точности систем спутникового позиционирования и их парных комбинаций, составляет 180 с. Наилучшая комбинация – GPS+BeiDou (рис. 6).

Результаты оценки данных (п. Рудный, Казахстан) по критерию «трех сигм» приведены в таблице. Анализ значений обоснован

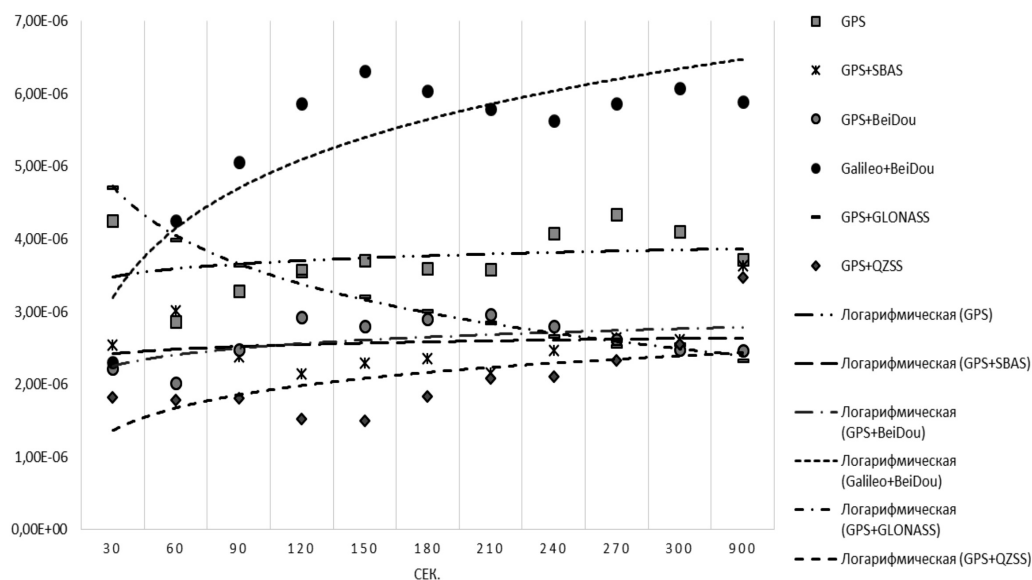


Рис. 6. Зависимость среднего расстояния до эталонного центра от объёма выборки (п. Рудный)

Fig. 6. Dependence of the average distance to the reference center on the sample size (Rudny)

Результаты оценки данных (п. Рудный, Казахстан) по критерию «трех сигм»  
Results of data evaluation (Rudniy, Kazakhstan) according to the criterion of «three sigma»

| Системы позиционирования | Математическое ожидание | Стандартное отклонение | 1sigma  | 2sigma  | 3sigma  | Количество точек | Доверительный интервал (0,95) |
|--------------------------|-------------------------|------------------------|---------|---------|---------|------------------|-------------------------------|
| GLONASS+BeiDou           | 5889331,75              | 0,24195                | 0,65145 | 0,97649 | 0,98893 | 723              | 0,03527                       |
| GPS+BeiDou               | 5889330,10              | 0,27612                | 0,66867 | 0,96687 | 1,00000 | 664              | 0,04200                       |
| GPS+GLONASS              | 5889330,33              | 0,28061                | 0,77842 | 0,92086 | 0,97986 | 695              | 0,04172                       |
| GPS+SBAS                 | 5889330,14              | 0,37960                | 0,67891 | 0,95224 | 1,00000 | 1361             | 0,04034                       |
| GPS+QZSS                 | 5889331,17              | 0,43572                | 0,60634 | 0,97887 | 1,00000 | 1420             | 0,04533                       |
| Galileo+BeiDou           | 5889329,97              | 0,43600                | 0,56364 | 1,00000 | 1,00000 | 385              | 0,08710                       |
| GPS                      | 5889330,16              | 0,52882                | 0,66667 | 0,95798 | 1,00000 | 714              | 0,07758                       |
| BeiDou                   | 5889329,84              | 0,65315                | 0,72488 | 0,93900 | 1,00000 | 836              | 0,08855                       |
| GLONASS                  | 5889332,81              | 0,68317                | 0,87734 | 0,96104 | 0,97547 | 693              | 0,10173                       |
| Galileo                  | 5889327,00              | 1,95891                | 0,67878 | 0,94138 | 0,99414 | 853              | 0,26292                       |

вызывает предположение о том, что данные, полученные от систем спутникового позиционирования или их комбинаций, распределены нормально.

По результатам экспериментов, проведенных в Санкт-Петербурге (рис. 7),

установлено, что объем выборки, необходимой для адекватной оценки точности систем спутникового позиционирования и их парных комбинаций, составляет 180 с. Наилучшая комбинация — GPS+BeiDou (см. рис. 6).

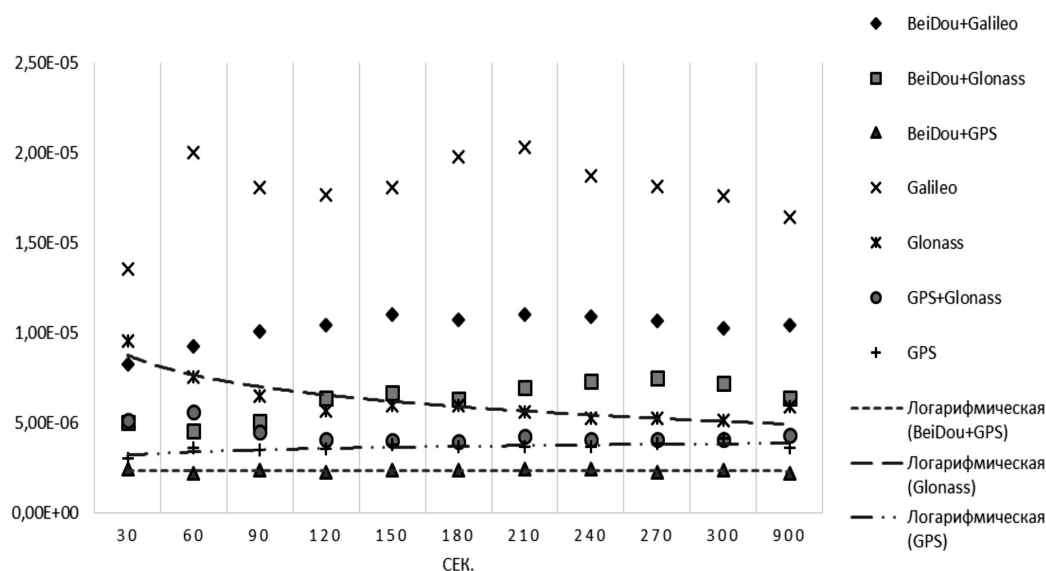


Рис. 7. Зависимость среднего расстояния до «эталонного» центра от размера выборки (Санкт-Петербург)

Fig. 7. Dependence of the average distance to the «reference» center on the sample size (St. Petersburg)

По результатам экспериментов, проведённых в г. Конаклы (Турция), установлено, что объём выборки, необходимой для адекватной оценки точности систем спутникового позиционирования и их парных комбинаций, составляет 120 с. Наилучшая комбинация – GPS+BeiDou.

### Выводы

В работе предложена методика выбора наиболее точной системы спутникового позиционирования или их комбинаций. Представленное решение может использоваться для навигационного обеспечения в задачах управления автономными наземными роботами/транспортными средствами. Использование представленного решения позволяет сократить время выбора наиболее точной системы спутникового позиционирования или парных комбинаций систем в условиях: отсутствия дифференциальных поправок и полей точности систем спутниковой

навигации, сформированных для данной местности, и сведений о начальном географическом положении объекта;

ограниченных вычислительных мощностей автономного объекта;

использования одноканального приёмника навигационных сигналов.

По результатам проведённых исследований можно сделать вывод о том, что при использовании одноканального приёмника навигационных сигналов для выбора наиболее точной системы спутникового позиционирования или их парных комбинаций необходимо с помощью представленного в данной статье алгоритма провести оценку данных, полученных от систем позиционирования и их комбинаций, в объёме 180 с для каждого варианта.

Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 16-29-04319.

### СПИСОК ЛИТЕРАТУРЫ

1. Макаренко Г.К., Алешечкин А.М. Исследование алгоритма фильтрации при определении координат объекта по сигналам спутниковых радионавигационных систем // Доклады ТУСУР. 2012. № 2(26). Ч. 2. С. 15–18.
2. Дворкин В.В., Карутин С.Н. Высокоточные навигационные определения по сигналам ГНСС // Вестник Сибирского гос. аэрокосмического университета им. академика М.Ф. Решетнева. 2013. № 6 (52). С. 70–76.
3. Xingxing Li, Maorong Ge, Xiaolei Dai, Xiaodong Ren, Mathias Fritsche, Jens Wickert, Harald Schuh Accuracy and reliability of multi-GNSS real-time precise positioning: GPS, GLONASS, BeiDou, and Galileo // Journal of Geodesy. 2015. Vol. 89. Issue 6. Pp. 607–635.
4. Xingxing Li, Xiaohong Zhang, Xiaodong Ren, Mathias Fritsche, Jens Wickert, Harald Schuh Precise positioning with current multi-constellation Global Navigation Satellite Systems: GPS, GLONASS, Galileo and BeiDou // Scientific Reports 5. 2015. Article No. 8328.
5. Глобальная служба навигационных спутниковых систем. The International GNSS Service // URL: <http://www.igs.org/> (Дата обращения: 01.09.2017).
6. Зайцев Г.Ф. Теория автоматического управления и регулирования. 2-е изд., перераб. и доп. Киев: Выща шк. Головное изд-во, 1989. 431 с.
7. Скрыпник О.Н., Нечаев Е.Е., Арёфьев Р.О. Построение и анализ полей точности GPS на основе программно-аппаратных средств NI GPS Simulation Toolkit // Научный вестник Московского государственного технического университета гражданской авиации. 2014. № 209. С. 5–12.
8. Гаврилов А.В. Использование фильтра Калмана для решения задач уточнения координат БПЛА // Современные проблемы науки и образования. 2015. № 1-1.
9. Schüler T. On ground-based GPS tropospheric delay estimation. München: Univ. der Bundeswehr, 2001. 364p.
10. Хуторова О.Г., Тептин Г.М., Васильев А.А., Хуторов В.Е., Шлычков А.П. Пассивное зондирование структуры коэффициента преломления радиоволн в тропосфере сетью приёмников спутниковых навигационных систем в г. Казани // Известия вузов. Радиофизика. 2011. Т. 54. № 1. С. 1–8.

Статья поступила в редакцию 04.11.2017.

## REFERENCES

1. **Makarenko G.K., Aleshechkin A.M.** Study on filtering algorithms to determine the coordinates of an object by the signals of satellite navigation systems. *Proceedings of TUSUR*, 2012, No. 2 (26), Part 2. (rus)
2. **Dvorkin V.V., Karutin S.N.** Precise positioning according to GNSS signal. *Vestnik Sibirskogo gosudarstvennogo aerokosmicheskogo universiteta imeni akademika M.F. Reshetneva (VestnikSibGAU)*, 2013, No. 6 (52), Pp. 70–76. (rus)
3. **Xingxing Li, Maorong Ge, Xiaolei Dai, Xiaodong Ren, Mathias Fritsche, Jens Wickert, Harald Schuh** Accuracy and reliability of multi-GNSS real-time precise positioning: GPS, GLONASS, BeiDou, and Galileo. *Journal of Geodesy*, 2015, Vol. 89, Issue 6, Pp. 607–635.
4. **Xingxing Li, Xiaohong Zhang, Xiaodong Ren, Mathias Fritsche, Jens Wickert, Harald Schuh** Precise positioning with current multi-constellation Global Navigation Satellite Systems: GPS, GLONASS, Galileo and BeiDou. *Scientific Reports* 5, 2015, Article No. 8328.
5. *Globalnaya sluzhba navigatsionnykh sputnikovykh sistem. The International GNSS Service*. Available: <http://www.igs.org/> (Accessed: 01.09.2017).
6. **Zaytsev G.F.** *Teoriya avtomaticheskogo upravleniya i regulirovaniya [Theory of automatic control and regulation]*. Kiyev: Vyshcha shk. Publ., 1989, 431 p. (rus)
7. **Skrypnik O.N., Nechayev Ye.Ye., Arefyev R.O.** Construction and analysis of GPS accuracy fields on the basis of hardware-software means NI GPS Simulation Toolkit. *Scientific bulletin of the Moscow state technical university of civil aviation*, 2014, No. 209, Pp. 5–12. (rus)
8. **Gavrilov A.V.** Ispolzovaniye filtra Kalmana dlya resheniya zadach utochneniya koordinat BPLA [Using the Kalman filter to solve the UAV coordinate refinement problems]. *Sovremennyye problemy nauki i obrazovaniya [Modern problems of science and education]*, 2015, No. 1-1. (rus)
9. **Schüler T.** On ground-based GPS tropospheric delay estimation. München: Univ. der Bundeswehr, 2001, 364 p.
10. **Khutorova O.G., Teptin G.M., Vasilyev A.A., Khutorov V.Ye., Shlychkov A.P.** Passivnoye zondirovaniye struktury koeffitsiyenta prelomleniya radiovoln v troposphere setyu priyemnikov sputnikovykh navigatsionnykh sistem v g. Kazani [Sensing of the structure of the radio wave refractivity in the troposphere by a network of satellite navigation system receivers in the city of Kazan]. *Izvestiya vuzov. Radiofizika [Radiophysics and Quantum Electronics]*, 2011. Vol. 54, No. 1, Pp. 1–8. (rus)

Received 04.11.2017.

## СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

**КУРОЧКИН Леонид Михайлович**

**KUROCHKIN Leonid M.**

E-mail: kurochkinl@spbstu.ru

**КУРОЧКИН Михаил Александрович**

**KUROCHKIN Mikhail A.**

E-mail: kurochkin\_ma@spbstu.ru

**ПОПОВ Сергей Геннадьевич**

**POPOV Sergey G.**

E-mail: popovserge@spbstu.ru

**ПОПОВ Михаил Викторович**

**POPOV Mikhail V.**

E-mail: spbmvp@gmail.com



DOI: 10.18721/JCSTCS.10408

УДК 004.523

## СИСТЕМА СОНИФИКАЦИИ В ЗАДАЧЕ НАХОЖДЕНИЯ ПУТИ И ОБХОДА ПРЕПЯТСТВИЙ

*А.А. Мархотин<sup>1</sup>, А.В. Кривошейкин<sup>1</sup>, И.Н. Осипенко<sup>2</sup>, Г.Г. Rogozинский<sup>3</sup>*

<sup>1</sup> Санкт-Петербургский государственный университет кино и телевидения,  
Санкт-Петербург, Российская Федерация;

<sup>2</sup> Saber Interactive, Санкт-Петербург, Российская Федерация;

<sup>3</sup> Санкт-Петербургский государственный университет телекоммуникаций  
имени профессора М.А. Бонч-Бруевича, Санкт-Петербург, Российская Федерация

Приведена модель системы сонификации данных для нахождения пути и обхода препятствий. При отсутствии препятствий по ходу движения звуковой сигнал, генерируемый системой, указывает направление на цель. В случае возникновения препятствий звуковой сигнал указывает направление безопасного движения в зависимости от текущего положения пользователя. Отображение данных о близости препятствий происходит при помощи математического аппарата нечёткой логики. Система разработана в среде графического программирования Pure Data. Приведены описание и результаты эксперимента прохождения виртуального пространства с препятствиями испытуемыми, ориентировавшимися при помощи звука.

**Ключевые слова:** сонификация; нечёткая логика; навигация; поиск пути; вспомогательные технологии.

**Ссылка при цитировании:** Мархотин А.А., Кривошейкин А.В., Осипенко И.Н., Rogozинский Г.Г. Система сонификации в задаче нахождения пути и обхода препятствий // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2017. Т. 10. № 4. С. 89–96. DOI: 10.18721/JCSTCS.10408

## SONIFICATION SYSTEM IN FINDING ROUTE AND DETOUR OF OBSTACLES

*A.A. Markhotin<sup>1</sup>, A.V. Krivosheykin<sup>1</sup>, I.N. Osipenko<sup>2</sup>, G.G. Rogozinsky<sup>3</sup>*

<sup>1</sup> St. Petersburg State University of Film and Television, St. Petersburg, Russian Federation;

<sup>2</sup> Saber Interactive, St. Petersburg, Russian Federation;

<sup>3</sup> Bonch-Bruevich St. Petersburg State University of Telecommunications,  
St. Petersburg, Russian Federation

The paper presents a sonification system for finding route and detour of obstacles. In the absence of obstacles in the direction of motion, the sound signal generated by the system indicates the direction to the target. If there are obstacles, the sound signal

indicates the direction of safe movement depending on the current position of the user. The data representation concerning closeness of the obstacles uses the mathematical apparatus of fuzzy logic. The system was designed in the graphical programming environment Pure Data. The description and the results of the experiment on passing through the virtual space with obstacles by several testees using sound are presented.

**Keywords:** sonification; fuzzy logic; navigation; pathfinding; assistive technology.

**Citation:** Markhotin A.A., Krivosheykin A.V., Osipenko I.N., Rogozinsky G.G. Sonification system in finding route and detour of obstacles. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2017, Vol. 10, No. 4, Pp. 89–96. DOI: 10.18721/JCSTCS.10408

## Введение

Процесс ориентации в пространстве, поиск пути и обход возникающих препятствий связан с непрерывным сбором, анализом и интерпретацией информации об этом пространстве. Обычно основной информацией такого типа, особенно для человека, является информация визуальная. Однако в том случае, когда визуальное восприятие затруднено, в качестве приемника информации об окружающем пространстве можно использовать другие биосенсорные системы человека, и в первую очередь речь идет о слуховой системе. Характерный пример — ультразвуковой датчик парковки [1], более известный как парктроник (parking sensor), который в настоящее время повсеместно используется водителями автотранспорта.

Однако применение сонификации, т. е. представления информации в виде неречевых звуковых сигналов [2], в случае управления транспортным средством может быть значительно более сложным в плане отображения данных об окружающем пространстве. Примером тому служит посадка летательного аппарата на поверхность с неизвестным рельефом в условиях плохой видимости. Ещё менее формализованной является задача ориентации в пространстве человека с ограниченными возможностями по зрению. В данном направлении проводились различные исследования, в рамках которых разрабатывались специальные технологии и устройства: от навигации при помощи времяпролётной камеры (time-of-flight) [3] до распознавания объектов при помощи озвучивания изображений [4].

Подробный обзор наиболее значимых

разработок, анализ и сравнение данных устройств в соответствии с их принципом действия и спецификой представлен в [5–7] и находится за рамками данной статьи. Выводы, приведенные в указанных выше исследованиях, заключаются в следующем. На данный момент не существует универсального устройства, которое обеспечивало бы комфортное ориентирование в пространстве людей с ограниченными возможностями по зрению. Самым популярным таким «устройством» по-прежнему остаётся белая трость. Этим определяется актуальность дальнейших исследований в данном направлении и, в частности, исследования, приведенного в рамках настоящей статьи.

Таким образом, цель нашего исследования — разработка модели системы сонификации с нечётким отображением данных для решения задачи нахождения пути и обхода препятствий.

## Система сонификации

Система сонификации для нахождения пути и обхода препятствий включает в себя два основных функциональных элемента: блок формирования звукового сигнала, указывающего направление на конечную цель, и блок нечеткого отображения данных об окружающих препятствиях (рис. 1).

Реализация концепции отображения данных об окружающем пространстве при помощи изменения параметров звукового сигнала в рамках данной системы сонификации заключается в следующем. Пользователь через наушники (головные телефоны) слышит генерируемый системой сигнал, баланс уровней между правым и левым каналом которого указывает направление на

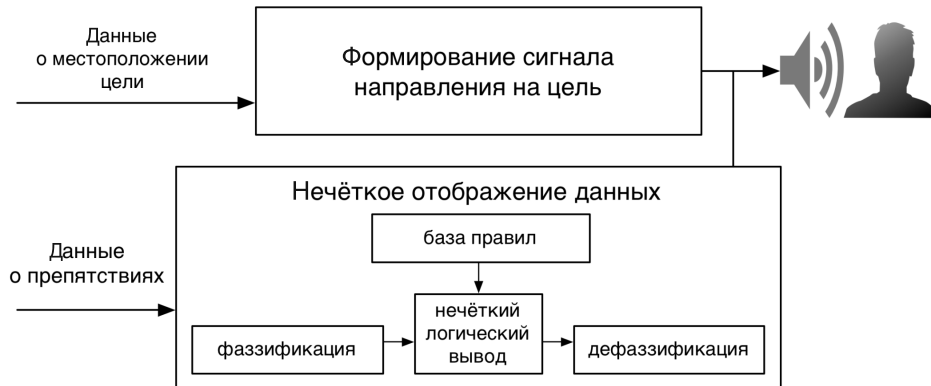


Рис. 1. Структурная схема системы сонификации

Fig. 1. Block diagram of the sonification system

цель. При возникновении препятствий на пути блок нечёткого отображения данных изменяет параметры синтезируемого звука так, что баланс сигнала смещается в ту или иную сторону в зависимости от того, с какой стороны эффективнее будет обойти препятствие. Нечёткость отображения данных о препятствиях заключается в отношении между расстоянием до препятствия и силой отклонения сигнала. Преобразование данных при помощи нечёткого вывода происходит по следующим правилам (рис. 2):

- 1) если препятствие близко, то отклонить сигнал сильно;
- 2) если препятствие недалеко, то отклонить сигнал несильно.

В данном случае «близко», «недалеко», «отклонить сильно» и «отклонить несильно» есть ни что иное как нечёткие множества входных и выходных данных соответственно.

Использование методов нечёткой логики в рамках разработанной системы сонификации обусловлено преимуществами этих методов с точки зрения простоты формализации зависимости отклонения сигнала по панораме от близости препятствия и отсутствием необходимости создания сложного математического описания данного процесса. По этой же причине нечёткая логика широко применяется в разработках систем навигации для мобильных роботов

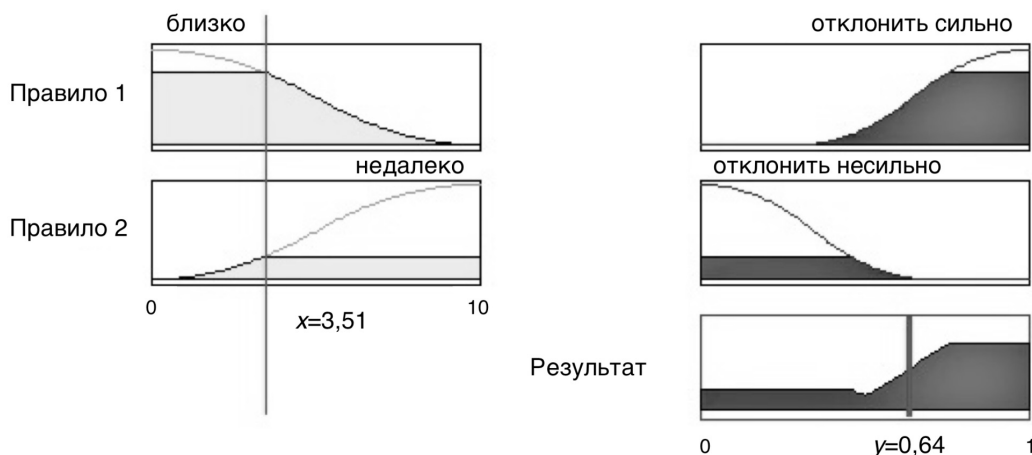


Рис. 2. Правила нечёткого вывода системы сонификации

Fig. 2. Fuzzy inference of the sonification system

[8–11]. Здесь также стоит отметить, что помимо традиционных промышленных задач [12] нечёткая логика находит своё применение в трудноформализуемых областях человеческой деятельности и, в частности, в работе со звуком, а именно в вопросах синтеза звука и алгоритмической композиции [13].

Кроме того, авторы исследовали применение нечёткого отображения при разработке систем сонификации данных сенсорной активности [14, 15], что является актуальным направлением развития этой системы в области создания реальных устройств. В настоящий момент система сонификации представляет собой виртуальную модель, описание которой приведено далее.

### Эксперимент

В ходе эксперимента, целью которого была проверка работоспособности разработанной модели системы сонификации, испытуемым было предложено достичь цели в виртуальном пространстве, преодолевая препятствия и ориентируясь исклю-

чительно при помощи звука. Виртуальное пространство было создано в среде графического программирования Pure Data. На рис. 3 представлен интерфейс программы – виртуальное пространство, в котором точка ближе к низу экрана обозначает пользователя, круг и квадрат – препятствия, линия наверху – финишная черта, её пересечение обозначает достижение цели.

Здесь стоит отметить, что на этапе виртуального моделирования процесса ориентации не ставится задача обнаружения и распознавания препятствий. Системе сонификации известны местоположение, форма, и расстояние до препятствий, однако они неизвестны пользователю. Эксперимент состоит в проверке адекватности нечёткого отображения данных о препятствиях реакциям пользователя, при которых путь в обход препятствий будет найден. Блок-схема алгоритма программы представлена на рис. 4.

При первом запуске программы на стадии обучения пользователю предлагалось, основываясь на зрительном восприятии,

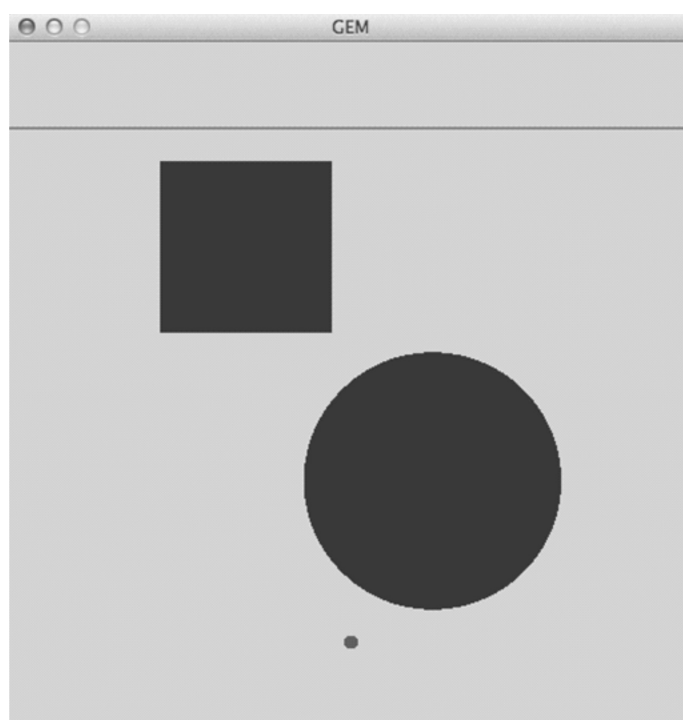


Рис. 3. Окно виртуального пространства в Pure Data

Fig. 3. The virtual space

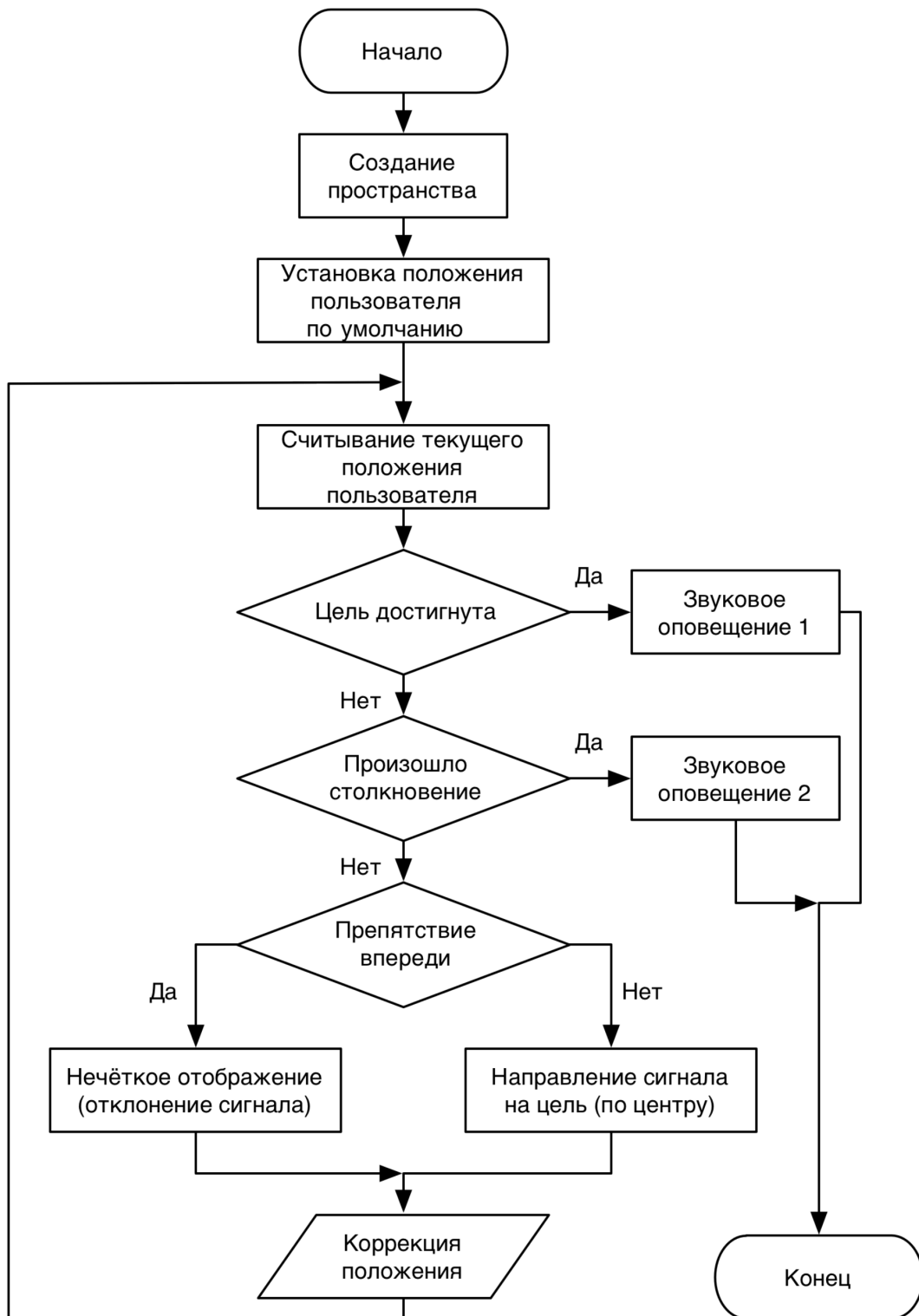


Рис. 4. Блок-схема алгоритма программы симуляции нахождения пути и обхода препятствий

Fig. 4. Flowchart of the program



**Результаты прохождения сцен**  
**The results of passing scenes**

|         | Испытуемый  |             |          |          |
|---------|-------------|-------------|----------|----------|
|         | 1           | 2           | 3        | 4        |
| Сцена 1 | Пройдена    | Пройдена    | Пройдена | Пройдена |
| Сцена 2 | Не пройдена | Пройдена    | Пройдена | Пройдена |
| Сцена 3 | Пройдена    | Пройдена    | Пройдена | Пройдена |
| Сцена 4 | Пройдена    | Не пройдена | Пройдена | Пройдена |

сопоставить информацию о препятствиях и звуковом сигнале, в зависимости от своего положения. Затем, при запуске последующих сцен с другим расположением препятствий, от пользователя требовалось не глядя на экран достичь финишной черты.

**Результаты эксперимента**

В эксперименте принимали участие четыре человека. Данные о прохождении испытуемыми различных сцен с отметкой о прохождении сведены в таблицу.

Выбранное небольшое число испытуемых и возможных сцен обусловлено тем, что на данном этапе исследования в первую очередь требуется проверка обоснованности выбора самого метода отображения данных, т. е. ответ на вопрос о его пригодности в дальнейших разработках. Таким образом, проводить точные измерения численных показателей эффективности выбранного метода пока преждевременно.

Тем не менее, как видно из таблицы, из 16 попыток прохождения сцены испытуемыми успешно завершились 14. Следовательно, практически в 90 % случаев испытуемый смог пройти незнакомую сцену, ориентируясь только по звуку. Допущенные ошибки возникли в случаях, когда непосредственно впереди движения препятствий не было, и звуковой сигнал указывал прямо. Однако при этом препятствия располагались близко справа или слева от направления движения, и любое малейшее отклонение от курса приводило к столкновению.

**Заключение**

Исходя из результатов исследования можно сделать практический вывод, что кроме направления непосредственно на цель следует ввести параллельно информацию о близости препятствий, даже если они не находятся прямо по курсу движения. Этого не было сделано изначально, т. к. авторы сочли целесообразным не перегружать звуковой ландшафт дополнительной информацией, чтобы впоследствии, при переходе от модели к реальному миру, сохранить максимальное количество доступного внимания человека для звуков окружающей среды.

Естественно, поскольку данная модель является первым шагом на пути к полноценной системе сонификации для нахождения пути и обхода препятствий и представляет собой виртуальную двухмерную модель, говорить о полной её адекватности реальному миру преждевременно. При разработке прикладных решений на базе предложенного подхода для таких задач, как управление транспортным средством в условиях затрудненной видимости или помощь человеку с ограниченными возможностями по зрению, неизбежно возникает вопрос обнаружения и распознавания препятствий, что является темой для отдельного исследования. Однако принципиальные моменты, такие как нечеткое отображение данных о близости препятствия в параметры отклонения звукового сигнала и сама концепция ведения таким образом пользователя при помощи звука между препятствиями, могут быть сохранены и проработаны более детально в дальнейших исследованиях.

## СПИСОК ЛИТЕРАТУРЫ

1. Mercedes Benz Parktronic system (PTS) // URL: <http://www.micro-tronik.com/Support/Info/Benz-PTS.htm> (Дата обращения: 24.04.2017).
2. Sonification report: Status of the field and research agenda // URL: <http://www.icad.org/websiteV2.0/References/nsf.html> (Дата обращения: 24.04.2017).
3. Brock M., Kristensson P.O. Supporting blind navigation using depth sensing and sonification // Proc. of the 2013 ACM Conf. on Pervasive and Ubiquitous Computing Adjunct Publication. Zurich, 2013. Pp. 255–258. DOI: 10.1145/2494091.2494173
4. Yoshida T., Kitani K.M., Koike H., Belongie S., Schlei K. EdgeSonic: Image Feature Sonification for the Visually Impaired // Proc. of the 2nd Augmented Human Internat. Conf. Tokyo, 2011. Article No. 11. DOI: 10.1145/1959826.1959837
5. Brabyn J.A. New developments in mobility and orientation aids for the blind // IEEE Trans. Biomedical Engineering. 1982. Vol. 29(4). Pp. 285–289.
6. Shoval S., Ulrich I., Borenstein J. Computerized obstacle avoidance systems for the blind and visually impaired // Intelligent Systems and Technologies in Rehabilitation Engineering. CRC Press: Boca Raton, 2001. Pp. 413–447.
7. Hermann T., Hunt A. The Sonification Handbook. Berlin: Logos Verlag, 2011. 564 p.
8. Faisal M., Hedjar R., Sulaiman M., Al-Mutib K. Fuzzy logic navigation and obstacle avoidance by a mobile robot in an unknown dynamic environment // Internat. Journal of Advanced Robotic Systems. 2013. Vol. 10 // URL: <http://journals.sagepub.com/doi/10.5772/54427> (Дата обращения: 24.04.2017).
9. Kumar Pradhan S., Ramakrushna Parhi D., Kumar Panda A. Fuzzy logic techniques for navigation of several mobile robots // Applied Soft Computing. 2009. Vol. 9. No. 1. Pp. 290–30.
10. Raudonis V., Maskeliunas R. Trajectory based fuzzy controller for indoor navigation. Computational Intelligence and Informatics. 2011. Pp. 69–72.
11. Das T., Kan N. Design and implementation of an adaptive fuzzy logic based controller for wheeled mobile robots // IEEE on Control Systems Technology. 2006. Vol. 14. No. 3. Pp. 501–510.
12. Тэрано Т., Асаи К., Сугэно М. Прикладные нечеткие системы. М.: Мир, 1993. 366 с.
13. Рогозинский Г.Г., Мархотин А.А. Методы нечеткой логики в компьютерной музыке и алгоритмической композиции // Актуальные проблемы инфотелекоммуникаций в науке и образовании. IV науч.-техн. и науч.-метод. конф. СПб., 2015. С. 362–366.
14. Мархотин А.А., Кривошейкин А.В., Рогозинский Г.Г., Уолш Р. Нечеткое отображение в системе сонификации данных беспроводной сенсорной сети // Научно-технический вестник информационных технологий, механики и оптики. 2016. Т. 16. № 6. С. 1073–1077. DOI: 10.17586/2226-1494-2016-16-6-1073-1077
15. Ефимов В.В., Рогозинский Г.Г., Мархотин А.А., Чесноков М.А. Нечёткое отображение в процессе сонификации данных сенсорных сетей // Труды ЦНИИС. СПб филиал. 2016. Т. 1. № 1 (2). С. 77–84.

Статья поступила в редакцию 28.04.2017.

## REFERENCES

1. Mercedes Benz Parktronic system (PTS). Available: <http://www.micro-tronik.com/Support/Info/Benz-PTS.htm> (Accessed: 24.04.2017).
2. Sonification report: Status of the field and research agenda. Available: <http://www.icad.org/websiteV2.0/References/nsf.html> (Accessed: 24.04.2017).
3. Brock M., Kristensson P.O. Supporting blind navigation using depth sensing and sonification. *Proceedings of the 2013 ACM conference on Pervasive and ubiquitous computing adjunct publication*, Zurich, 2013, Pp. 255–258. DOI: 10.1145/2494091.2494173
4. Yoshida T., Kitani K.M., Koike H., Belongie S., Schlei K. EdgeSonic: Image Feature Sonification for the Visually Impaired. *Proceedings of the 2nd Augmented Human International Conference*, Tokyo, 2011, Article No. 11. DOI: 10.1145/1959826.1959837
5. Brabyn J.A. New developments in mobility and orientation aids for the blind. *IEEE Trans. Biomedical Engineering*, 1982, Vol. 29(4), Pp. 285–289.
6. Shoval S., Ulrich I., Borenstein J. Computerized obstacle avoidance systems for the blind and visually impaired. *Intelligent Systems and Technologies in Rehabilitation Engineering*, CRC Press: Boca Raton, 2001, Pp. 413–447.
7. Hermann T., Hunt A. *The Sonification Handbook*. Berlin, Germany: Logos Verlag, 2011, 564 p.
8. Faisal M., Hedjar R., Sulaiman M., Al-Mutib K. Fuzzy logic navigation and obstacle avoidance by a mobile robot in an unknown

dynamic environment. *International Journal of Advanced Robotic Systems*, 2013, Vol. 10. Available: <http://journals.sagepub.com/doi/10.5772/54427> (Accessed: 24.04.2017).

9. Kumar Pradhan S., Ramakrushna Parhi D., Kumar Panda A. Fuzzy logic techniques for navigation of several mobile robots. *Applied Soft Computing*, 2009, Vol. 9, No. 1, Pp. 290–30.

10. Raudonis V., Maskeliunas R. Trajectory based fuzzy controller for indoor navigation. *Computational Intelligence and Informatics*. 2011, Pp. 69–72.

11. Das T., Kan N. Design and implementation of an adaptive fuzzy logic based controller for wheeled mobile robots. *IEEE on Control Systems Technology*, 2006, Vol. 14, No. 3, Pp. 501–510.

12. Terano T., Asai K., Sugeno M. *Prikladnyye nechetkiye sistemy* [Applied Fuzzy Systems]. Moscow: Mir Publ., 1993, 366 p. (rus)

13. Rogozinskiy G.G., Markhotin A.A. Metody

nechetkoy logiki v kompyuternoy muzyke i algoritmicheskoy kompozitsii [Methods of fuzzy logic in computer music and algorithmic composition]. *Actual Problems of Information Telecommunications in Science and Education. IV Scientific-Technical and Scientific-Methodical Conference*, St. Petersburg, 2015, Pp. 362–366. (rus)

14. Markhotin A.A., Krivosheykin A.V., Rogozinskiy G.G., Uolsh R. Fuzzy mapping in data sonification system of wireless sensor network. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2016, Vol. 16, No 6, Pp. 1073–1077. DOI: 10.17586/2226-1494-2016-16-6-1073-1077

15. Yefimov V.V., Rogozinskiy G.G., Markhotin A.A., Chesnokov M.A. Nechetkoye otobrazheniye v protsesse sonifikatsii dannykh sensorykh setey [Fuzzy mapping in sensor networks sonification]. *Proceedings of the CNIIS. St. Petersburg Branch*, 2016, Vol. 1, No. 1 (2), Pp. 77–84. (rus)

Received 28.04.2017.

#### СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

**МАРХОТИН Арсений Андреевич**

**MARKHOTIN Arseny A.**

E-mail: markhotin@gmail.com

**КРИВОШЕЙКИН Анатолий Валентинович**

**KRIVOSHEYKIN Anatoly V.**

E-mail: krivav@yandex.ru

**ОСИПЕНКО Иван Николаевич**

**OSIPENKO Ivan N.**

E-mail: ivanosipenko@gmail.com

**РОГОЗИНСКИЙ Глеб Гендрихович**

**ROGOZINSKY Gleb G.**

E-mail: gleb.rogozinsky@gmail.com



DOI: 10.18721/JCSTCS.10409

УДК 004.896

## УПРАВЛЕНИЕ ГРУППОЙ АВТОНОМНЫХ РОБОТОВ С ИСПОЛЬЗОВАНИЕМ ПОЛЯРНЫХ КООРДИНАТ

*В.А. Онуфриев*

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

Рассмотрена проблема координирования группы автономных роботов с целью перемещения из произвольных позиций в определенную точку. Предложен подход, в рамках которого статичный лидер группы на основе дальномеров вычисляет полярные координаты каждого робота, которые затем передаются по беспроводной сети и используются для вычисления направления и пути перемещения к определенной цели. Приведена алгоритмическая модель, применяемая для вычисления и сохранения координат роботов ведущим. Описан предлагаемый способ коммуникации между роботами на базе сетей Wi-Fi и стека протоколов TCP/IP. Определен требуемый набор бортового оборудования для каждого робота.

**Ключевые слова:** подвижные роботы; групповое управление; сетевая система; интеллектуальные системы; киберфизические системы.

**Ссылка при цитировании:** Онуфриев В.А. Управление группой автономных роботов с использованием полярных координат // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2017. Т. 10. № 4. С. 97–106. DOI: 10.18721/JCSTCS.10409

## AUTONOMOUS ROBOTS GROUP CONTROL USING POLAR COORDINATES

*V.A. Onufriev*

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

Autonomous robots group coordination in order to move from arbitrary position to specific point is considered in this paper. The author proposes the approach where the static group's leader computes each robot's polar coordinates based on the distance sensor's measures. Then it transmits these coordinates via a wireless network to be used for direction and disposition computations. The algorithmic model of this process, which is used for computation and saving the robots' coordinates is presented. The paper also describes the proposed methods of the robots' communication based on Wi-Fi networks and TCP/IP stack. In the conclusion a minimal set of on-board equipment is defined.

**Keywords:** mobile robots; group control; network-centric system; intelligent systems; cyber-physical system.

**Citation:** Onufriev V.A. Autonomous robots group control using polar coordinates. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2017, Vol. 10, No. 4, Pp. 97–106. DOI: 10.18721/JCSTCS.10409

**Введение.** Известно, что в настоящее время роботы используют в различных областях науки и техники. Чаще всего это связано с работой в опасных либо вредных

для человека условиях [7, 8]. Эти направления развиваются давно и успешно, но для многих задач не может привлекаться только один робот ввиду того, что его возмож-

ности всегда будут ограничены имеющимся на борту оборудованием, зарядом батарей, ограниченным радиусом действия и т. д. Для снятия этого ограничения следует использовать сразу несколько роботов, которые могут выполнять разные действия, но связаны при этом общей целью.

Несмотря на то что существуют исследования, в которых каждый робот следует заложенному в него алгоритму «без оглядки» на остальных [14], групповое управление с взаимодействием между роботами видится более перспективным, т. к. в этом случае здесь может быть организована сетцентрическая система управления [3], подразумевающая принятие решений внутри системы, ее общую открытость и постановку целей самой системой. Возможность коммуникации между роботами (т. е. частями системы) позволяет лучше оценивать имеющиеся ресурсы и принимать решения в условиях неопределенности. Кроме того, коммуникация между автономными устройствами отвечает современному тренду кибер-физических систем [12, 15]. С повышением общего уровня неопределенности эффективность систем, не способных адаптироваться и перестраиваться, заметно падает [1].

Управление группой роботов несет в себе множество дополнительных задач и сложностей, однако главная из них — определение местоположения каждого отдельного робота [4]. Решению этой задачи и посвящена статья.

Задача определения местоположения сводится к определению вектора координат для любого робота, достаточным образом описывающего его нахождение.

Основная проблема состоит в том, что не существует более-менее универсального решения этой задачи, она решается для каждой конкретной ситуации. Современное «state of arts» координирования роботов подразумевает для каждого конкретного случая разработку частного решения, которое затем уже нельзя применить для других, более общих, задач [5, 10].

Для универсализации решения данной задачи можно было бы использовать бор-

товое оборудование с мощными процессорами и встроенными GPS-модулями, как в работах [13, 16], однако подобные решения удорожают стоимость каждого робота в отдельности, что в пересчете на их группу делает затраты на проект необоснованно высокими. Сюда же относится и применение видеокамер с последующей обработкой изображения.

Таким образом, основная проблема — отсутствие универсального способа определения местоположений — будет решаться в данной статье с учетом главного ограничения: стоимости бортового оборудования роботов.

**Постановка задачи.** Имеем группу из  $N$  роботов, в начальный момент времени находящихся в заранее неизвестных точках пространства любой геометрической формы. Другие предметы в этом пространстве отсутствуют либо имеют размеры намного меньше размеров роботов.

Перед нами две задачи:

1) создание алгоритма определения точного местоположения каждого робота. При этом алгоритм должен быть применим в указанных выше условиях (т. е. в общем случае);

2) определение набора бортовых устройств, достаточного для вычисления местоположения робота. Количество устройств должно быть минимальным, т. к. они увеличивают стоимость робота, потребляют электрическое питание и усложняют схему подключения.

Решение этих задач позволит создать децентрализованную систему из роботов, которые по сигналу любого из них способны перемещаться к роботу, подавшему сигнал. При этом все роботы подвижны и каждый из них в момент сигнала может находиться в произвольной точке. Автономность роботов подразумевает отсутствие сигналов управления с удаленного компьютера. Кроме того, должна быть организована возможность получения на компьютере информации о текущем положении роботов. Такая система может применяться для создания поисковой группы роботов, перемещающейся к участнику, нашедшему искомым предмет, который может быть





найден в любой точке пространства заранее неизвестного размера и формы.

**Существующие способы определения местоположения.** Проблема определения местоположения роботов в условиях неопределенности не является тривиальной. В отличие от описанной в [4] системе, где роботы перемещаются исключительно вдоль линий с координатными метками на ней, в более общем случае, очевидно, невозможно предусмотреть заранее заданной координатной сети. Использование роботами статичных объектов в качестве «ориентиров», как в [11] (система позиционирования на основе двух неподвижных ИК-маяков), также не может претендовать на роль более общего решения ввиду неподвижности маяков и необходимости их установки в заранее определенных точках, является лишь узкоспециализированным способом решения проблемы координации.

Обратим внимание на то, что функция определения координат робота может, в самом общем случае, выполняться *либо самим роботом, либо каким-то внешним устройством*. Изучим обе эти группы подходов.

Первый способ координации можно отнести к первой группе подходов. Он принципиально похож на рассмотренный в работе [11], где робот измеряет своё расстояние до статичных ИК-маяков и на основе этого вычисляет свои координаты.

Рассмотрим прямоугольную комнату неизвестных заранее размеров (рис. 1).

Находящийся в произвольной точке робот должен совершить множество измерений, чтобы однозначно определить свое местоположение, т. к. ему нужно определить кратчайшие расстояния до двух стен. Однако робот может занять четыре различных положения с одинаковыми расстояниями  $a$  и  $b$ : расстояниями до левой и правой стен соответственно. Для решения этой задачи необходимо ввести еще измерения углов 1 и 2, показанных на рис. 1, а также угол 3 (направляющий к одной из стен либо к полюсу Земли), для чего должен быть использован магнитометр.

Роботы свободно перемещаются по пространству в поисках определенного предмета, самостоятельно определяя свои координаты.

Преимущество данного способа — простота алгоритма, а недостатки — требование отсутствия посторонних предметов в комнате, ограничение объема комнаты расстоянием, равным радиусу действия дальномеров, и ограничение на форму комнаты и стен.

Таким образом, данный способ не подходит для решения задачи при использовании комнаты произвольной формы и с возможностью нахождения посторонних предметов.

Принципиально иным является второй способ ориентировки в пространстве, который можно использовать для децентрализованной поисковой системы. Он относится

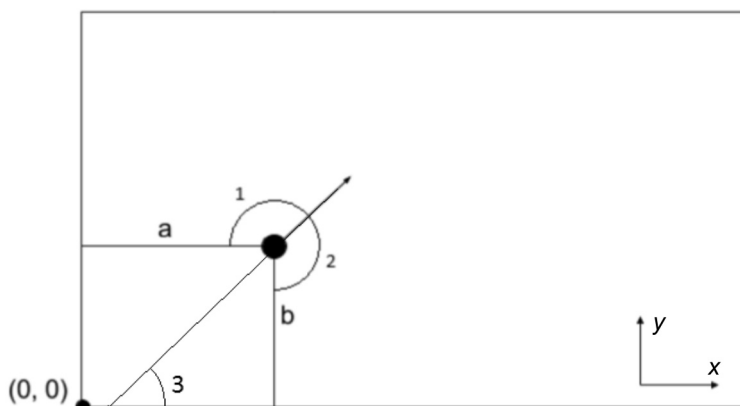


Рис. 1. Первый подход к определению местоположения робота

Fig. 1. The first approach to define the robot's position

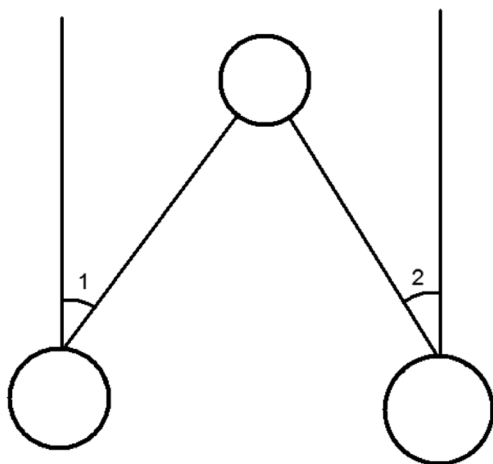


Рис. 2. Второй подход к определению местоположения робота

Fig. 2. The second approach to define the robot's position

ко второй группе методов и похож на представленный в работе [18], где группа роботов следует за «лидером», опираясь только на свое расположение относительно него. В этом случае у роботов нет необходимости вычислять свои координаты. Для применения данного подхода, например, в задаче поиска объектов, каждый робот должен быть способен одновременно отслеживать местоположение ведущего и сканировать территорию в поисках предмета, а ведущий должен «вести» всю группу в определенном направлении.

На рис. 2 находящиеся сзади роботы должны поддерживать значение углов 1 и 2 и расстояние до ведущего в определенном интервале.

Главным преимуществом данного подхода является то, что ведомым роботам не требуется вычислять свои координаты, а для слежения за ведущим используются относительно простые алгоритмы и дешевые датчики.

Вместе с тем здесь имеются и недостатки. Каждый робот должен постоянно тратить ресурсы для слежения за ведущим, для чего требуется либо дополнительное процессорное время, либо дополнительное оборудование. Также остается нерешенным вопрос о том, каким образом должен координироваться уже сам ведущий.

**Предлагаемый способ определения местоположений и алгоритмы.** На базе двух описанных подходов мы предлагаем новый, основанный на постулате, что координаты робота должны вычисляться не самим роботом, а другим, внешним по отношению к нему устройством. Так, в работе [17] для этого используются внешние по отношению к роботам программные агенты. В нашем подходе эта задача ложится на ведущего робота (хотя эту роль может играть любой из роботов группы), что обеспечивает децентрализованность и автономность системы.

Ведущий с момента начала решения задачи поиска будет оставаться статичным. Благодаря этому его неизменную позицию можно считать началом координат для всех других роботов. Он обозначен черным ромбом на рис. 3, и его местоположение определено координатами  $(0, 0)$ . Вместе с тем остальные роботы подвижны. На рисунке два из них обозначены как  $a$  и  $b$ .

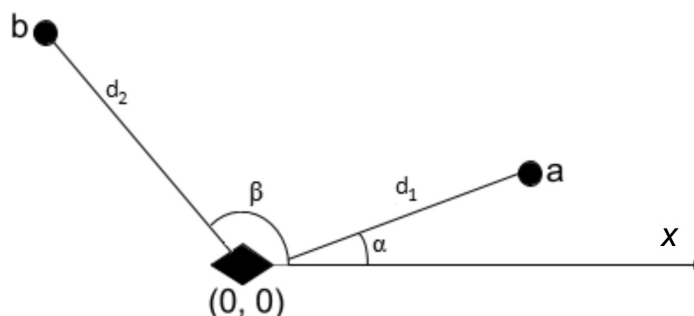


Рис. 3. Предлагаемый подход к определению местоположения робота

Fig. 3. The proposed approach to define the robot's position



Рис. 4. Алгоритм работы ведущего робота

Fig. 4. The leading robot's operation algorithm

Особенность ведущего робота состоит в том, что он снабжен сразу несколькими дальномерами, которые крепятся на сервоприводы. Благодаря этому ведущий всегда обладает информацией о том, на каком расстоянии и в каком направлении от него располагается тот или иной робот. Например, на рис. 3 робот  $a$  расположен на расстоянии  $d_1$  под углом  $\alpha$  к фиксированной оси ведущего, обладая при этом, по сути, полярными координатами  $(d_1, \alpha)$ . Аналогично определяются полярные координаты робота  $b : (d_2, \beta)$ . Ось отсчета  $X$  ведущий робот может выбрать относительно, например, нулевого положения серводвигателя, поворачивающего дальномеры.

Таким образом, ведущий робот, находящийся в произвольной начальной точке, должен непрерывно сканировать пространство вокруг себя, определяя тем самым полярные координаты остальных роботов. Подробнее процесс описан в алгоритмической модели (рис. 4).

Рассмотрим работу на примере задачи поиска группой роботов предмета в комнате заранее неизвестных размеров. Ведущий робот, оставаясь неподвижным и сканируя пространство для определения полярных координат роботов вокруг себя, сохраняет их в своей памяти. Робот, нашедший объект, связывается с ведущим и сообщает о необходимости сигнализировать другим роботам. В ответ ведущий отправляет каждому участнику не только сам сигнал, но и информацию о текущем положении сигнализирующего робота, а также каждому сообщается о его собственном текущем положении, чтобы можно было построить путь к сигнализировавшему роботу. Схема действий отражена на рис. 5.

Последней задачей является координирование перемещения каждого отдельного робота из его текущего положения к точке сигнализирующего участника. Предположим, что робот  $b$  отправил сигнал всей группе роботов. Движение робота  $a$  из по-



Рис. 5. Последовательность при обнаружении объекта

Fig. 5. The action sequence when a new object is detected

лярных координат  $(d_1, \alpha)$  к роботу  $b$  с координатами  $(d_2, \beta)$  может быть описано вектором  $s$  (рис. 6).

Для перемещения к вызвавшему роботу необходимо повернуть в направлении, определяемом углом  $\gamma$ , и пройти путь длиной в модуль вектора  $s$ . Разложив этот вектор по осям, получим, что  $\gamma = \tan^{-1}\left(\frac{s_y}{s_x}\right)$ , где  $s_x$  и  $s_y$  — проекции  $s$  на оси координат. Так как  $s = \vec{d}_2 - \vec{d}_1$ , то получается, что  $s_y = d_2 \sin(\beta) - d_1 \sin(\alpha)$ ,  $s_x = d_2 \cos(\beta) - d_1 \cos(\alpha)$ , а длина вектора вычисляется  $\sqrt{s_x^2 + s_y^2}$ .

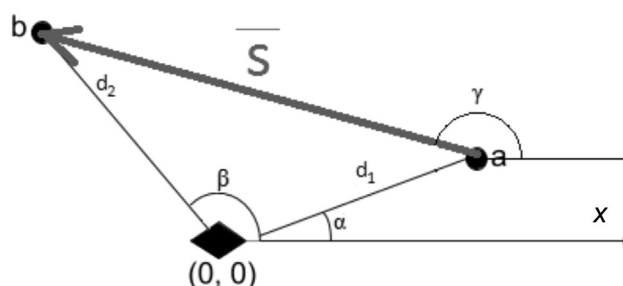


Рис. 6. Вектор перемещения робота в указанную точку

Fig. 6. A vector of robot's displacement to defined position

Сказанное налагает дополнительные требования на бортовое оборудование каждого из роботов. Каждый из роботов должен по требованию повернуть на строго определенный угол в пространстве. Для этого необходим магнитометр — датчик, определяющий текущее направление робота.

Вместе с тем, т. к. отслеживанием и вычислением координат участников группы занимается ведущий робот, необходимо реализовать упомянутый выше канал обмена данными между всеми роботами, поскольку каждый должен иметь возможность запросить у ведущего не только координаты объекта, но и собственные текущие координаты, чтобы построить траекторию перемещения к нему.

**Решение задачи обеспечения коммуникации.** Условие поддержки децентрализованного управления означает, что система должна работать автономно и выполнять задачи даже в случае, если несколько роботов вышли из строя. Из этого следует, что способ коммуникации должен позволять любому роботу в любой момент времени связываться с любым другим вне зависимости от их местоположения и текущих задач. Более того, стремление генерализовать систему для решения более широкого круга задач вынуждает увеличить радиус действия связи между роботами минимум до 100 м на открытой местности и до 25 м в помещении. Однако предложенный выше подход позволяет не налагать специальные требования на канал связи, как, например, в работе [6], где прямо подразумевалось, что канал связи должен не только реализовать передачу данных, но и предусмотреть возможность опре-

деления, с какой стороны пришел сигнал.

Описанный в предыдущем разделе подход для координации роботов осуществим при наличии возможности отправки ведущим данных с координатами любому из подвижных роботов как по запросу от них, так и без запроса.

Участники группы должны самостоятельно организовать сеть для обмена данными между собой. Более того, эта сеть должна быть создана ими даже в том случае, если часть из них выведена из строя.

Таким требованиям соответствуют стандарты передачи данных Wi-Fi и ZigBee. При этом второй, при всех своих достоинствах, в числе которых можно назвать низкое энергопотребление и большую гибкость в организации топологий соединения [9], требует значительно более дорогое оборудование. Использование Wi-Fi в свою очередь подразумевает высокое энергопотребление и меньшую гибкость топологий, и более сложную организацию сети, но меньшую стоимость самих модулей.

Для указанной задачи подходит любой из этих стандартов связи, но их реализация будет очень различаться. Так, в случае с ZigBee вопрос создания сети решается аппаратно, все устройства являются равнозначными. В случае с Wi-Fi (мы создавали сеть на основе Wi-Fi-модулей ESP8266) любой робот может выступать как в роли точки доступа, так и в роли подключаемого к ней устройства. Поэтому достаточно любому из них (например, ведущему) выступить в роли точки доступа, тогда остальные роботы подключатся к нему.

При использовании протоколов передачи данных в случае с ZigBee любое сообщение, отправленное любым роботом, одновременно получают все участники сети либо один определенный участник.

В случае с использованием Wi-Fi необходимо организовать TCP/IP канал. Для этого один из роботов должен стать TCP-сервером, а остальным необходимо подключаться к нему в режиме TCP-клиентов. Подробности приведены на рис. 7.

**Требуемое бортовое оборудование.** Для

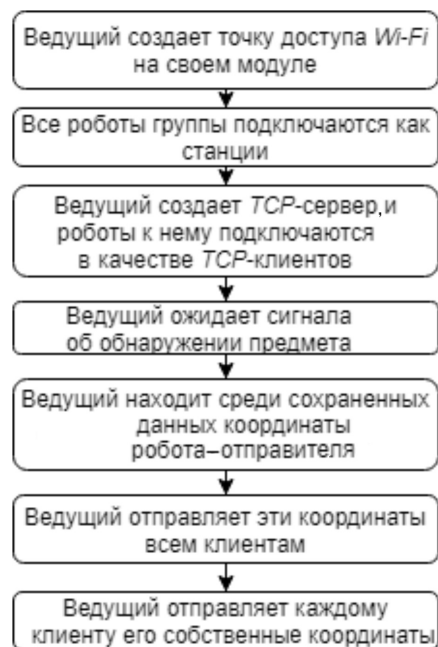


Рис. 7. Реализация последовательности на уровне передачи данных

Fig. 7. The data interchange sequence

реализации описанного выше алгоритма необходимо обеспечить следующее оборудование для участников группы:

модули связи — в данном случае рекомендуем использовать Wi-Fi-модули, например, ESP8266;

датчики расстояния — для этой цели подойдут ультразвуковые дальномеры;

магнитометры — они позволяют определять текущее направление движения робота, что необходимо для вычисления и контроля необходимого поворота;

сервоприводы — для определения угла (относительно оси робота), на котором располагается предполагаемый объект.

Отдельно стоит отметить, что для ведущего робота магнитометры не являются необходимым оборудованием, т. к. предполагается его статичность в основной фазе работы. Вместе с тем ему необходим сервопривод для кругового осмотра местности.

**Заключение.** Преимущество предложенного в статье подхода в том, что для любого подвижного робота задача определения его местоположения решается извне — ведущим роботом. Это позволяет упростить



его бортовое оборудование, уменьшить вес и высвободить вычислительные ресурсы для других задач. Также достоинством является отсутствие привязки к статическим объектам (кроме самого ведущего робота, который может в случае необходимости самостоятельно изменить свое местоположение).

Предложенный алгоритм представляет несложным для выполнения на самых простых и недорогих микроконтроллерах.

В то же время недостатком является необходимость допущения, что в окрестностях ведущего робота отсутствуют рельефные особенности и любые другие предметы, помимо искомого. Кроме того, подвижные роботы могут корпусами закрывать друг друга от ведущего.

Разработанный способ координирова-

ния роботов может использоваться при выполнении таких групповых задач, как патрулирование территории, обезвреживание объектов и поисковые работы.

Вместе с тем для более качественной реализации сетевцентрической системы требуется разработка содержания уточняющих сообщений, определение частоты их отправления, а также онтологии их коллективного поведения [2], которая станет базой для интеллектуальной системы управления.

Следующий шаг в развитии – организация передачи данных между рабочей станцией и ведущим роботом, что может быть сделано при помощи подключения станции к беспроводной сети группы роботов. Для этого потребуется реализация TCP-клиента на станции.

#### СПИСОК ЛИТЕРАТУРЫ

1. Амелин К.С., Амелина Н.О., Граничин О.Н. Адаптивная мультиагентная операционная система реального времени для комплексов БПЛА // Актуальные проблемы Российской космонавтики. Труды XXXVIII академических чтений по космонавтике. М.: Комиссия РАН, 2014. С. 654.
2. Городецкий В.И., Самойлов В.В., Троцкий Д.В. Базовая онтология коллективного поведения автономных агентов и ее расширения // Известия РАН. Теория и системы управления. 2015. № 5. С. 102–121.
3. Ефремов А.Ю., Максимов Д.Ю. Сетевцентрическая система управления – что вкладывается в это понятие? // Труды III Всеросс. конф. с междунар. участием: Технические и программные средства систем управления, контроля и измерения. М.: ИПУ РАН, 2012. С. 158–161.
4. Каляев И.А., Гайдук А.Р., Капустян С.Г. Модели и алгоритмы коллективного управления в группах роботов. М.: Физматлит, 2009. 280 с.
5. Каляев И.А., Гайдук А.Р., Капустян С.Г. Самоорганизация в мультиагентных системах // Известия ЮФУ. Технические науки. Тематический выпуск. 2010. С. 14–20.
6. Карпова И.П. Псевдоаналоговая коммуникация в группе роботов // Мехатроника, автоматизация, управление. 2016. № 2(17). С. 94–101.
7. Куафе Ф. Взаимодействие робота с внешней средой. М.: Мир, 1985. 285 с.
8. Мальцев П.П., Пономарев К.М., Степанов Ю.И. «Умная пыль» на основе микросистемной техники // Интеллектуальные робототехнические системы 2001: материалы молодежной, науч. шк. Таганрог: Изд-во ТРТУ, 2001. С. 220–232.
9. Онуфриев В.А., Удод А.С. Беспроводная система контроля параметров дистанционно распределенных сооружений // Научная сессия ТУСУР-2010: Матер. докладов Всеросс. науч.-техн. конф. студентов, аспирантов и молодых ученых. Томск: В-Спектр, 2010. Ч. 2. С. 170–173.
10. Amato C., Konidaris G.D., Cruz G., Maynor C.A., How J.P., Kaelbling L.P. Planning for decentralized control of multiple robots under uncertainty // Proc. of IEEE Internat. Conf. on Robotics and Automation. IEEE, 2015. Pp. 1214–1248.
11. Blankenship J., Mishal S. Enhancing the Pololu 3pi with Robotbasic. North Charleston SC: Createspace, 2010. 156 p.
12. Brettel M., Friederichsen N., Keller M., Rosenberg M. How virtualization, decentralization and network building change the manufacturing landscape: An industry 4.0 perspective // Internat. J. of Mechanical, Aerospace, Industrial, Mechatronics and Manufacturing Engineering. 2014. Vol. 8. No. 1. Pp. 37–44.
13. Feddema J.T., Lewis C., Schoenwald D.A. Decentralized control of cooperative robotic vehicles: Theory and application // IEEE Transactions on Robotics and Automation. 2002. Vol. 18. No. 5. Pp. 852–864.
14. Guilherme A.S., Kumar R.V., Campos M.F.M. Decentralized algorithms for multirobot



manipulation via caging // Springer Tracts in Advanced Robotics. 2003. Vol. 7. Pp. 257–274.

15. **Onufriev V. A., Sulerova A.S., Poteking V.V.** Cyber-physical systems application for the radio telescope's adaptive surface control task // Symp. on Automated Systems and Technologies. Hannover: PZH Verlag, 2016. Pp. 51–56.

16. **Vásárhelyi G., Virágh Cs., Somorjai G., et al.** Outdoor flocking and formation flight with autonomous aerial robots // IEEE/RSJ Internat. Conf. on Intelligent Robots and Systems. Digest.

Chicago, 2014. Pp. 3866–3873.

17. **Wang Y., de Silva C.W.** A machine-learning approach to multi-robot coordination // Engineering Applications of Artificial Intelligence. 2008. Vol. 21(3). Pp. 470–484.

18. **Wang Z., Takano Y., Hirata Y., Kosuge K.** A pushing leader based decentralized control method for cooperative object transportation // IEEE/RSJ Internat. Conf. on Intelligent Robots and Systems. Piscataway: IEEE press, 2004. Vol. 1. Pp. 1035–1040.

Статья поступила в редакцию 16.08.2017.

## REFERENCES

1. **Amelin K.S., Amelina N.O., Granichin O.N.** Adaptivnaya multiagentnaya operatsionnaya sistema realnogo vremeni dlya kompleksov BPLA [Adaptive multi-agent real-time operating system for UAV systems]. *Aktualnyye problemy Rossiyskoy kosmonavтики. Trudy XXXVIII akademicheskikh chteniy po kosmonavtike* [Actual problems of the Russian cosmonautics. Proceedings of the XXXVIII Academic Readings on Cosmonautics]. Moscow: Komissiya RAN, 2014, P. 654. (rus)

2. **Gorodetskiy V.I., Samoylov V.V., Trotskiy D.V.** Bazovaya ontologiya kolektivnogo povedeniya avtonomnykh agentov i yeye rasshireniya [The reference ontology of collective behavior of autonomous agents and its extensions]. *Izvestiya RAN. Teoriya i sistemy upravleniya* [Journal of Computer and Systems Sciences International], 2015, No. 5, Pp. 102–121. (rus)

3. **Yefremov A.Yu., Maksimov D.Yu.** Setetsentricheskaya sistema upravleniya – chto vkladывayetsya v eto ponyatiye? [The network-centric control system - what is included in this concept?] *Trudy III Vserossiyskoy konferentsii s mezhdunarodnym uchastiyem: Tekhnicheskkiye i programmnyye sredstva sistem upravleniya, kontrolya i izmereniya* [3rd All-Russian Conference with International Participation: Technical and Software Tools for Control, Monitoring and Measurement Systems]. Moscow: IPU RAN Publ., 2012, Pp. 158–161. (rus)

4. **Kalyayev I.A., Gayduk A.R., Kapustyan S.G.** Modeli i algoritmy kolektivnogo upravleniya v gruppakh robotov [Models and algorithms of collective management in groups of robots]. Moscow: Fizmatlit Publ., 2009, 280 p. (rus)

5. **Kalyayev I.A., Gayduk A.R., Kapustyan S.G.** Samoorganizatsiya v multiagentnykh sistemakh [Self-organization in multi-agent systems]. *Izvestiya SFedU. Engineering sciences*, 2010, Pp. 14–20. (rus)

6. **Karpova I.P.** Psevdoanalogovaya kommunikatsiya v gruppe robotov [Pseudo-analog communication in robot groups]. *Mechatronics*,

*Automation, Control*, 2016, No. 2(17), Pp. 94–101. (rus)

7. **Kuafe F.** *Vzaimodeystviye robota s vneshney sredoy* [Interaction of the robot with the external environment]. Moscow: Mir Publ., 1985, 285 p. (rus)

8. **Maltsev P.P., Ponomarev K.M., Stepanov Yu.I.** “Umnaya pyl” na osnove mikrosistemnoy tekhniki [“Clever dust” based on microsystem technology]. *Intellectual Robotics Systems 2001: Proceed. youth, scientific school*. Taganrog: TRTU Publ., 2001, Pp. 220–232. (rus)

9. **Onufriyev V.A., Udod A.S.** Besprovodnaya sistema kontrolya parametrov distantsionno raspredelennykh sooruzheniy [Wireless monitoring system for parameters of remotely distributed structures]. *Nauchnaya sessiya TUSUR-2010: Materialy dokladov Vserossiyskoy nauchno-tekhnicheskoy konferentsii studentov, aspirantov i molodykh uchenykh* [Scientific session of TUSUR-2010: Reports of the All-Russian Scientific and Technical Conference of Students, PhD Students and Young Scientists]. Tomsk: V-Spektr Publ., 2010, Part 2, Pp. 170–173. (rus)

10. **Amato C., Konidaris G.D., Cruz G., Maynor C.A., How J.P., Kaelbling L.P.** Planning for decentralized control of multiple robots under uncertainty. *Proceedings of IEEE International Conference on Robotics and Automation*, IEEE, 2015, Pp. 1214–1248.

11. **Blankenship J., Mishal S.** *Enhancing the Pololu 3pi with Robotbasic*. North Charleston SC: Createspace, 2010, 156 p.

12. **Brettel M., Friederichsen N., Keller M., Rosenberg M.** How virtualization, decentralization and network building change the manufacturing landscape: an industry 4.0 perspective. *International Journal of Mechanical, Aerospace, Industrial, Mechatronics and Manufacturing Engineering*, 2014, Vol. 8, No. 1, Pp. 37–44.

13. **Feddema J.T., Lewis C., Schoenwald D.A.** Decentralized control of cooperative robotic

vehicles: theory and application. *IEEE transactions on robotics and automation*, 2002, Vol. 18, No. 5, Pp. 852–864.

14. **Guilherme A.S., Kumar R.V., Campos M.F.M.** Decentralized algorithms for multirobot manipulation via caging. *Springer Tracts in Advanced Robotics*, 2003, Vol. 7, Pp. 257–274.

15. **Onufriev V. A., Sulerova A.S., Poteking V.V.** Cyber-physical systems application for the radio telescope's adaptive surface control task. *AST - Symposium on Automated Systems and Technologies*, Hannover: PZH Verlag, 2016, Pp. 51–56.

16. **Vásárhelyi G., Virágh Cs., Somorjai G., et al.** Outdoor flocking and formation flight with

autonomous aerial robots. *IROS 2014 Conference Digest - IEEE/RSJ International Conference on Intelligent Robots and Systems*, Chicago, 2014, Pp. 3866–3873.

17. **Wang Y., de Silva C.W.** A machine-learning approach to multi-robot coordination. *Engineering Applications of Artificial Intelligence*, 2008, Vol. 21(3), Pp. 470–484.

18. **Wang Z., Takano Y., Hirata Y., Kosuge K.** A pushing leader based decentralized control method for cooperative object transportation. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Piscataway: IEEE press, 2004, Vol. 1, Pp. 1035–1040.

*Received 16.08.2017.*

#### СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS

**ОНУФРИЕВ Вадим Александрович**

**ONUFRIEV Vadim A.**

E-mail: ovavadim@gmail.com

DOI: 10.18721/JCSTCS.10410

УДК 004.652.4

## РЕДАКТОР ВИЗУАЛЬНОГО ПРОЕКТИРОВАНИЯ СХЕМ И ПРОЦЕДУР ЗАПОЛНЕНИЯ OLAP-КУБОВ МНОГОМЕРНЫХ БАЗ ДАННЫХ

*С.Г. Попов, А.В. Коненков, Т.Н. Самочадина*

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

Разработка OLAP-кубов для использования в системах бизнес-анализа требует согласованных действий администратора баз данных и аналитика. Их взаимодействие, являясь итеративным процессом, замедляет процесс проектирования компоненты хранения и обобщения данных. Описан новый специализированный графический редактор, который позволит аналитику самостоятельно выполнить большую часть операций по созданию логической и физической схемы подсистемы хранения, ускорит процесс разработки системы. Сформулированы требования к графическому представлению схем OLAP и цепочек загрузки, а также к функциям редактора. Приведены примеры реализации интерфейса прототипа.

**Ключевые слова:** база данных; система управления базами данных; BI; аналитик; системный администратор; OLAP; ETL; редактор схем; графическая нотация.

**Ссылка при цитировании:** Попов С.Г., Коненков А.В., Самочадина Т.Н. Редактор визуального проектирования схем и процедур заполнения OLAP-кубов многомерных баз данных // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2017. Т. 10. № 4. С. 107–117. DOI: 10.18721/JCSTCS.10410

## VISUAL DESIGN EDITOR FOR SCHEMES AND PROCEDURES OF FILLING OLAP CUBES IN MULTIDIMENSIONAL DATABASES

*S.G. Popov, A.V. Konenkov, T.N. Samochadina*

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

The development of OLAP cubes for use in business analysis systems requires coordinated actions of the database administrator and analyst. Their interaction is an iterative process, and it slows down the design of the storage component and generalization of the data. We have developed a new analytic tool, which will allow to perform most of the operations to control a logical and physical scheme. It will speed up the process of storage subsystem development. This tool, described in this paper, is a specialized graphic editor. The paper formulates the requirements for graphical representation of OLAP cubes and ETL, requirements for editor functions and implementing the interface examples.

**Keywords:** database; database management system; BI; analyst; system administrator; OLAP; ETL; schematic editor; graphical notation.

**Citation:** Popov S.G., Konenkov A.V., Samochadina T.N. Visual design editor for schemes and procedures of filling OLAP cubes in multidimensional databases. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems, 2017, Vol. 10, No. 4, Pp. 107–117. DOI: 10.18721/JCSTCS.10410

## Введение

Особенности функционирования классических технологий создания многомерных баз данных для использования в системах бизнес-анализа, основанных на OLAP-решениях, предполагают одновременное участие как администратора данных, так и аналитика на этапах создания схемы куба и заполнения его данными. Администратор баз данных обеспечивает технологическое управление источниками данных и созданием куба, а аналитик задаёт семантическое наполнение создаваемой OLAP-системы. Реализуемый в этом случае классический подход обладает существенным недостатком: взаимодействие аналитика с администратором в процессе разработки OLAP-системы происходит в форме сходящихся ручных или автоматических итераций («создание описания куба» — «загрузка данных в куб» — «анализ результата»). При этом на каждом шаге попеременно участвуют администратор и аналитик.

Анализ средств автоматизации создания OLAP-кубов показал, что сокращение числа итераций для достижения приемлемого результата может быть обеспечено двумя путями: обеспечением автоматического формирования OLAP-схем в условиях известной или неизвестной схемы исходных наборов данных [10] и применением интеллектуальных редакторов данных для интеграции исходных метаданных в многомерные кубы.

Несмотря на активное развитие первого способа, состоящего в интеграции технологий OLAP и средств Data Mining [4–6], систематизации схем на основе принципов иерархической кластеризации [1, 2] и автоматизации генерации схемы кубов OLAP на основе концептуальных, в том числе и графических моделей [3, 11], представленные методы являются «обратными» и строят схему кубов OLAP по уже существующим данным и метаданным. В этом случае аналитик — потребитель автоматически сгенерированной схемы OLAP, что может снизить эффективность его последующей работы.

Альтернативный подход к процессу соз-

дания кубов — применение графического представления схемы куба и цепочек загрузки данных непосредственно в процессе его разработки. Такой подход является «прямым», управляется и контролируется аналитиком и обеспечивает прозрачность представления схемы. Наиболее распространённый способ задания управления метаданными — извлечение метаданных из исходных, в том числе и реляционных, подсистем и сохранение их в централизованном репозитории [13–15] с их последующей визуализацией. Процесс визуализации обеспечивает как визуальное представление данных из кубов в разрезах и выборках, основанное на запросах на графическом языке с генерацией куба [7], средствах 2,5D и 3D отображения данных, основанных на упорядочивании  $k$ -деревьев [8], так и отображения схем OLAP-хранилищ с одновременной поддержкой методов демонстрации данных [9, 12].

Особенностью отображения схемы OLAP-кубов и цепочек загрузки в подсистеме заполнения OLAP-кубов является требование к прозрачности представления исходных метаданных: источники данных, образующие схему куба, могут являться реляционными схемами, плоскими файлами или другими кубами. В этом случае вопрос представления единой схемы становится задачей интеграции метаданных разнородных баз данных [15].

Многомерность кубов и гетерогенность источников данных для заполнения куба являются ключевой особенностью задачи построения схем и цепочек запросов к кубам OLAP, что предъявляет дополнительные требования к средствам графического представления. Многомерность куба предполагает большое число связей в схемах «звезда» или «снежинка», что требует точного визуального представления атрибутов, участвующих в связях, и сокращения числа соединений между таблицами, при условии сохранения логики связей по ключам. Гетерогенность источников данных требует унификации изображения основных объектов общей схемы: таблиц и связей, выражающихся в едином стиле изображения объектов как полученных из разных ис-



точников, так и добавленных в схему при её редактировании. Эти требования выражаются в формальных и изобразительных ограничениях, налагаемых как на графическое средство, так и на процесс формирования изображения аналитиком.

### **Требования к графическому представлению схем и цепочек загрузки OLAP-кубов многомерных баз данных**

Современные визуальные редакторы схем могут быть ориентированы на конкретную РСУБД, поставляться её производителями или сообществом разработчиков, или одновременно поддерживать синтаксис языка манипулирования данными сразу нескольких СУБД и поставляться сторонними производителями.

К первому типу редакторов относятся Oracle Developer Data Modeler и Query Builder, MS SQL Visual Database Tools, IBM DB2 Control Center, PostgreSQL pgModeler. Их характеризует максимальное использование особенностей СУБД конкретного производителя, таких как максимальный учёт диалекта языка запросов, синтаксиса языка описания данных, поддержки языковых конструкций динамического SQL, прямого обращения к репозиторию метаданных. Такие редакторы не обеспечивают взаимодействие не только с несколькими СУБД различных производителей, но, нередко, и с несколькими базами данных, расположенными на одном или на нескольких серверах, что делает их применение невозможным для задачи построения объединённой схемы.

Второй тип редакторов представлен двумя подтипами продуктов, реализующими управление базами данных и выполняющими функции business intelligence. К первому подтипу относятся такие продукты, как DBeaver, MySQL Workbench, DbDesigner, DbSchema Diagram Designer. Их особенностью является возможность одновременного подключения к нескольким СУБД, однако создание общей схемы из нескольких источников в них невозможно. Вторым подтипом программных решений являются редакторы графических схем в BI продуктах, таких как Галактика BI, IBM Cognos

Analytics или Power BI Desktop. Эти редакторы схем обеспечивают интеграцию разнородных источников данных, однако являются готовыми коммерческими программными решениями с закрытым исходным кодом. Проведённый анализ решений показал, что коммерческие редакторы в полной мере не обеспечивают реализацию задачи комплексной интеграции схем баз данных при создании схем OLAP-кубов и цепочек загрузки.

Графические представления схем перечисленных программных продуктов основаны на вариациях представления графической нотации метамоделей IDEF1X\*. Формальное описание нотации не предполагает фиксированного визуального представления схемы, поэтому каждый разработчик редактора реализует представление в соответствии с частными техническими требованиями, выдвинутыми на этапе проектирования продукта. Рассматриваемые решения имеют ограниченную функциональность в части редактирования связей: они обеспечивают соединение атрибутов в автоматическом режиме, при этом линии соединений перекрываются таблицами, пересекаются друг с другом под произвольными углами, что создаёт трудности чтения схемы при большом числе таблиц и соединений.

Анализируя существующие решения изображения схем и особенности изображения схем OLAP-кубов и цепочек загрузки, в качестве базовых, были выдвинуты следующие требования:

- 1) фиксированного положения первичного простого или составного ключа;
- 2) указания типа всех ключей;
- 3) ручного изменения порядка следования атрибутов в таблице;
- 4) отображения связи ключей в разных таблицах стрелкой;
- 5) фиксации места начала и окончания связи (связь начинается около атрибута и заканчивается около атрибута);

---

\* IEEE. IEEE Standard for Conceptual Modeling Language Syntax and Semantics for IDEF1X: Std 1320.2-1998. IEEE, 1998. No. Std 1320.2-1998.

6) изображения ребра ломаной линией, состоящей из прямых частей;

7) исключения перекрытия изображения связи таблицей;

8) неперевышения шириной имени атрибута ширины таблицы;

9) возможности редактирования каждой связи вручную.

Предъявляемые требования к внешнему виду и функциональному наполнению редактора носят декларативный характер и не определяют критериев формирования изображения схемы: одну и ту же схему можно отобразить многими способами, причём некоторые из них будут лучше других с точки зрения визуального восприятия аналитиком. В этом случае качество отображения схемы формализуется с помощью таких понятий, как формальные требования и изобразительное соглашение.

#### **Критерии формирования графического представления схем и цепочек загрузки OLAP-кубов многомерных баз данных**

Анализ текущей практики показал, что к графическому изображению схемы OLAP-кубов могут быть выдвинуты два типа требований: формальные и неформальные. Формальные требования основаны на математических правилах, а неформальные — на основе изобразительных соглашений.

Особенностью формальных требований является их локальность: они применяются

к отдельным объектам или к их группам, а не ко всему изображению в целом, и общего критерия, совмещающего требования ко всему изображению, сформулировать невозможно. Поэтому общие требования выражаются в форме неформальных, изобразительных требований, «отвечающих» за организацию изображения схемы в целом.

**Формальные требования.** Формальные требования описывают такие свойства изображений, которые можно формально специфицировать и затем вычислить, чтобы повысить наглядность изображения. Они предъявляются к каждому объекту изображения, к совокупности объектов одного или нескольких типов, или ко всему изображению в целом. В таблице приведён перечень объектов изображения и формальные требования к ним.

В результате автоматического применения формальных критериев может быть получен локальный минимум по одному из критериев, однако оптимизация на основе целевой функции, составленной из композиции критериев, не может быть сформулирована в явной форме ввиду противоречия между требованиями. Другим недостатком оптимизационных методов построения изображения является высокая вычислительная сложность. Даже в случае корректного решения оптимизационной задачи результат может не удовлетворять аналитика в силу формально верного, но сложно вос-

#### **Формальные требования к изображению схемы**

##### **Formal requirements for schematic representation**

| Объект(ы)       | Минимизируемые параметры                                                                             |
|-----------------|------------------------------------------------------------------------------------------------------|
| Связь           | Длина одной связи                                                                                    |
| Связи           | Суммарная длина всех связей, число пересечений, разность между самой длинной и самой короткой связью |
| Изгиб связи     | Числа изгибов каждой линии, разность между минимальным и максимальным числом изгибов                 |
| Изгибы связей   | Минимизация числа различных углов изгибов линий                                                      |
| Таблица         | Минимизация площади, занимаемой таблицей                                                             |
| Таблицы         | Минимизация разности площадей таблиц                                                                 |
| Таблицы и связи | Минимизация общей площади изображения, приведение отношений сторон к формату листа                   |

принимаемого зрительно размещения.

**Изобразительные соглашения.** Графическое изображение схемы базы данных предназначено для использования аналитиками в повседневной практике. В этом случае требования к изображению могут формулироваться в виде требований к субъективным характеристикам, могут формулироваться на основе неформальных соглашений и являться общими правилами и критериями, сформулированными ко всему изображению схемы.

Изобразительные соглашения — это правила, определяющие общий стиль изображения элементов таблиц и связей между ними. Стиль формируется, исходя из требований к проектируемому редактору схем баз данных. Формулировка требований к изображению схем OLAP-кубов содержит требования к изображению таблиц, атрибутов в них, ключей и соединений между ними в части ручного управления связями схемы.

**Изображение таблиц.** Таблицы должны изображаться прямоугольными областями. Каждая таблица в своей верхней части должна содержать название, ниже следуют атрибуты, составляющие первичный ключ этой таблицы. Далее могут следовать прочие атрибуты таблицы с указанием их типа.

**Изображение атрибутов.** Расположение первичного ключа фиксировано, расположение прочих атрибутов может изменяться пользователем. При перемещении атрибутов перемещаются и все связи, начинающиеся или заканчивающиеся у этих атрибутов. Ширина таблицы не может быть меньше максимальной ширины имени атрибута, а высота определяется числом атрибутов.

**Изображение связей.** Логическая связь атрибутов таблиц изображается ребрами: каждое ребро начинается у названия атрибута одной таблицы и заканчивается у атрибута другой. Ребро начинается обычной линией, а заканчивается указателем — стрелкой, при этом для каждой связи фикс-



Рис. 1. Требования к графическому представлению схем

Fig.1 Schematic diagram representation requirements

сируется её тип «один к одному» или «один ко многим». Атрибут, являющийся первичным ключом в связи «один ко многим», отмечается единицей. Атрибут, являющийся внешним ключом в связи «один ко многим», отмечается символом бесконечности.

Физически атрибут одной таблицы может одновременно быть связан сразу с несколькими атрибутами другой таблицы. В этом случае ребро разветвляется так, что связь имеет общую часть и точку разветвления.

Ребро может иметь неограниченное число изгибов, которые добавляются пользователем, в основном, для исключения пересечений с таблицами и другими рёбрами.

Предложенный набор требований приведён на рис. 1. Требования обеспечивают расширенный перечень функций ручного управления для непротиворечивого отображения схемы и цепочек загрузки OLAP-куба многомерной базы данных.

### Модель представления логического и физического уровней схемы OLAP-куба

Для реализации представления схемы вводятся два уровня описания связи: логический, определяющий схему связи между таблицами по ключам, и физический – уровень отображения, который будет отражать внешний вид соединений на схеме.

Тогда логическая схема куба и цепочек загрузки задаётся графом  $G = (E, V)$ , где вершинами являются структуры, описывающие таблицы схемы  $E = Dim(A_i, Key_i, Dim(B_{Ai}))$ , где  $A_i$  – имена таблиц,  $Key_i$  – ключевые атрибуты таблицы  $A_i$ ,  $B_{Ai}$  – перечень внешних ключей таблицы,  $i = (1, 2, \dots, n)$  – число таблиц в схеме,  $j = (1, 2, \dots, k)$  – число внешних ключей каждой таблицы. Набор ребер  $V$  задаётся матрицей смежности на наборе первичных и внешних ключей  $Key_i$  и массиве  $Dim(B_{Ai})$ .

Логический уровень обеспечивает представление семантики схемы и напрямую отражается в схемы хранилища метаданных – репозитория или *information\_schema*, а физический – задаёт визуальное представление отображения элементов и сохраняется в редакторе схем.

На логическом уровне каждому ребру  $v_i$  множества рёбер  $V$  графа  $G$  ставится в соответствие граф физических связей  $L_i = \{\{l_1, \dots, l_m\}, \{l_1, \dots, l_i\}, \dots, \{l_p, \dots, l_s\}\}$ , отображающий совокупность всех отрезков и всех связей одного атрибута таблицы.

Иллюстрация предложенного подхода приведена на рис. 2 и 3, где изображены примеры представления графов для логического и физического уровней соответственно. Для реализации логического уровня представления схемы куба применяется схематичное изображение связей. Физи-

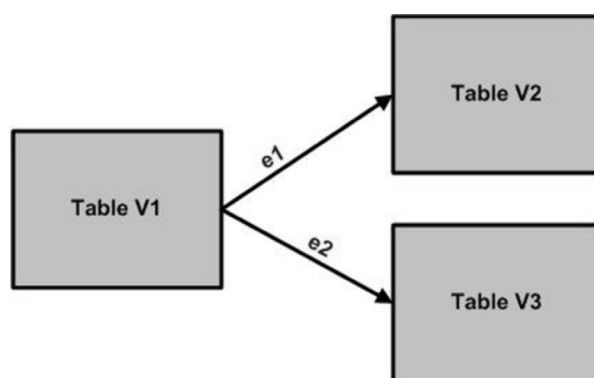


Рис. 2. Пример графического представления логического уровня представления связи

Fig. 2. Example of graphical representation of logical level of communication representation

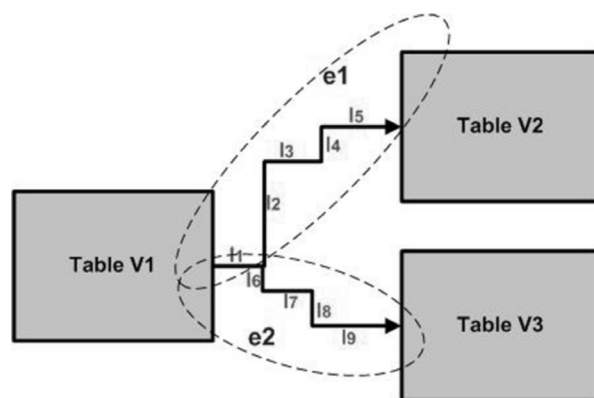


Рис. 3. Пример графического представления физического уровня представления связи

Fig. 3. Example of graphical representation of the physical level of communication representation

ческое представление позволяет изменять графическое представление связи независимо от семантики схемы.

Реализация предложенной двухуровневой модели обеспечивает независимость логической структуры схемы от физического представления и реализована в редакторе схем OLAP-кубов.

### Реализация редактора графических представления схем OLAP-кубов

В результате анализа требований к графическому представлению и, основываясь на существующих средствах графического отображения, сформулированы функции графического редактора схем. В нём определены два типа сущностей: таблицы и связи. Для каждого из них выделены функции, позволяющие обеспечить ручное управление схемами.

**Управление таблицами.** Функция обеспечивает формирование, изменение и удаление таблицы, редактирование заголовка таблицы, названий атрибутов и

их типов, удаление атрибутов, изменение порядка следования атрибутов с учётом перемещения связей. Реализация интерфейса управления таблицами обеспечивает функционально полное управление наполнением атрибутами схемы OLAP-куба, такими как размерности и словари. Пример интерфейса управления таблицей приведён на рис. 4.

**Управление связями.** Обеспечивает управление изображением связей между таблицами в части создания, удаления и изменения типа связи. Управление связями осуществляется путём перемещения точек разветвления линий. В этом случае каждое ребро изображается в виде ломаной линии, состоящей из чередующихся вертикальных и горизонтальных сегментов. На первом этапе описание логической связи извлекается из репозитория метаданных и становится основой физической связи. Реализован и обратный процесс: графическое формирование связи обеспечивает её сохранение в репозитории метаданных.

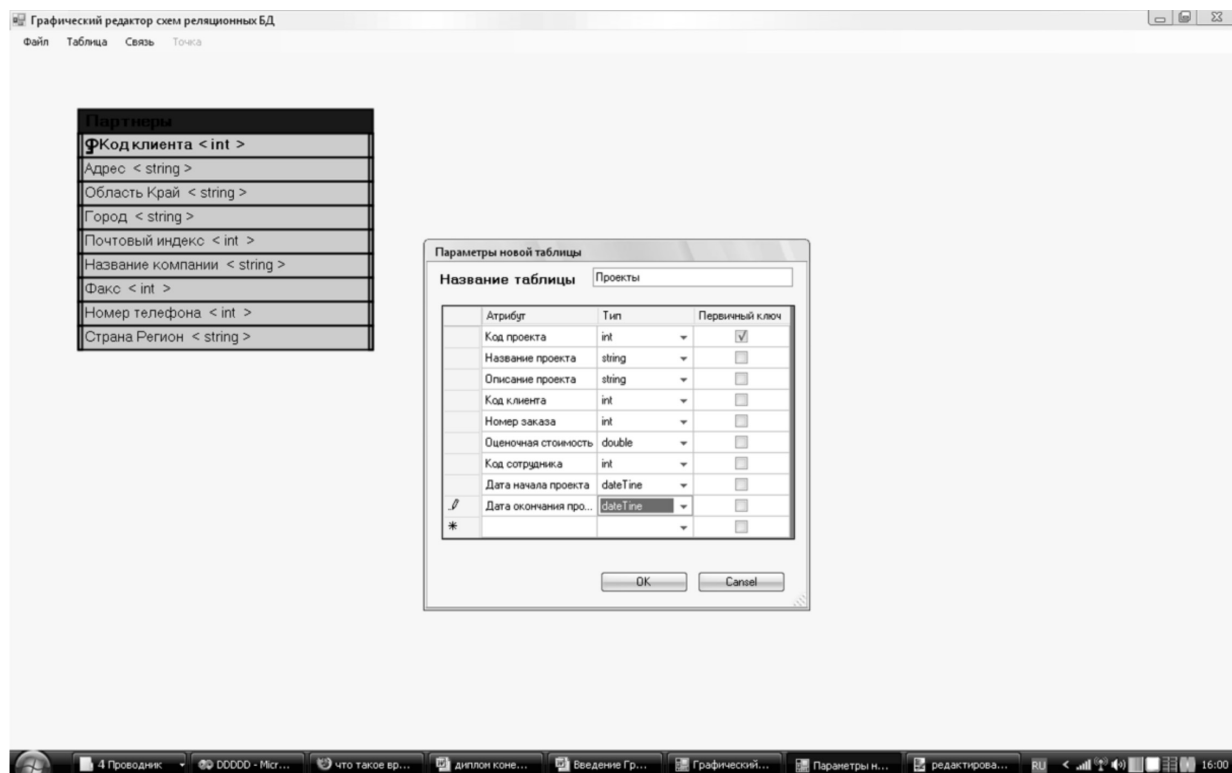


Рис. 4. Пример графического представления управления таблицей

Fig. 4. Example of graphical representation of table management



**Управление физическим отображением связей.** Обеспечивается управлением частями существующей связи путём управления добавляемыми точками ветвления. На каждую часть связи можно добавить и удалить произвольное число точек ветвления. Каждую точку ветвления можно перемещать по части связи, что обеспечивает излом линии в произвольных местах. При необходимости редактирование точки обеспечивает объединение и разделение связей, что обеспечивает организацию физического представления соединения. Нотация обеспечивает визуальное представление связей между частями OLAP-куба в вариантах изображения как «звезда» и «снежинка», так и схем произвольной формы. Внешний вид схемы OLAP-куба в представленной нотации приведён на рис. 5.

Сформулированный набор операций над сущностями обеспечивает функцио-

нально полное управление объектами из пользовательского интерфейса.

Прототип редактора реализован в виде компилированного приложения на языке программирования C# на платформе .Net с использованием технологии Windows Forms для операционной системы Windows 7.0. Приложение содержит модули взаимодействия с репозиторием метаданных, управления сохранением схем кубов и графического интерфейса пользователя. Модуль взаимодействия с репозиторием метаданных обеспечивает считывание и запись унифицированных описаний таблиц и описаний логических связей между ними. Модуль управления схемами кубов обеспечивает сохранение данных об относительном расположении таблиц, частей связей на экране. Модуль графического интерфейса обеспечивает отрисовку элементов на экране и вызов экранных форм для управ-

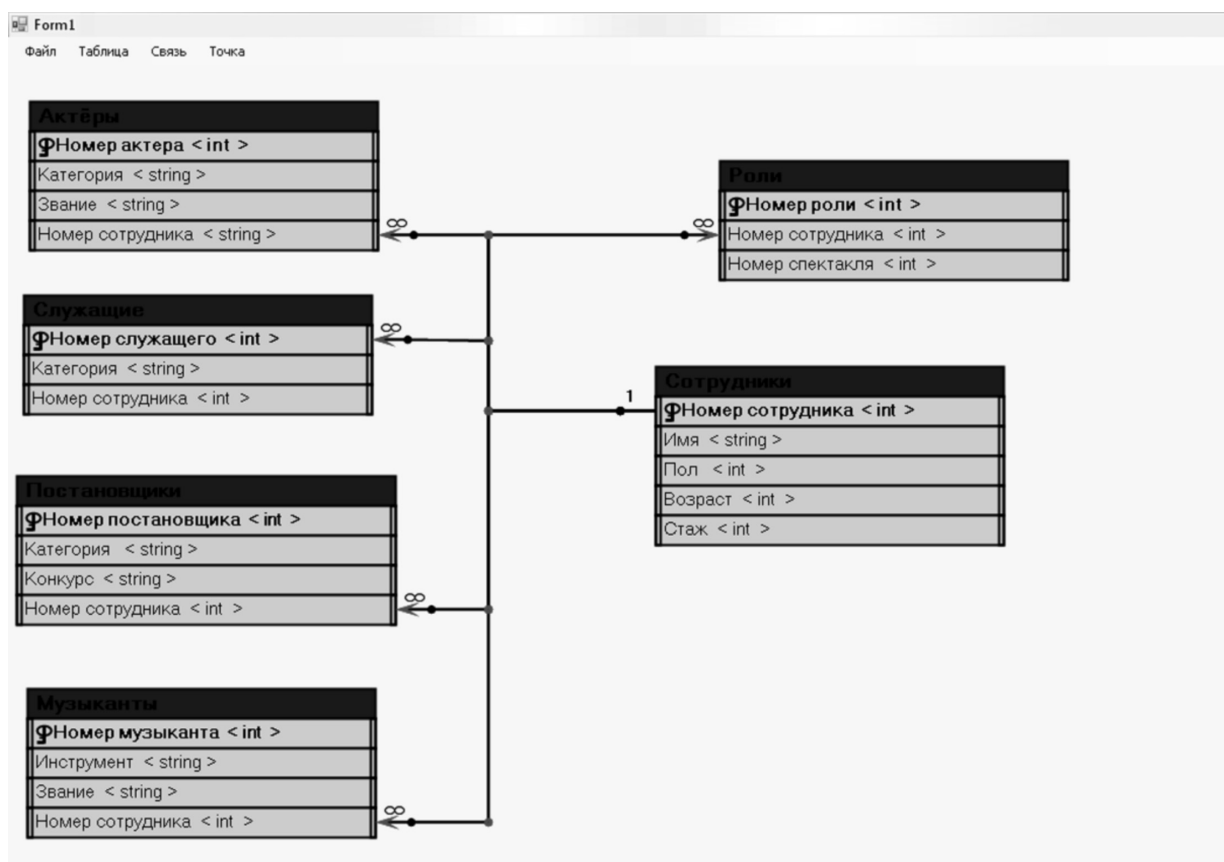


Рис. 5. Пример графического представления схемы OLAP-куба

Fig. 5. Example of graphical representation of the OLAP cube schema

ления изображением. Прототип редактора использовался в качестве средства редактирования схем OLAP-кубов на этапе эскизного проектирования программно-технологической платформы интерактивного стратегирования и бизнес-анализа с элементами прогнозирования для государственных и коммерческих организаций.

### Заключение

Проанализированы способы визуального проектирования схем многомерных баз данных, выделены формальные и изобразительные требования к графическому редактору, сформулирован перечень функций программного обеспечения, спроектирован и реализован прототип графического редактора схем реляционных баз данных.

Для реализации редактора предложена адаптивная динамическая двухуровневая графовая модель представления связей схемы базы данных, обеспечивающая ручное управление изображением связей. На верхнем уровне граф описывает логические связи между таблицами, а на нижнем совокупность связей атрибута отдельной таблицы физически задаётся уточняющим графом.

На этапе эскизного проекта реализован прототип графического редактора схем

OLAP-кубов многомерной базы. Прототип обеспечивает загрузку схем данных из гетерогенных источников, обеспечивая унификацию графического представления таблиц, атрибутов и связей; реализует отображение многомерных кубов в форме реляционной схемы данных с возможностью ручного управления числом и положением соединений в схеме и размещением реляционных таблиц куба; конкретизацию соединений по ключам между таблицами до имени атрибута, что гарантирует тонкую ручную настройку внешнего вида изображения. Указанная совокупность особенностей отражает специфику построения схем OLAP-кубов и цепочек загрузки. Применение прототипа редактора показало реализуемость данного подхода для проектирования схем OLAP-кубов. Предложенные в редакторе решения могут использоваться на этапах технического проектирования программно-технологической платформы интерактивного стратегирования.

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

### СПИСОК ЛИТЕРАТУРЫ

1. Wang X., Yi B. The Application of Data Cubes in Business Data Visualization // *Computing in Science & Engineering*. 2012. Vol. 14. No. 6. Pp. 44–50.
2. Макарова Е.С., Миронов В.В. Проектирование концептуальной модели данных для задач Web-OLAP на основе ситуационно-ориентированной базы данных // *Вестник УГАТУ*. 2012. Т. 16. № 6 (51). С. 177–188.
3. Скворцова Д.А., Чмырь Д.А. Использование многомерных OLAP-кубов, как инструмента Business Intelligence, при стратегическом управлении бизнес-процессами компании // *Экономика: теория и практика*. 2016. № 3. С. 70–77.
4. Едуш М.В., Витиска Н.И. Исследование проблемы поиска центральной таблицы для многомерного куба (или групп таблиц) в реляционной БД // *Информатизация и связь*. 2013. № 3. С. 158–161.
5. 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.
6. Кобец Д.А., Балашов И.В., Данилов И.Д., Лупян Е.А., Сычугов И.Г., Толопин В.А. Использование BI-технологий для создания инструментов для анализа данных спутникового мониторинга // *Современные проблемы дистанционного зондирования земли из космоса*. 2015. № 4. С. 17–27.
7. Djiroun R., Boukhalfa K., Alimazighi Z., Atigui F., Bimonte S. A data cube design and construction methodology based on OLAP queries // *IEEE/ACS 13th Internat. Conf. of Computer Systems and Applications*. Agadir, 2016. Pp. 1–8.
8. Lafon S., Bouali F., Guinot C., Venturini G. Hierarchical reorganization of dimensions in OLAP visualizations // *IEEE Transactions on Visualization*.

tion and Computer Graphics. 2013. Vol. 19. No. 11. Pp. 1833–1845.

9. **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.

10. **Usman M., Asghar S., Fong S.** Data mining and automatic OLAP schema generation // *5th Internat. Conf. on Digital Information Management*. Thunder Bay, ON, 2010. Pp. 35–43.

11. **Nabli A., Feki J., Gargouri F.** Automatic construction of multidimensional schema from OLAP requirements // *The 3rd ACS/IEEE Internat. Conf. on Computer Systems and Applications*. Cairo, 2005. Pp. 28–24.

12. **Kovačević I., Mekterović I.** Alternative business intelligence engines // *40th International Con-*

*vention on Information and Communication Technology, Electronics and Microelectronics*. Opatija, 2017. Pp. 1385–1390.

13. **Ristić S., Kordić S., Čeliković M., Dimitrieski V., Luković I.** A model-to-model transformation of a generic relational database schema into a form type data model // *Federated Conf. on Computer Science and Information Systems*. Gdansk, 2016. Pp. 1577–1580.

14. **Malki M., Flory A., Rahmouni M.K.** Extraction of object-oriented schemas from existing relational databases: A form-driven approach // *Informatica*. 2002. Vol. 13(1). Pp. 47–72.

15. **Попов С.Г.** Методика построения оптимального репозитория схем реляционных баз данных // *Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление*. 2010. № 5 (108). С. 91–97.

*Статья поступила в редакцию 14.09.2017.*

## REFERENCES

1. **Wang X., Yi B.** The Application of Data Cubes in Business Data Visualization. *Computing in Science & Engineering*, 2012, Vol. 14, No. 6, Pp. 44–50.

2. **Makarova Ye.S., Mironov V.V.** Designing a conceptual data model for Web-OLAP tasks based on a situationally-oriented database. *Newsletter UGATU*, 2012, Vol. 16, No. 6 (51), Pp. 177–188. (rus)

3. **Skvortsova D.A., Chmyr D.A.** Using multidimensional OLAP cubes as a tool of Business Intelligence, in the strategic management of the company's business processes. *Economics: theory and practice*, 2016, No. 3, Pp. 70–77. (rus)

4. **Yedush M.V., Vitiska N.I.** The research of the central table search problem for a multidimensional cube (or groups of tables) in a relational database. *Informatization and communication*, 2013, No. 3, Pp. 158–161. (rus)

5. **Naumov V.N., Lychagina Ye.B., Sharababayeva L.Yu.** Using BI-systems to provide information and analytical activities of public authorities. *Management Consulting*, 2016, No. 3(87), Pp. 144–153. (rus)

6. **Kobets D.A., Balashov I.V., Danilov I.D., Lupyan Ye.A., Sychugov I.G., Tolopin V.A.** Using BI-technologies to create tools for analyzing satellite monitoring data. *Modern problems of remote sensing of earth from space*, 2015, No. 4, Pp. 17–27. (rus)

7. **Djiroun R., Boukhalfa K., Alimazighi Z., Atigui F., Bimonte S.** A data cube design and construction methodology based on OLAP queries. *2016 IEEE/ACS 13th International Conference of Computer Systems and Applications*, Agadir, 2016, Pp. 1–8.

8. **Lafon S., Bouali F., Guinot C., Venturini G.** Hierarchical reorganization of dimensions in OLAP visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 2013, Vol. 19, No. 11, Pp. 1833–1845.

9. **Bhowmik T., Sarkar A., Debnath N.C.** OLAP umbrella: Visualization model for multidimensional databases. *ACS/IEEE International Conference on Computer Systems and Applications*, Hammamet, 2010, Pp. 1–8.

10. **Usman M., Asghar S., Fong S.** Data mining and automatic OLAP schema generation. *5th International Conference on Digital Information Management*, Thunder Bay, ON, 2010, Pp. 35–43.

11. **Nabli A., Feki J., Gargouri F.** Automatic construction of multidimensional schema from OLAP requirements. *The 3rd ACS/IEEE International Conference on Computer Systems and Applications*, Cairo, 2005, Pp. 28–24.

12. **Kovačević I., Mekterović I.** Alternative business intelligence engines. *40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, Opatija, 2017, Pp. 1385–1390.

13. **Ristić S., Kordić S., Čeliković M., Dimitrieski V., Luković I.** A model-to-model transformation of a generic relational database schema into a form type data model. *Federated Conference on Computer Science and Information Systems (FedCSIS)*, Gdansk, 2016, Pp. 1577–1580.

14. **Malki M., Flory A., Rahmouni M.K.**

Extraction of object-oriented schemas from existing relational databases: A form-driven approach. *Informatica*, 2002, Vol. 13(1), Pp. 47–72.

15. **Попов S.G.** Method of construction of

optimal repository of relational schemas databases. *St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunication and Control Systems*, 2010, No. 5 (108), Pp. 91–97. (rus)

*Received 14.09.2017.*

#### **СВЕДЕНИЯ ОБ АВТОРАХ / THE AUTHORS**

**ПОПОВ Сергей Геннадьевич**

**POPOV Sergey G.**

E-mail: popovserge@spbstu.ru

**КОНЕНКОВ Андрей Викторович**

**KONENKOV Andrey V.**

E-mail: konenkoff@mail.ru

**САМОЧАДИНА Татьяна Николаевна**

**SAMOCHADINA Tatiana N.**

E-mail: tatiana.samochadina@gmail.com

**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 10, № 4, 2017**

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](mailto:infocom@spbstu.ru)

Компьютерная верстка *А.Н. Смирнов*

Перевод на английский язык *А.С. Колгатина*

Лицензия ЛР № 020593 от 07.08.97

---

Подписано в печать 29.12.2017. Формат 60×84 1/8. Бум. тип. № 1.  
Печать офсетная. Усл. печ. л. 13,95. Уч.-изд. л. 13,95. 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 10<sup>00</sup> до 18<sup>00</sup> Галина Александровна**

**или по e-mail: [infocom@spbstu.ru](mailto:infocom@spbstu.ru)**