

# Программные продукты и системы

DOI: 10.15827/0236-235X

/ Исследование  
свойств  
конкурентного  
двоичного дерева

/ Разработка чат-бота  
для отслеживания  
расписания учебных  
занятий

/ Об архитектуре  
системы поддержки  
принятия решений

/ Программный  
комплекс  
планирования  
производства



№ 3

ТОМ 36  
2023



---

Научно-исследовательский институт  
«Центрпрограммсистем»

---

# Программные продукты и системы

НАУЧНЫЙ ЖУРНАЛ

2023, том 36, № 3  
(год издания тридцать шестой)

**Главный редактор**  
Г.И. САВИН, академик РАН

## SOFTWARE & SYSTEMS

Research Journal

2023, vol. 36, no. 3

**Editor-in-Chief**  
G.I. SAVIN, Academician of the Russian Academy of Sciences

---

Research Institute CENTERPROGRAMSYSTEM

---

## © ПРОГРАММНЫЕ ПРОДУКТЫ И СИСТЕМЫ

Научный журнал

2023. Т. 36. № 3

DOI: 10.15827/0236-235X.142

### Главный редактор

**Г.И. САВИН**, академик РАН

### Научные редакторы номера:

**Н.А. СЕМЕНОВ**, д.т.н., профессор

**А.Н. СОТНИКОВ**, д.ф.-м.н., профессор

Издатель НИИ «Центрпрограммсистем»

(г. Тверь, Россия)

Учредитель **В.П. Куприянов**

Журнал зарегистрирован в Роскомнадзоре

3 марта 2020 г.

Регистрационное свидетельство ПИ № ФС 77-77843

Подписной индекс в каталоге

Урал-Пресс **70799**

ISSN 0236-235X (печатн.)

ISSN 2311-2735 (онлайн)

## РЕДАКЦИОННАЯ КОЛЛЕГИЯ

**Семенов Н.А.** – заместитель главного редактора, д.т.н., профессор Тверского государственного технического университета (г. Тверь, Россия)

**Сотников А.Н.** – заместитель главного редактора, д.ф.-м.н., профессор, заместитель директора Межведомственного суперкомпьютерного центра РАН (г. Москва, Россия)

**Афанасьев А.П.** – д.ф.-м.н., профессор Московского физико-технического института (технического университета), заведующий Центром распределенных вычислений Института проблем передачи информации РАН (г. Москва, Россия)

**Баламетов А.Б.** – д.т.н., профессор Азербайджанского научно-исследовательского и проектно-испытательского института энергетики (г. Баку, Азербайджан)

**Батыршин И.З.** – д.т.н., профессор Мексиканского института нефти (г. Мехико, Мексика)

**Борисов В.В.** – д.т.н., профессор филиала Национального исследовательского университета «МЭИ» в г. Смоленске (г. Смоленск, Россия)

**Голенков В.В.** – д.т.н., профессор Белорусского государственного университета информатики и радиоэлектроники (г. Минск, Беларусь)

**Елизаров А.М.** – д.ф.-м.н., профессор Института математики и механики им. Н.И. Лобачевского Казанского федерального университета (г. Казань, Россия)

**Еремеев А.П.** – д.т.н., профессор Национального исследовательского университета «МЭИ» (г. Москва, Россия)

**Кузнецов О.П.** – д.т.н., профессор Института проблем управления РАН (г. Москва, Россия)

**Курейчик В.М.** – д.т.н., профессор Инженерно-технологической академии Южного федерального университета (г. Таганрог, Россия)

**Лисецкий Ю.М.** – д.т.н., генеральный директор «S&T Ukraine» (г. Киев, Украина)

**Мамросенко К.А.** – к.т.н., доцент Московского авиационного института (национального исследовательского университета), руководитель Центра визуализации и спутниковых информационных технологий НИИСИ РАН (г. Москва, Россия)

**Мейер Б.** – доктор наук, профессор, заведующий кафедрой Высшей политехнической школы – ЕТН (г. Цюрих, Швейцария)

**Палюх Б.В.** – д.т.н., профессор Тверского государственного технического университета (г. Тверь, Россия)

**Серов В.С.** – д.ф.-м.н., профессор Университета прикладных наук Оулу (г. Оулу, Финляндия)

**Сулейманов Д.Ш.** – академик АН Республики Татарстан, д.т.н., профессор Казанского государственного технического университета (г. Казань, Республика Татарстан, Россия)

**Татарникова Т.М.** – д.т.н., доцент, профессор Санкт-Петербургского государственного электротехнического университета «ЛЭТИ» им. В.И. Ульянова (Ленина) (г. Санкт-Петербург, Россия)

**Ульянов С.В.** – д.ф.-м.н., профессор, ведущий научный сотрудник Объединенного института ядерных исследований (г. Дубна, Россия)

**Хорошевский В.Ф.** – д.т.н., профессор Московского физико-технического института (технического университета) (г. Москва, Россия)

**Шабанов Б.М.** – д.т.н., чл.-корр. РАН, директор Межведомственного суперкомпьютерного центра РАН (г. Москва, Россия)

**Язенин А.В.** – д.ф.-м.н., профессор Тверского государственного университета (г. Тверь, Россия)

## АССОЦИИРОВАННЫЕ ЧЛЕНЫ РЕДАКЦИИ

*Национальный исследовательский университет «МЭИ», г. Москва, Россия  
Технологический институт Южного федерального университета, г. Таганрог, Россия  
Тверской государственный технический университет, г. Тверь, Россия*

### АДРЕС ИЗДАТЕЛЯ И РЕДАКЦИИ

Россия, 170024,

г. Тверь, просп. Николая Корыткова, д. 3а

Телефон (482-2) 39-91-49

Факс (482-2) 39-91-00

E-mail: [red@cps.tver.ru](mailto:red@cps.tver.ru)

Сайт: [www.swsys.ru](http://www.swsys.ru)

Дата выхода в свет 16.09.2023 г.

Отпечатано ИПШ «Фактор и К»

Россия, 170100, г. Тверь, ул. Крылова, д. 26

Выпускается один раз в квартал

Год издания тридцать шестой

Формат 60×84 1/8. Объем 160 стр.

Заказ № 15. Тираж 1000 экз. Цена 550,00 руб

© SOFTWARE & SYSTEMS

Research journal

2023, vol. 36, no. 3

DOI: 10.15827/0236-235X.142

**Editor-in-chief**

**G.I. SAVIN**, Academician of RAS

**Science editors of the issue:**

**N.A. Semenov**, Dr.Sc. (Engineering), Professor

**A.N. Sotnikov**, Dr.Sc. (Physics and Mathematics), Professor

**Publisher Research Institute**

**CENTERPROGRAMSYSTEM** (Tver, Russian Federation)

**Founder V.P. Kupriyanov**

The journal is registered with the Federal Service  
for Supervision of Communications, Information Technology  
and Mass Communications (Roskomnadzor)  
March 3<sup>rd</sup>, 2020

Registration certificate IIII № ФС 77-77843

**ISSN 0236-235X** (print)

**ISSN 2311-2735** (online)

## EDITORIAL BOARD

**Semenov N.A.** – Deputy Editor-in-Chief, Dr.Sc. (Engineering), Professor of the Tver State Technical University (Tver, Russian Federation)

**Sotnikov A.N.** – Deputy Editor-in-Chief, Dr.Sc. (Physics and Mathematics), Professor, Deputy Director of the Joint Supercomputer Center of the Russian Academy of Sciences (Moscow, Russian Federation)

**Afanasiev A.P.** – Dr.Sc. (Physics and Mathematics), Professor of the Moscow Institute of Physics and Technology, Head of Centre for Distributed Computing of Institute for Information Transmission Problems (Moscow, Russian Federation)

**Balametov A.B.** – Dr.Sc. (Engineering), Professor of the Azerbaijan Scientific-Research & Design-Prospecting Power Engineering Institute (Baku, Azerbaijan)

**Batyrshin I.Z.** – Dr.Sc. (Engineering), Professor of the Mexican Petroleum Institute (Mexico City, Mexico)

**Borisov V.V.** – Dr.Sc. (Engineering), Professor of the MPEI Branch in Smolensk (Smolensk, Russian Federation)

**Golenkov V.V.** – Dr.Sc. (Engineering), Professor of the Belarusian State University of Informatics and Radioelectronics (Minsk, Republic of Belarus)

**Elizarov A.M.** – Dr.Sc. (Physics and Mathematics), Professor of the N.I. Lobachevsky Institute of Mathematics and Mechanics of the Kazan Federal University (Kazan, Russian Federation)

**Eremeev A.P.** – Dr.Sc. (Engineering), Professor of the National Research University "Moscow Power Engineering Institute" (Moscow, Russian Federation)

**Kuznetsov O.P.** – Dr.Sc. (Engineering), Professor of the Institute of Control Sciences of the Russian Academy of Sciences (Moscow, Russian Federation)

**Kureichik V.M.** – Dr.Sc. (Engineering), Professor of the Academy of Engineering and Technology of the Southern Federal University (Taganrog, Russian Federation)

**Lisetsky Yu.M.** – Dr.Sc. (Engineering), CEO of "S&T Ukraine" (Kiev, Ukraine)

**Mamrosenko K.A.** – Ph.D. (Engineering), Associate Professor of the Moscow Aviation Institute (National Research University), Head of the Center of Visualization and Satellite Information Technologies SRISA RAS (Moscow, Russian Federation)

**Meyer B.** – Dr.Sc., Professor, Head of Department in the Swiss Federal Institute of Technology in Zurich, ETH (Zurich, Switzerland)

**Palyukh B.V.** – Dr.Sc. (Engineering), Professor of the Tver State Technical University (Tver, Russian Federation)

**Serov V.S.** – Dr.Sc. (Physics and Mathematics), Professor of the Oulu University of Applied Sciences (Oulu, Finland)

**Suleimanov D.Sh.** – Academician of TAS, Dr.Sc. (Engineering), Professor of the Kazan State Technical University (Kazan, Republic of Tatarstan, Russian Federation)

**Tatarnikova T.M.** – Dr.Sc. (Engineering), Associate Professor, Professor of the St. Petersburg Electrotechnical University "LETI" (St. Petersburg, Russian Federation)

**Ulyanov S.V.** – Dr.Sc. (Physics and Mathematics), Professor of the Dubna International University for Nature, Society and Man (Dubna, Russian Federation)

**Khoroshevsky V.F.** – Dr.Sc. (Engineering), Professor of the Moscow Institute of Physics and Technology (Moscow, Russian Federation)

**Shabanov B.M.** – Dr.Sc. (Engineering), Corresponding Member of the RAS, Director of the Joint Supercomputer Center of the Russian Academy of Sciences (Moscow, Russian Federation)

**Yazenin A.V.** – Dr.Sc. (Physics and Mathematics), Professor of the Tver State University (Tver, Russian Federation)

## ASSOCIATED EDITORIAL BOARD MEMBERS

*National Research University "Moscow Power Engineering Institute", Moscow, Russian Federation  
Technology Institute at Southern Federal University, Taganrog, Russian Federation  
Tver State Technical University, Tver, Russian Federation*

### EDITORIAL BOARD AND PUBLISHER OFFICE ADDRESS

Nikolay Korytkov Ave, 3a, Tver, 170024, Russian Federation

Phone: (482-2) 39-91-49 Fax: (482-2) 39-91-00

E-mail: red@cps.tver.ru

Website: www.swsys.ru

Release date 16.09.2023

Printed in printing-office "Faktor i K"

Krylova St. 26, Tver, 170100, Russian Federation

Published quarterly. 36th year of publication

Format 60×84 1/8. Circulation 1000 copies

Prod. order № 15. Wordage 160 pages. Price 550,00 rub.

## Вниманию авторов

Журнал «Программные продукты и системы» публикует материалы научного и научно-практического характера по новым информационным технологиям, результаты академических и отраслевых исследований в области использования средств вычислительной техники. Практикуются выпуски тематических номеров по искусственному интеллекту, системам автоматизированного проектирования, по технологиям разработки программных средств и системам защиты, а также специализированные выпуски, посвященные научным исследованиям и разработкам отдельных вузов, НИИ, научных организаций.

Журнал «Программные продукты и системы» внесен в Перечень ведущих рецензируемых научных журналов и изданий, в которых должны быть опубликованы основные научные результаты диссертаций на соискание ученых степеней кандидата и доктора наук.

Информация об опубликованных статьях по установленной форме регулярно предоставляется в систему РИНЦ, в CrossRef и в другие базы и электронные библиотеки.

Журнал «Программные продукты и системы» включен в ядро коллекции РИНЦ, размещенное на платформе Web of Science в виде базы данных RSCI.

Автор статьи отвечает за подбор, оригинальность и точность приводимого фактического материала. При перепечатке ссылка на журнал обязательна. Статьи публикуются бесплатно.

### Условия публикации

К рассмотрению принимаются оригинальные материалы, отвечающие редакционным требованиям и соответствующие тематике журнала. Группы научных специальностей:

#### **1.2. Компьютерные науки и информатика**

1.2.1. Искусственный интеллект и машинное обучение (физико-математические науки).

1.2.2. Математическое моделирование, численные методы и комплексы программ (физико-математические науки, технические науки)

#### **2.3. Информационные технологии и телекоммуникации**

2.3.1. Системный анализ, управление и обработка информации, статистика (технические науки, физико-математические науки).

2.3.2. Вычислительные системы и их элементы (технические науки).

2.3.3. Автоматизация и управление технологическими процессами и производствами (технические науки).

2.3.5. Математическое и программное обеспечение вычислительных систем, комплексов и компьютерных сетей (технические науки, физико-математические науки).

2.3.6. Методы и системы защиты информации (технические науки, физико-математические науки).

2.3.7. Компьютерное моделирование и автоматизация (технические науки, физико-математические науки).

2.3.8. Информатика и информационные процессы (технические науки).

Работа представляется в электронном виде в формате Word. Объем статьи вместе с иллюстрациями – не менее 10 000 знаков. Диаграммы, схемы, графики должны быть доступными для редактирования (Word, Visio, Excel). Заголовок должен быть информативным; сокращения, а также терминологию узкой тематики желательно в нем не использовать. Количество авторов на одну статью – не более 4, количество статей одного автора в номере, включая соавторство, – не более 2. Список литературы, наличие которого обязательно, должен включать не менее 10 пунктов.

Необходимы также содержательная структурированная аннотация (не менее 250 слов), ключевые слова (7–10) и индекс УДК. Название статьи, аннотация и ключевые слова должны быть переведены на английский язык (машинный перевод недопустим), а фамилии авторов, названия и юридические адреса организаций (если нет официального перевода) – транслитерированы по стандарту BGN/PCGN.

Вместе со статьей следует прислать экспертное заключение о возможности открытого опубликования материала и авторскую справку. Обязательно соблюдение автором договора (публичной оферты).

### Порядок рецензирования

Все статьи, поступающие в редакцию (соответствующие тематике и оформленные согласно требованиям к публикации), подлежат двойному слепому рецензированию в течение месяца с момента поступления, рецензия отправляется авторам.

В редакции сформирован устоявшийся коллектив рецензентов, среди которых члены редколлегии журнала, эксперты из числа крупных специалистов в области информатики и вычислительной техники ведущих вузов страны, а также ученые и специалисты НИИСИ РАН, МСЦ РАН (г. Москва) и НИИ «Центрпрограмм-систем» (г. Тверь).

Редакция журнала «Программные продукты и системы» в своей работе руководствуется сводом правил Кодекса этики научных публикаций, разработанным и утвержденным Комитетом по этике научных публикаций (Committee on Publication Ethics – COPE).

## Построение совершенных нормальных форм булевых функций для схемотехнических реализаций протоколов аутентификации с использованием Maple

А.А. Оленев  
И.А. Калмыков  
К.А. Киричек

### Ссылка для цитирования

Оленев А.А., Калмыков И.А., Киричек К.А. Построение совершенных нормальных форм булевых функций для схемотехнических реализаций протоколов аутентификации с использованием Maple // Программные продукты и системы. 2023. Т. 36. № 3. С. 349–360. doi: 10.15827/0236-235X.142.349-360

### Информация о статье

Поступила в редакцию: 15.03.2023

После доработки: 02.05.2023

Принята к публикации: 17.05.2023

**Аннотация.** Одной из перспективных областей применения дискретной математики являются протоколы аутентификации с нулевым разглашением знаний, построенные на основе модулярных кодов класса вычетов. Использование этих кодов позволяет заменить вычислительное устройство, реализующее операцию возведения в степень по модулю, на кодопреобразователь. В результате сложная вычислительная операция будет выполнена за один такт. Эффективность работы кодопреобразователей во многом зависит от правильности перехода от таблицы истинности к совершенным нормальным формам булевых функций. Авторами данной статьи разработаны программный код и графическое интерактивное приложение для ЭВМ, которое позволяет получать совершенные дизъюнктивные и (или) совершенные конъюнктивные нормальные формы согласно описанному пользователем содержанию таблиц истинности и выводить результат в соответствующем поле с использованием логических функций библиотеки Logic или в виде формулы. Совершенные формы можно получить с использованием описания таблицы истинности в виде минтермов (макстермов) булевой функции, а также номеров наборов минтермов (макстермов). В разработанном приложении существует возможность выбора типа получаемой совершенной формы, и оно содержит справочные данные по использованию. Программный код и весь графический интерфейс написаны с помощью встроенного языка и библиотек системы компьютерной алгебры Maple. Созданное интерактивное приложение интуитивно понятно и доступно даже непрофессиональным программистам (преподавателям математики, студентам). Для удобства программный код оформлен в виде графического приложения, требующего для работы установленной на компьютере системы Maple. Разработанное приложение может быть использовано образовательными организациями, в которых преподаются математическая логика, дискретная математика или их разделы.

**Ключевые слова:** математическая логика, булевы функции, совершенные дизъюнктивные и конъюнктивные нормальные формы, системы компьютерной алгебры, Maple

**Благодарности.** Исследование выполнено за счет гранта РФФИ № 23-21-00036

Появление низкоорбитальных систем спутникового Интернета делает достаточно актуальной проблему аутентификации космических аппаратов. Это связано с тем, что увеличение числа их группировок (например, Starlink содержит более 2 000 спутников) приводит к повышению вероятности навязывания неавторизованного контента чужим спутником. Для предотвращения такого деструктивного воздействия в работах [1, 2] предлагается использовать систему опознавания космических аппаратов, построенную на основе протокола аутентификации с нулевым разглашением знаний. В этом протоколе проверяющий (запросчик) генерирует сигнал-вопрос, который передается претенденту (ответчику). В зависимости от полученного сигнала-вопроса претендент вычисляет ответ и передает его проверяющему. Запросчик проверяет данный ответ. Если он правильный, претендент считается авторизованным абонентом. Для обеспечения

высокой имитостойкости к подбору правильного ответа в этих протоколах применяются операции возведения в степень по модулю  $Q$ . Так как в качестве модуля используются большие простые числа, на выполнение операций с ними требуются значительные временные затраты. В результате увеличивается интервал, в течение которого злоумышленник может подбирать правильный ответ. Уменьшить временные затраты на опознание спутника можно за счет использования *модулярных кодов классов вычетов* (МККВ), являющихся арифметическими [3–5]. Для построения МККВ выбираются простые числа  $p_1, p_2, \dots, p_n$ , для которых справедливо

$$p_1 < p_2 < \dots < p_n. \quad (1)$$

Это основания модулярного кода. Их произведение определяет диапазон МККВ:

$$P_n = \prod_{i=1}^n p_i. \quad (2)$$

Тогда любое целое число  $A < P_n$  можно однозначно представить в виде набора остатков, получаемых при делении числа  $A$  на основания МККВ,

$$A = (a_1, a_2, \dots, a_n), \tag{3}$$

где  $a_i \equiv A \pmod{p_i}, i = 1, \dots, n$ .

В МККВ модульные операции можно свести к соответствующим операциям над остатками. Пусть заданы два простых числа:  $A = (a_1, a_2, \dots, a_n)$  и  $W = (w_1, w_2, \dots, w_n)$ . Тогда для операций сложения, вычитания и умножения справедливо

$$A + W = ((a_1 + w_1) \pmod{p_1}, \dots, (a_n + w_n) \pmod{p_n}), \tag{4}$$

$$A - W = ((a_1 - w_1) \pmod{p_1}, \dots, (a_n - w_n) \pmod{p_n}), \tag{5}$$

$$A \cdot W = ((a_1 \cdot w_1) \pmod{p_1}, \dots, (a_n \cdot w_n) \pmod{p_n}), \tag{6}$$

где  $w_i \equiv W \pmod{p_i}, i = 1, \dots, n$ .

Использование модулярного кода классов вычетов позволяет повысить скорость выполнения модульных операций. Анализ выражений (4)–(6) показывает, что в МККВ вычисления выполняются параллельно по основаниям кода. Рассмотрим это на примере. Пусть заданы основания МККВ  $p_1 = 2, p_2 = 3, p_3 = 5$ . Рабочий диапазон равен 30. Выполним операцию умножения двух чисел:

$$A = 4 = (0, 1, 4) \text{ и } W = 7 = (1, 1, 2).$$

Получаем

$$A \cdot W = ((0 \cdot 1) \pmod{2}, (1 \cdot 1) \pmod{3}, (4 \cdot 2) \pmod{5}) = (0, 1, 3) = 28.$$

Разрядность операндов-остатков  $a_1, a_2, \dots, a_n$  значительно меньше самого числа  $A$ , что также позволяет повысить скорость вычислений. Например, если в качестве оснований МККВ взять шестизрядные простые числа  $p_1 = 37, p_2 = 41, p_3 = 43, p_4 = 47, p_5 = 53, p_6 = 59, p_7 = 61$ , то рабочий диапазон МККВ  $P_n = 584803025179$ . Тогда число  $A = 584803025170$  можно представить в виде набора остатков МККВ:  $A = (28, 32, 34, 38, 44, 50, 52)$ . В этом случае 40-разрядное число представляется в виде набора шестизрядных остатков. Очевидно, что использование МККВ позволяет повысить скорость выполнения модульных операций.

Тогда, подобрав основания МККВ так, чтобы выполнялось условие  $P_n > Q$ , операцию вычисления истинного статуса спутника в протоколе аутентификации [2]

$$C = g^U g^{S[j]} g^{T[j]} \pmod{Q} \tag{7}$$

можно заменить на  $n$  параллельных вычислений в МККВ:

$$\begin{cases} C_1 = g^{U_1} g^{S_1[j]} g^{T_1[j]} \pmod{p_1}, \\ C_2 = g^{U_2} g^{S_2[j]} g^{T_2[j]} \pmod{p_2}, \\ \vdots \\ C_n = g^{U_n} g^{S_n[j]} g^{T_n[j]} \pmod{p_n}, \end{cases} \tag{8}$$

где  $U, S[j], T[j]$  – секретные параметры протокола на  $j$ -м сеансе аутентификации;  $g$  – порождающий элемент мультипликативной группы по модулю  $p_i$ ;  $U_i \equiv U \pmod{p_i}, S[j] \equiv S[j] \pmod{p_i}, T[j] \equiv T[j] \pmod{p_i}, i = 1, \dots, n$ .

Дальнейшего снижения временных затрат на аутентификацию космического аппарата можно достичь за счет использования вместо вычислительного устройства, выполняющего операцию возведения в степень по модулю  $p_i$ , кодопреобразователя. Если на его вход подать значение показателя степени, например  $S_i[j]$ , то на выходе получим значение  $g^{S_i[j]} \pmod{p_i}$ , где  $i = 1, \dots, n$ . Пусть в качестве основания МККВ выбрано число  $p_1 = 11$ . Для данного модуля порождающим элементом мультипликативной группы можно взять  $g = 2$ . Пусть  $S_1[j] = 5$ . Используя вычислительное устройство, получаем  $g^{S_1[j]} \pmod{p_1} = 2^5 \pmod{11} = 10$ .

Эту мультипликативную операцию можно выполнить с помощью кода преобразователя. Если на его вход подать число 5, то на выходе кодопреобразователя появится число 10. В данном случае операция возведения в степень по модулю выполнится за один такт.

Однако для разработки такого преобразователя необходимо грамотно использовать методы дискретной математики. Освоение разделов дискретной математики и математической логики предполагает изучение таких тем, как множества, математическая индукция, математическая логика, отношения, функции, анализ алгоритмов, теория графов, комбинаторика, теория вероятностей, рекуррентные соотношения [6–8]. Усвоение этих разделов позволяет понять и решить задачу синтеза кодопреобразователя, которая состоит в построении схемы для заданной булевой функции или системы булевых функций на основе определенной системы логических элементов. Как правило, исходное описание для синтеза схемы задается либо в виде таблицы истинности, либо в аналитической форме в виде формулы [9]. При решении задачи синтеза комбинационной схемы, реализующей заданную булеву функцию, предварительно производятся минимизация

булевой функции и дальнейшее упрощение минимальной формы путем факторизации и декомпозиции [10, 11]. Комбинационная схема строится в заданном базисе [12]. Вопросы построения и оптимизации функционально-структурных описаний цифровых устройств рассмотрены в [13]. Методы, алгоритмы и программы решения задач минимизации *дизъюнктивных нормальных форм* (ДНФ) представлений булевых функций, которые широко используются при проектировании цифровых систем для уменьшения сложности (площади кристаллов) функциональных комбинационных блоков, рассмотрены, например, в [14, 15].

Анализ источников и задачи синтеза кодопреобразователя (комбинационной схемы) показывает необходимость построения *совершенных дизъюнктивных нормальных форм* (СДНФ) или *совершенных конъюнктивных нормальных форм* (СКНФ) по таблицам истинности, являющихся одним из наиболее наглядных элементов формальной логики и одной из возможных форм представления функционирования комбинационного аппарата.

То есть необходимо получить ДНФ (КНФ), называемую канонической или совершенной, причем все ее элементарные конъюнкции (дизъюнкции) являются конститuentами единицы (нуля). При этом элементарные конъюнкции (дизъюнкции) называются конститuentами единицы (нуля), если содержат в прямом или инверсном виде все переменные, являющиеся аргументами булевой функции [9].

Для решения задач дискретной математики и математической логики можно использовать средства вычислительной техники: так, вопросы использования методов дискретной математики и описания важнейших алгоритмов на дискретных структурах изложены в [16–18].

Для работы с элементами логики и булевой алгебры и их изучения можно использовать систему компьютерной алгебры Maple [8, 19, 20]. Эта система универсальна, гибка в использовании, одна из немногих, имеющих свою библиотеку для работы с булевыми функциями Logic, которая позволяет наглядно представить построение таблиц истинности (соответствия) и исследовать логическую эквивалентность составных высказываний. Однако библиотека Logic не имеет встроенных функций, позволяющих описать минтерм или макстерм и, соответственно, получить СДНФ или СКНФ согласно запросу пользователя.

Таким образом, несмотря на очевидную полезность и необходимость функций библио-

теки Logic, система не предусматривает описания таблицы истинности, а результаты преобразований составных высказываний выдает в упрощенном виде, который не всегда совпадает с совершенными формами.

В связи с этим предлагается программное решение, позволяющее развить систему Maple, а точнее – ее библиотеку Logic. Его использование в процессе обучения будет способствовать улучшению понимания обучающимися построения для булевых функций СДНФ и СКНФ, а также позволит изучать дискретную математику, математическую логику и другие дисциплины с использованием системы компьютерной алгебры.

Исследования по созданию программных кодов для изучения математической логики проводят как отечественные [21, 22], так и зарубежные [23–25] специалисты. В работе [22] представлены исходные коды для изучения различных разделов дискретной математики в виде программных рабочих листов, в том числе и булевой алгебры, и исследованы основные тенденции использования системы компьютерной алгебры Maple по компьютерному экспериментированию в дискретной математике. Описаны методы доказательств законов теории множеств, получения СДНФ и СКНФ, а также предложена программная реализация этих и других методов, используемых в дискретной математике. В работе [24] представлена модификация программного исходного кода для построения СДНФ и СКНФ без упрощений. Программный код реализован на языке программирования универсальной системы компьютерной алгебры Mathematica. Это сделано, чтобы помочь студентам, изучающим дискретную математику, лучше понять процесс построения и использования булевых функций.

Сегодня существуют приложения для получения СДНФ и СКНФ, написанные практически на всех популярных языках программирования, однако получение СДНФ и СКНФ по таблице истинности или в числовой форме для создания различных схемотехнических решений, в том числе и кодопреобразователей, не рассматривается. В данной статье представлен способ создания исходного кода программы и интерактивного приложения для построения СДНФ и СКНФ [21, 26], полностью написанного в среде системы Maple с помощью библиотеки Maplelet.

Разработанное в Maple приложение для персонального компьютера может помочь полу-

читать СДНФ и СКНФ по таблице истинности двумя способами:

- с помощью первоначального описания макстермов или минтермов в виде набора логических констант true или false;

- используя десятичные номера макстермов или минтермов (в числовой форме).

Таким образом, цифровизация всех сфер жизни, в частности, образования, обуславливает актуальность создания исходного кода программы и интерактивных приложений с простым и понятным интерфейсом для выполнения разнообразных задач.

### Разработка процедур нахождения СДНФ и СКНФ

Для получения СДНФ или СКНФ в системе Maple были созданы функции для описания минтерма или макстерма в выбранной форме, входящих в СДНФ или СКНФ соответственно, а затем выполнены дизъюнкция или конъюнкция описанных макстермов или минтермов соответственно. Рассмотрим порядок нахождения СДНФ.

1. *Описание макстерма или минтерма с использованием констант true или false.*

Процедура **Описание\_таблицы\_истинности\_SDNF** позволяет выполнять нахождение СДНФ по предъявленному условию – описание минтермов с использованием констант true или false.

Данная процедура состоит из двух частей: непосредственно нахождения минтерма и объединения полученных макстермов.

Аргументами для процедуры нахождения минтерма являются описание строки таблицы истинности (представленной с использованием констант true или false), принимающей значение true, и наименования используемых переменных. Чтобы создать `minterm`, связанный с элементом таблицы истинности, сначала инициализируется `minterm`, равный NULL. Затем в цикле `for` с переменной цикла `i` и диапазоном для всех используемых переменных проверяется, принимает ли описываемая переменная истинностное значение true или false. Если переменная имеет значение true, то `minterm` обновляется, объединяясь с именем соответствующей переменной. Если запись имеет значение false, то `minterm` обновляется, формируя конъюнкцию с отрицанием переменной. Итогом работы цикла является соответствующий `minterm`:

```
Описание_минтерма:=proc(row::
::list(truefalse),nep::list(symbol))%
получение минтерма
```

```
local минтерм, i; % используемые пе-
ременные
uses Logic;
if nops(row) <> nops(nep) then
error "Проверьте правильность
указания количества переменных";
end if;
минтерм := NULL; % получение описан-
ного минтерма
for i from 1 to nops(row) do
if row[i] then
минтерм:= минтерм &and nep[i];
else
минтерм:= минтерм &and &not
(nep[i]);
end if;
end do;
return минтерм;
end proc;
```

Для завершения разработки СДНФ необходимо сформировать дизъюнкцию `minterm`, созданную процедурой **Описание\_минтерма**. Вторая часть процедуры – **Описание\_таблицы\_истинности\_SDNF** – использует в качестве аргументов описание строк таблицы истинности, принимающих значения true и описанных с использованием констант true или false, и наименования используемых переменных. Сначала проверяется, не был ли переданный набор пустым. Если да, то возвращается выражение false. Если аргументы заданы правильно, то инициализируется результат NULL. Для каждого необходимого элемента таблицы истинности вызывается процедура **Описание\_минтерма** и с помощью оператора `&or` (операция «или») выходные данные этой процедуры добавляются к имеющемуся результату:

```
> Описание_таблицы_истинности_SDNF:=
proc(T::set(list(true-
false)),переменные::list(symbol))
local Накопление, значения, форма; %
используемые переменные
if T = {} then % проверка наличия
описания функции
return false;
end if;
Накопление:= NULL; % формирование
СДНФ
for значения in T do
форма:= Описание_минтерма(значе-
ния,переменные);
Накопление:= Накопление &or форма;
end do;
return Накопление;
end proc;
```

Разработанная процедура нахождения СКНФ отличается только порядком нахождения макстерма (используется операция дизъ-

юнкции в системе Maple – &or) и операцией накопления результатов макстермов, то есть необходимо произвести замену операции дизъюнкции (в системе Maple – &or) на операцию конъюнкции (в системе Maple – &and).

2. *Описание макстерма или минтерма в форме номера набора.*

Для нахождения СКНФ или СДНФ по номерам наборов дополнительно используется перевод в вид описания минтермов или макстермов с использованием констант true или false, рассмотренный ранее. Приведем фрагмент программного кода, отвечающего за данный перевод:

```
> число:= ListTools:-Reverse(convert(x,base,2));
   m:= nops(число); % определение
необходимого числа разрядов в минтерме
(макстерме)
   k:= nops(пер); % определение
числа используемых переменных
   if m > k then error "Требуется
%d цифр", m end if; % определение необ-
ходимого числа разрядов
   A:=[0 $ (k-m), op(число)]; %
заполнение значением «нуль» избыточных
разрядов
   for j to nops(A) do % пред-
ставление числа с использованием кон-
стант true, false
       if A[j]=1 then A:=subsop
(j=true,A) else A:=subsop(j=false,A);
end if;
   end do;
```

Данная процедура обеспечивает следующие преобразования:

- перевод десятичного числа (номера набора) в двоичный вид;
- проверка соответствия необходимого числа разрядов (строк в таблице истинности) и количества разрядов полученного числа; в случае несоответствия (меньшего числа разрядов при представлении числа в двоичном виде) заполнение недостающих позиций значениями «нуль»;
- представление двоичного числа с применением констант true или false, используя цикл for.

Таким образом, процедура, приведенная выше, позволяет преобразовать десятичное число в описание двоичного числа, представленного с использованием констант true или false. Полученные значения могут применяться в качестве первого аргумента процедуры **Описание\_таблицы\_истинности\_SDNF**, а значит, можно получить и СДНФ, порядок получения которой описан выше.

## Среда разработки

Современные компьютерные технологии, в том числе и система Maple, позволяют решать разнообразные задачи как дискретной математики, так и математической логики. В большинстве случаев демонстрацию решения конкретной задачи можно упростить, используя windows-приложения оконной, а не консольной категории. Система Maple позволяет создавать такие оконные приложения, поскольку имеет широкий спектр компонентов, направленных на создание графического интерфейса, обладающего функциональными возможностями, аналогичными системам визуального программирования. Эти возможности обеспечиваются использованием библиотеки Maplet. Встроенный высокоуровневый язык математических расчетов Maple дает возможность применять встроенные решения для создания новых приложений. Задачей авторов данного исследования являлось написание интерактивного приложения, полностью созданного в рамках этой программной среды, с графическим интерфейсом, выдающим СДНФ или СКНФ по выбору пользователя при вводе данных, описывающих таблицу истинности (соответствия).

Эта задача была успешно решена. Программный код написан на языке Maple с использованием библиотеки Maplet и представляет собой отдельное приложение в формате Maplets. Процедура получения СДНФ или СКНФ выглядит следующим образом. После запуска программного кода появляется графический интерфейс, обеспечивающий выбор описания исследуемой таблицы истинности, то есть окно графического интерфейса, в котором нужно выбрать форму описания таблицы истинности (соответствия). Исходное положение представлено на рисунке 1. Нажатие кнопки «Описание наборов» или «Номера наборов» обеспечивает выбор формы описания таблицы истинности. Далее производится переход на один из выбранных Maplets, позволяющих внести данные из таблицы истинности (соответствия) и выбрать представляемую форму (рис. 2). Генерация самой формы производится по нажатию кнопки «Получить». Интерфейс пользователя имеет максимально удобный минималистичный вид. Пользователь вводит в соответствующее поле необходимые данные описания минтерма или макстерма, количество переменных, от которых зависит булева функция, а также выбирает вид представляемой

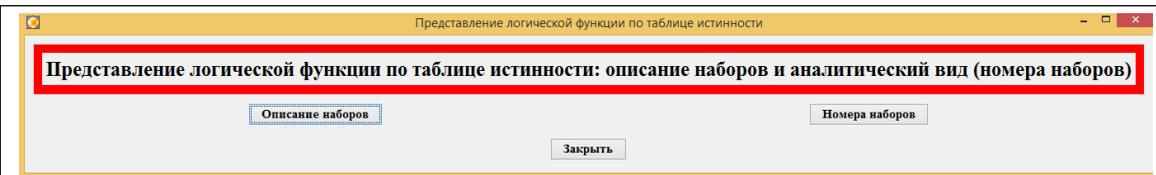


Рис. 1. Общий вид Maplets для получения СДНФ или СКНФ по таблице истинности  
 Fig. 1. General view of maplets for obtaining SDNF or SKNF according to the truth table

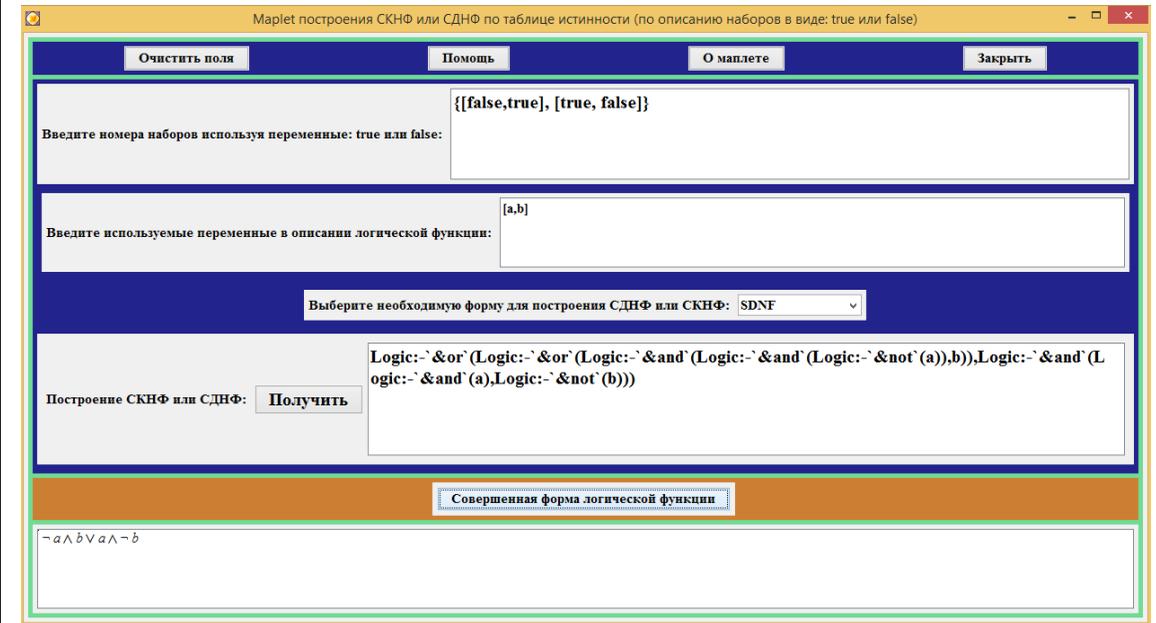


Рис. 2. Пример получения СДНФ для таблицы истинности булевой функции двух переменных для случая 1  
 Fig. 2. An example of obtaining SDNF for the truth table of a Boolean function of two variables for the case 1

функции. Далее нажимает кнопку «Получить». В итоге появляется представление выбранной формы.

В таблице 1 показан пример ввода данных.

Таблица 1  
 Таблица истинности  
 A truth table  
 Table 1

№ набора	<i>a</i>	<i>b</i>	<i>F(a, b)</i>
0	false	false	false
1	false	true	true
2	true	false	true
3	true	true	false

Значения для ввода формируются следующим образом.

Для первого случая (получение СДНФ с использованием описания минтерма в виде логи-

ческих переменных true, false) используемые переменные *a* и *b* представляются в виде логических констант: первый набор – [false, true], второй набор – [true, false]. Итог выполнения программного кода представлен на рисунке 2.

Для второго случая (получение СДНФ по номерам наборов) используемые переменные остаются прежними – *a* и *b*, номера наборов – [1, 2]. Итог выполнения программного кода представлен на рисунке 3.

Итоговые значения совпадают, что подтверждает правильность разработанных функций. Получение СКНФ осуществляется аналогичным способом.

Опишем подробнее применяемые команды для выбора необходимого представления.

Нажатием кнопки «Описание наборов» открывается Maplet «Maplet построения СКНФ или СДНФ по таблице истинности (по описанию наборов в виде true или false)».

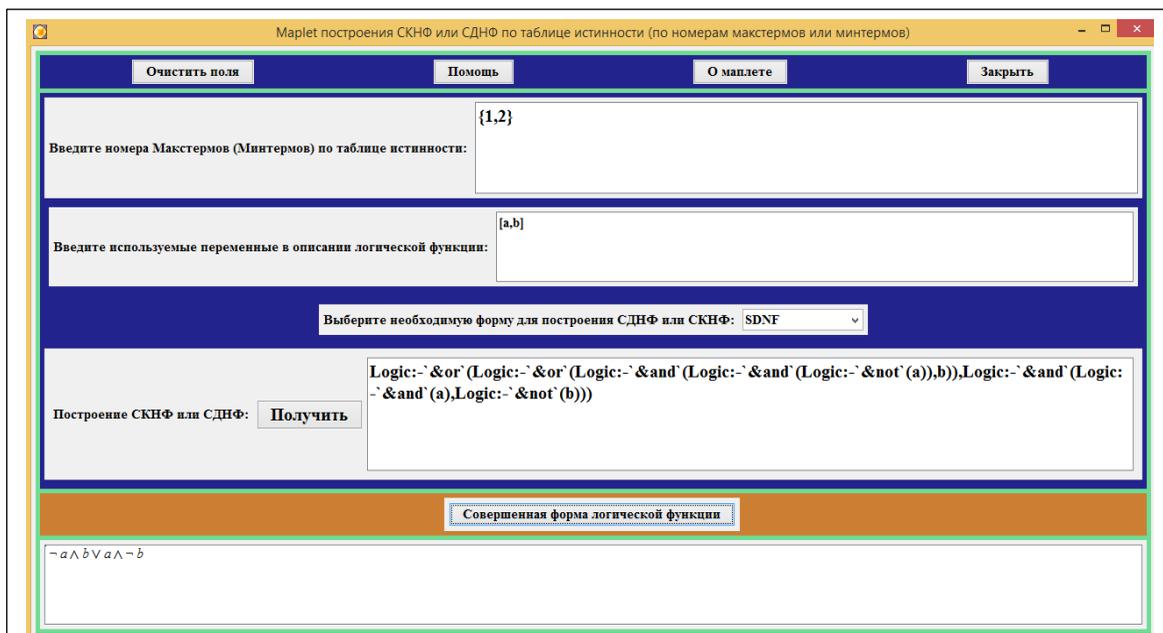


Рис. 3. Пример получения СДНФ для таблицы истинности булевой функции двух переменных для случая 2

Fig. 3. An example of obtaining SDNF for the truth table of a Boolean function of two variables for the case 2

Начало работы данного Maplets сопровождается вводом данных, который обеспечивает выполнение разработанной функции «**Описание таблицы истинности SKNF или SDNF**». Входными данными являются (T::set(list(true false)),переменные::list(symbol),вид::symbol), где переменная «T» обеспечивает просмотр представления вводимых данных в виде описания значений таблицы истинности, принимающей значения в виде логических значений true или false, то есть используются правила описания дизъюнктивной или конъюнктивной формы. Данные переменные описывают выходные значения логической функции. Вторая составляющая, от которой зависит функция, «переменные» – обеспечивает формирование наименований переменных, от которых зависит СДНФ или СКНФ, эти данные вводятся в виде списка. Третья переменная – «вид» – указывает формируемый вид булевой функции (СДНФ или СКНФ).

Нажатие функциональных кнопок «Очистить поля» и «Заккрыть» приводит к выполнению описанного действия – закрытию Maplet или удалению введенных значений. Использование вспомогательных кнопок «Помощь» и «О маплете» предоставляет вспомогательную информацию о порядке использования Maplet или его предназначении.

Выпадающий список позволяет подключить необходимый порядок вычисления (обеспечивает подключение необходимых функций вычисления). Приведем фрагмент программного кода, осуществляющего такой вывод:

```
> Описание_таблицы_истинности_
SKNF_или_SDNF_по_номерам := proc(T::set
(nonnegint), переменные::list(symbol),
вид::symbol)
local Накопление, значения, форма; %
используемые переменные
if вид=SDNF then СДНФ_по_но-
мерам(T, переменные); % процедуры для
преобразования
elif вид=SKNF then СКНФ_по_но-
мерам(T, переменные);
else error "Нет такой формы";
end if;
end proc;
```

**Создание графического интерфейса пользователя для программного кода**

Для разработки интерактивного приложения получения СДНФ или СКНФ в зависимости от вводимых данных использован метод визуально-ориентированного программирования [19]. Данный метод представлен в Maple с использованием специальной библиотеки Maplet, функции которой обеспечивают формирование графического интерфейса. Графический интерфейс разрабатываемого интерак-

тивного приложения создается путем описания нужных элементов будущего приложения из этой библиотеки. Необходимые действия прописываются при активации элементов и при обработке событий элементов пользовательского интерфейса.

Для удобства использования подготовленного приложения был разработан графический интерфейс GUI [27].

Нажатием кнопок «Описание наборов» и «Номера наборов» обеспечивается выбор формы вводимых данных и описания макстермов или минтермов в выбранном виде. Нажатие любой из них приводит к открытию соответствующего Maplet.

При описании наборов в виде переменных true или false Maplet (**Maplet построения СКНФ или СДНФ по таблице истинности (по описанию наборов в виде: true или false)**) имеет три зоны действия: вспомогательную, ввода информации и вывода результата.

Вспомогательная зона содержит четыре кнопки: справочные «Помощь» и «О маплете» и функциональные «Очистить поля» и «Закрыть».

Зона ввода включает в себя два поля ввода информации и поле с выпадающим списком. Поля ввода обеспечивают входными данными разработанные внутренние функции **Описание\_таблицы\_истинности\_SDNF**, **Описание\_таблицы\_истинности\_SKNF**, **СДНФ\_по\_номерам**, **СКНФ\_по\_номерам**, то есть позволяют формировать макстерм или минтерм. Выбранное значение из поля со списком дает возможность получить булеву функцию – СДНФ или СКНФ.

Зона вывода результата включает две кнопки – «Получить» и «Совершенная форма логической функции». Нажатие кнопки «Получить» позволяет сформировать СКНФ или СДНФ в соответствующем поле вывода в виде значений библиотеки Logic, а нажатие кнопки «Совершенная форма логической функции» – получить СДНФ или СКНФ в виде математической записи.

Все элементы созданного интерфейса GUI для удобства использования имеют соответствующие обозначения.

Виды Maplet (**Maplet построения СКНФ или СДНФ по таблице истинности (по номерам макстермов или минтермов)**) существенных отличий по структуре между собой не имеют. Выбор необходимого вида преобразования обеспечивается первоначальным выбором, а также данными, выводимыми по умолчанию.

Для этого в Maplet (**Maplet построения СКНФ или СДНФ по таблице истинности (по описанию наборов в виде: true или false)**) по умолчанию в полях ввода выводится форма представления вводимых данных: {[true, false]}, для **Maplet построения СКНФ или СДНФ по таблице истинности (по номерам макстермов или минтермов)**: {0}. В поле ввода «Введите используемые переменные в описание логической функции» по умолчанию выводится [a, b], то есть булева функция зависит от двух переменных. При ошибках формата ввода будет выведено сообщение об ошибке. Изменения возможны без перезапуска приложения.

Существуют ограничения использования приложения: для описания входных переменных необходимо использовать строчные буквы латинского алфавита, ввод наименований с верхними или нижними индексами запрещен.

Эффективность использования программного кода и созданного интерактивного графического приложения проверялась в ходе учебных занятий по математической логике, дискретной математике, а также в научных исследованиях при выполнении построения кодопреобразователей.

Таблица 2

Усвоение учебного материала (%)

Table 2

Acquisition of learning material (%)

Тема	Традиционный метод	Использование СКА Maple	Интерактивное графическое приложение
Таблицы истинности	64	68	72
Совершенные КНФ и ДНФ	59	64	76
Построение комбинационных схем	52	65	67

Анализ полученных данных (табл. 2) показывает, что по сравнению с традиционными методами обучения использование как программного кода, так и интерактивного графического приложения дает улучшение понимания учебного материала от 4 до 18 процентов.

### Заключение

Сферы применения дискретной математики постоянно расширяются. Ее методы нашли приложение в системах тестирования программных продуктов, при разработке реляционных БД, в логистике. Новой областью использования дискретной математики могут стать криптографические протоколы аутентификации, реализованные в модулярных кодах классов вычетов. С помощью данных методов можно заменить вычислительное устройство, реализующее операцию возведения в степень по модулю, на кодопреобразователь. В результате сложная вычислительная операция будет выполнена за один такт. Очевидно, что эффективность работы кодопреобразователей определяется прежде всего правильным переходом от таблицы истинности к СДНФ и СКНФ. Авторы данного исследования разработали программный код и интерактивное графическое

приложение для получения СДНФ и СКНФ, получаемых из описания таблицы истинности двумя способами: описанием макстермов (минтермов) с использованием логических констант true и false и описанием макстермов (минтермов) по номерам наборов. Интерактивные приложения позволяют получать представление и в виде формулы. Для написания исходного кода программ использован встроенный язык системы компьютерной алгебры Maple. Для удобства применения программные коды представлены в виде интерактивных приложений, способных к запуску на любом компьютере с установленной системой Maple. Благодаря возможности выбора вида формы данный программный код и графические интерактивные приложения можно использовать для решения широкого круга задач, связанных с синтезом простейших цифровых устройств, а также для преподавания математической логики и дискретной математики.

### Список литературы

1. Kalmykov I., Lapina M., Provornov I., Voloshin E. Development of imitation-resistant authentication protocol for low-orbital space satellite communication system. *Ceur WS Proc.*, 2019, vol. 2500, pp. 1–10.
2. Pashintsev V.P., Zhuk A.P., Kalmykov M.I., Olenev A.A. Redundant modular codes for development of fault-tolerant systems of satellite identification. *IJETER*, 2020, vol. 8, no. 7, pp. 3160–3168. doi: 10.30534/ijeter/2020/47872020.
3. Valueva M.V., Nagornov N.N., Lyakhov P.A., Valuev G.V., Chervyakov N.I. Application of the residue number system to reduce hardware costs of the convolutional neural network implementation. *Math. and Comput. in Simulation*, 2020, vol. 177, pp. 232–243. doi: 10.1016/j.matcom.2020.04.031.
4. Tchernykh A., Schwiegelsohn U., Talbi E., Babenko M. Towards understanding uncertainty in cloud computing with risks of confidentiality, integrity, and availability. *J. of Computational Sci.*, 2019, vol. 36, art. 100581. doi: 10.1016/j.jocs.2016.11.011.
5. Samimi N., Kamal M., Afzalli-Kusha A., Pedram M. Res-DNN: A Residue number system-based DNN accelerator unit. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 2020, vol. 67, no. 2, pp. 658–671. doi: 10.1109/TCSI.2019.2951083.
6. Mosley A. *An Introduction to Logic: From Everyday Life to Formal Systems*. Northampton, Massachusetts, 2019, 385 p.
7. O’Leary M.L. *A First Course in Mathematical Logic and Set Theory*. NJ, John Wiley & Sons Publ., 2016, 416 p.
8. Rosen K.H. *Discrete Mathematics and its Applications*. NY, McGraw-Hill Publ., 2019, 1118 p.
9. Rusnak P. Transformation of Boolean expression into disjunctive or conjunctive normal form. *CERes J.*, 2017, vol. 3, no. 1, pp. 43–49.
10. Бибило П.Н., Романов В.И. Система логической оптимизации функционально-структурных описаний цифровых устройств на основе продукционно-фреймовой модели представления знаний // Проблемы разработки перспективных микро- и нанoeлектронных систем. 2020. № 4. С. 9–16. doi: 10.31114/2078-7707-2020-4-9-16.
11. Михеева Е.А., Еникеева А.Ф. Минимизация булевых функций геометрическим методом // Ученые записки УлГУ. Сер. Математика и информационные технологии. 2018. № 1. С. 72–82.
12. Riznyk V., Solomko M. Minimization of Boolean functions by combinatorial method. *Technology Audit and Production Reserves*, 2017, vol. 4, no. 36, pp. 49–64. doi: 10.15587/2312-8372.2017.108532.
13. Feng J., Zhao R., Cui Y. Simplification of logical functions with application to circuits. *Electronic Research Archive*, 2022, vol. 30, no. 9, pp. 3320–3336. doi: 10.3934/era.2022168.
14. Поттосин Ю.В. Метод многоблочной параллельной декомпозиции системы частичных булевых функций // Информатика. 2017. № 3. С. 92–98.
15. Поттосин Ю.В. Параллельная декомпозиция системы частичных булевых функций // Вестн. ТГУ. УВТИИ. 2018. № 45. С. 83–91. doi: 10.17223/19988605/45/10.
16. Alharbi E. Truth graph: A novel method for minimizing Boolean algebra expressions by using graphs. In: *LNAI. Proc. Diagrams*, 2020, pp. 461–469. doi: 10.1007/978-3-030-54249-8\_36.
17. Martin E. Disjunctive logic programs, answer sets, and the cut rule. *Archive for Math. Logic*, 2022, vol. 61, no. 7-8, pp. 903–937. doi: 10.1007/s00153-022-00821-x.

18. Rozo J.H.B. Using Boolean algebra to model the economic decision-making. *Brazilian J. of Business*, 2021, vol. 3, no. 2, pp. 1413–1426. doi: 10.34140/bjbv3n2-009.
19. Оленев А.А., Киричек К.А., Потехина Е.В. Математическая логика: построение логических схем из логических элементов в Maple // Вестн. КРАУНЦ. Физико-матем. науки. 2021. Т. 36. № 3. С. 155–164. doi: 10.26117/2079-6641-2021-36-3-155-164.
20. Durcheva M., Varbanova E. Applications of CAS in the teaching and learning of discrete mathematics. *Math. in Comput. Sci.*, 2017, vol. 11, pp. 305–314. doi: 10.1007/s11786-017-0310-8.
21. Olenov A.A., Zvereva L.G., Saieg T.H. Improving the learning and teaching of mathematical logic elements using Maple. *ЖНЕТР*, 2022, vol. 22, no. 8, pp. 51–57. doi: 10.33423/jhetp.v22i8.5315.
22. Оленев А.А., Киричек К.А., Потехина Е.В. Программа для визуализации основных логических операций: Свид. о регистр. ПрЭВМ № 2021613965. Рос. Федерация, 2021.
23. Durcheva M., Nikolova E. Modeling mathematical logic using MAPLE. *AIP Conf. Proc.*, 2018, vol. 2048, art. 060010. doi: 10.1063/1.5082125.
24. Nieto S., Ramos H. Constructing extended Boolean functions from truth tables using the Mathematica system. *Proc. SIIE*, 2016, pp. 1–6. doi: 10.1109/SIIE.2016.7751828.
25. Бибило П.Н., Логонова И.П. Экспериментальное сравнение эффективности программ минимизации систем булевых функций в классе дизъюнктивных нормальных форм // Информатика. 2022. Т. 19. № 2. С. 26–55. doi: 10.37661/1816-0301-2022-19-2-26-55.
26. Оленев А.А., Калмыков И.А., Киричек К.А., Проворнов И.А. Программа для визуализации построения совершенных нормальных форм формул алгебры логики: Свид. о регистр. ПрЭВМ № 2023614273. Рос. Федерация, 2023.
27. Thompson I. *Understanding Maple*. UK, Cambridge University Press, 2017, 233 p.

### Constructing perfect normal forms of Boolean functions for circuit implementations of authentication protocols using Maple

Aleksandr A. Olenov  
Igor A. Kalmykov  
Kseniya A. Kirichek

#### For citation

Olenov, A.A., Kalmykov, I.A., Kirichek, K.A. (2023) 'Constructing perfect normal forms of Boolean functions for circuit implementations of authentication protocols using Maple', *Software & Systems*, 36(3), pp. 349–360 (in Russ.). doi: 10.15827/0236-235X.142.349-360

#### Article info

Received: 15.03.2023

After revision: 02.05.2023

Accepted: 17.05.2023

**Abstract.** One of the promising areas of applying discrete mathematics are zero-knowledge authentication protocols based on modular residue class codes (MRCC). Using MRCC allows replacing a computing device that implements the operation of raising to a power modulo with a code converter. As a result, a complex computational operation is executable in one machine phase. It is obvious that the efficiency of code converters depends largely on the correct transition from the truth table to perfect normal forms of Boolean functions. The authors of this article have developed a software code and a graphical application for a computer that allows obtaining perfect disjunctive or conjunctive forms in accordance with the content

of the truth tables described by the user and displaying the result in the corresponding field in mathematical form. Perfect forms can be obtained using the description of the truth table both in the form of minterms (maxterms) of a Boolean function using the logical variables true or false, and using the numbers of sets of minterms (maxterms). There is a possibility to choose the type of the obtained perfect form. The developed application contains reference data on its use. The software code and the entire graphical user interface are written using the built-in language of the Maple computer algebra system, as well as using the Maplet library. It was possible to create an interactive application that is user-friendly for non-professional programmers (mathematics teachers, students). For the convenience of the end user, the program is designed as a graphical application that requires only the Maple system installed. The developed application can be used by educational organizations that study the academic disciplines of mathematical logic, discrete mathematics or their sections.

**Keywords:** mathematical logic, Boolean functions, perfect disjunctive and conjunctive normal forms, computer algebra systems, Maple

**Acknowledgements.** The study has been funded by a grant from the Russian Science Foundation, grant number 23-21-00036

### Reference List

1. Kalmykov, I., Lapina, M., Provornov, I., Voloshin, E. (2019) 'Development of imitation-resistant authentication protocol for low-orbital space satellite communication system', *Ceur WS Proc.*, vol. 2500, pp. 1–10.
2. Pashintsev, V.P., Zhuk, A.P., Kalmykov, M.I., Olenev, A.A. (2020) 'Redundant modular codes for development of fault-tolerant systems of satellite identification', *IJETER*, 8(7), pp. 3160–3168. doi: 10.30534/ijeter/2020/47872020.
3. Valueva, M.V., Nagornov, N.N., Lyakhov, P.A., Valuev, G.V., Chervyakov, N.I. (2020) 'Application of the residue number system to reduce hardware costs of the convolutional neural network implementation', *Math. and Comput. in Simulation*, 177, pp. 232–243. doi: 10.1016/j.matcom.2020.04.031.
4. Tchernykh, A., Schwiegelsohn, U., Talbi, E., Babenko, M. (2019) 'Towards understanding uncertainty in cloud computing with risks of confidentiality, integrity, and availability', *J. of Computational Sci.*, 36, art. 100581. doi: 10.1016/j.jocs.2016.11.011.
5. Samimi, N., Kamal, M., Afzalli-Kusha, A., Pedram, M. (2020) 'Res-DNN: A Residue number system-based DNN accelerator unit', *IEEE Transactions on Circuits and Systems I: Regular Papers*, 67(2), pp. 658–671. doi: 10.1109/TCSI.2019.2951083.
6. Mosley, A. (2019) *An Introduction to Logic: From Everyday Life to Formal Systems*. Northampton, Massachusetts, 385 p.
7. O'Leary, M.L. (2016) *A First Course in Mathematical Logic and Set Theory*. NJ: John Wiley & Sons Publ., 416 p.
8. Rosen, K.H. (2019) *Discrete Mathematics and its Applications*. NY: McGraw-Hill Publ., 1118 p.
9. Rusnak, P. (2017) 'Transformation of Boolean expression into disjunctive or conjunctive normal form', *CERes J.*, 3(1), pp. 43–49.
10. Bibilo, P.N., Romanov, V.I. (2020) 'The system of logical optimization of functional structural descriptions of digital circuits based on production-frame knowledge representation model', *Problems of Advanced Micro- and Nanoelectronic Systems Development*, (4), pp. 9–16 (in Russ.). doi: 10.31114/2078-7707-2020-4-9-16.
11. Mikheeva, E.A., Enikeeva, A.F. (2018) 'Minimization of Boolean functions by a geometric method', *Scientific Notes of UIGU. Series: Mathematics and Information Technology*, (1), pp. 72–82 (in Russ.).
12. Riznyk, V., Solomko, M. (2017) 'Minimization of Boolean functions by combinatorial method', *Technology Audit and Production Reserves*, 4(36), pp. 49–64. doi: 10.15587/2312-8372.2017.108532.
13. Feng, J., Zhao, R., Cui, Y. (2022) 'Simplification of logical functions with application to circuits', *Electronic Research Archive*, 30(9), pp. 3320–3336. doi: 10.3934/era.2022168.
14. Pottosin, Yu.V. (2017) 'A method for multi-block parallel decomposition of a system of partial Boolean functions', *Informatics*, (3), pp. 92–98 (in Russ.).
15. Pottosin, Yu.V. (2018) 'Parallel decomposition of a system of partial Boolean functions', *Tomsk State University J. of Control and Comput. Sci.*, (45), pp. 83–91 (in Russ.). doi: 10.17223/19988605/45/10.
16. Alharbi, E. (2020) 'Truth graph: A novel method for minimizing Boolean algebra expressions by using graphs', in *LNAI. Proc. Diagrams*, pp. 461–469. doi: 10.1007/978-3-030-54249-8\_36.
17. Martin, É. (2022) 'Disjunctive logic programs, answer sets, and the cut rule', *Archive for Math. Logic*, 61(7-8), pp. 903–937. doi: 10.1007/s00153-022-00821-x.
18. Rozo, J.H.B. (2021) 'Using Boolean algebra to model the economic decision-making', *Brazilian J. of Business*, 3(2), pp. 1413–1426. doi: 10.34140/bjbv3n2-009.
19. Olenev, A.A., Kirichek, K.A., Potekhina, E.V. (2021) 'Mathematical logic: Construction of logic circuits from logical elements in Maple', *Bull. KRASEC. Physical and Math. Sci.*, 36(3), pp. 155–164 (in Russ.). doi: 10.26117/2079-6641-2021-36-3-155-164.
20. Durcheva, M., Varbanova, E. (2017) 'Applications of CAS in the teaching and learning of discrete mathematics', *Math. in Comput. Sci.*, 11, pp. 305–314. doi: 10.1007/s11786-017-0310-8.
21. Olenev, A.A., Zvereva, L.G., Saieg, T.H. (2022) 'Improving the learning and teaching of mathematical logic elements using Maple', *JHETP*, 22(8), pp. 51–57. doi: 10.33423/jhetp.v22i8.5315.
22. Olenev, A.A., Kirichek, K.A., Potekhina, E.V. (2021) *Program for Visualization of basic Logical Operations*, Pat. RF, № 2021613965.
23. Durcheva, M., Nikolova, E. (2018) 'Modeling mathematical logic using MAPLE', *AIP Conf. Proc.*, 2048, art. 060010. doi: 10.1063/1.5082125.

24. Nieto, S., Ramos, H. (2016) 'Constructing extended Boolean functions from truth tables using the Mathematica system', *Proc. SIE*, pp. 1–6. doi: 10.1109/SIE.2016.7751828.
25. Bibilo, P.N., Loginova, I.P. (2022) 'Experimental comparison of the effectiveness of programs for minimizing systems of Boolean functions in the class of disjunctive normal forms', *Informatics*, 19(2), pp. 26–55 (in Russ.). doi: 10.37661/1816-0301-2022-19-2-26-55.
26. Olenev, A.A., Kalmykov, I.A., Kirichek, K.A., Provornov, I.A. (2023) *A Program for Visualizing the Construction of Perfect Normal Forms of Logic Algebra Formulas*. Pat. RF, № 2023614273.
27. Thompson, I. (2017) *Understanding Maple*. UK: Cambridge University Press, 233 p.

**Авторы**

**Оленев Александр Анатольевич**<sup>1</sup>, к.т.н.,  
доцент, доцент кафедры математики, информатики  
и цифровых образовательных технологий,  
olenevalexandr@gmail.com

**Калмыков Игорь Анатольевич**<sup>2</sup>, д.т.н., профессор,  
профессор кафедры информационной безопасности  
автоматизированных систем,  
kia762@yandex.ru

**Киричек Ксения Александровна**<sup>1</sup>, к.п.н., доцент,  
зав. кафедрой математики, информатики  
и цифровых образовательных технологий,  
kirichekka@mail.ru

**Authors**

**Aleksandr A. Olenev**<sup>1</sup>, Ph.D. (Engineering),  
Associate Professor of the Department of Mathematics,  
Informatics and Digital Educational Technologies,  
olenevalexandr@gmail.com

**Igor A. Kalmykov**<sup>2</sup>, Dr.Sc. (Engineering),  
Professor of the Department of Information Security  
of Automated Systems,  
kia762@yandex.ru

**Kseniya A. Kirichek**<sup>1</sup>, Ph.D. (Education),  
Associate Professor, Head of Chair of the Department  
of Mathematics, Informatics and Digital Educational  
Technologies, kirichekka@mail.ru

<sup>1</sup> Ставропольский государственный педагогический институт, г. Ставрополь, 355029, Россия

<sup>2</sup> Северо-Кавказский федеральный университет, г. Ставрополь, 355017, Россия

<sup>1</sup> Stavropol State Pedagogical Institute, Stavropol, 355029, Russian Federation

<sup>2</sup> North Caucasian Federal University, Stavropol, 355017, Russian Federation

## Fast quantum search algorithm modelling on conventional computers: Information analysis of the halting problem

Sergey V. Ulyanov  
Viktor S. Ulyanov

### For citation

Ulyanov, S.V., Ulyanov, V.S. (2023) 'Fast quantum search algorithm modelling on conventional computers: Information analysis of the halting problem', *Software & Systems*, 36(3), pp. 361–377 (in Russ.). doi: 10.15827/0236-235X.142.361-377

### Article info

Received: 31.03.2023

After revision: 19.06.2023

Accepted: 01.07.2023

**Abstract.** The paper describes the simplest technique for simulating a quantum algorithm based on the direct matrix representation of quantum operators. This approach is stable and precise, but it requires allocation of operator's matrices in the computer memory. Since the size of the operators grows exponentially, this approach is useful for simulating quantum algorithms with a relatively small number of qubits (e.g., approximately 11 qubits on a typical desktop computer). This approach enables relatively simple simulation of the operation of the solution quality control system and performance of a reliability analysis. A more efficient fast quality control simulation method is based on computing all or a part of operator matrices as needed on a current computational basis. This method makes it possible to avoid storing all or a part of the operator matrices. In this case, the number of qubits to be simulated (e.g., the number of input qubits or the number of qubits in the system state register) is affected by: (I) the exponential growth in the number of operations required to calculate the result of matrix products; and (II) the size of the state vector allocated in computer memory. It is reasonable if one embodiment of this approach involves simulating up to 19 or more qubits on typical desktop computer, and even more in a system with a vector architecture. Due to particularities of the memory addressing and access processes in a typical desktop computer (such as, for example, a Pentium PC), when a number of qubits is relatively small, the on-demand computing tends to be faster than the direct storage approach. The compute-on-demand approach benefits from applying the results of studying quantum operators and their structure that enables computing the matrix elements more efficiently. The paper considers effective simulation of Grover's quantum search algorithm using a computer with classical architecture.

**Keywords:** fast quantum search algorithm, quantum operators, state vector, algorithmic representation, quantum computing, minimum of Shannon information entropy, termination criteria

**Acknowledgements.** Fruitful discussion with P. Shor, Ch. Bennet, L. Levitin and V. Belavkin help us to formulate more clear results for quantum software engineering applications

The study of the on-demand computing approach for some quantum algorithms (QAs) can lead to a problem-oriented approach based on the quantum algorithm (QA) structure and state vector behavior [1–3]. For example, in Grover's quantum search algorithm (QSA) [4], the state vector always has one of the two different values: (I) one value corresponds to the probability amplitude of the answer; and (II) the second value corresponds to the probability amplitude of the rest of the state vector. Based on this assumption, it is possible to configure the algorithm using these two different values, and to efficiently simulate Grover's QSA. In this case, the primary limit is a representation of the floating-point numbers used to simulate the actual values of the probability amplitudes. After the superposition operation, these probability amplitudes are very small

$\left(\frac{1}{\sqrt{2^n}}\right)$ . Thus, it is possible to

simulate Grover's QSA with this approach simulating 1024 qubits or more without termination condition calculation and up to 64 qubits or more with termination condition estimation based on Shannon entropy.

Other QAs do not necessarily reduce to just two values. For those algorithms that reduce to a finite

number of values, we can use the techniques for simplifying the Grover's QSA, but the maximum number of input qubits to be simulated will tend to be smaller, because the probability amplitudes of other algorithms have relatively more complicated distributions.

Introduction of an external excitation can decrease the possible number of qubits for some algorithms. In some algorithms, the entanglement and interference operators can be bypassed (or simplified), and the output is computed based only on a superposition of the initial states (and destructive interference of the final output patterns) representing the state of the designed schedule of control gains. For example, it is possible to make a particular case of Deutsch-Jozsa's and Simon algorithms entanglement-free by using pseudo-pure quantum states [5].

The further disclosure begins with a comparative analysis of the temporal complexity of several representative QAs. The analysis is followed by an introduction of the generalized approach in QA simulation and algorithmic representation of quantum operators. Subsequent portions describe the structure representation of the QAs applicable to low level programming on classical computer (PC), generalizations of the approaches and intro-

duction of the general QA simulation tool based on fast problem-oriented QAs.

The simulation techniques are then applied to a quantum control algorithm.

It is possible to efficiently implement the matrix-based approach for a small number of input qubits. The above-mentioned matrix approach is a useful tool to illustrate complexity issues associated with QA simulation on classical computer.

### The structure of the QA gate system design

As shown in Fig. 1a, QA simulation can be represented as a generalized representation of a QA as a set of sequentially applied smaller quantum gates (see in details [1]).

Moreover, local optimization of QA components according to specific hardware realization

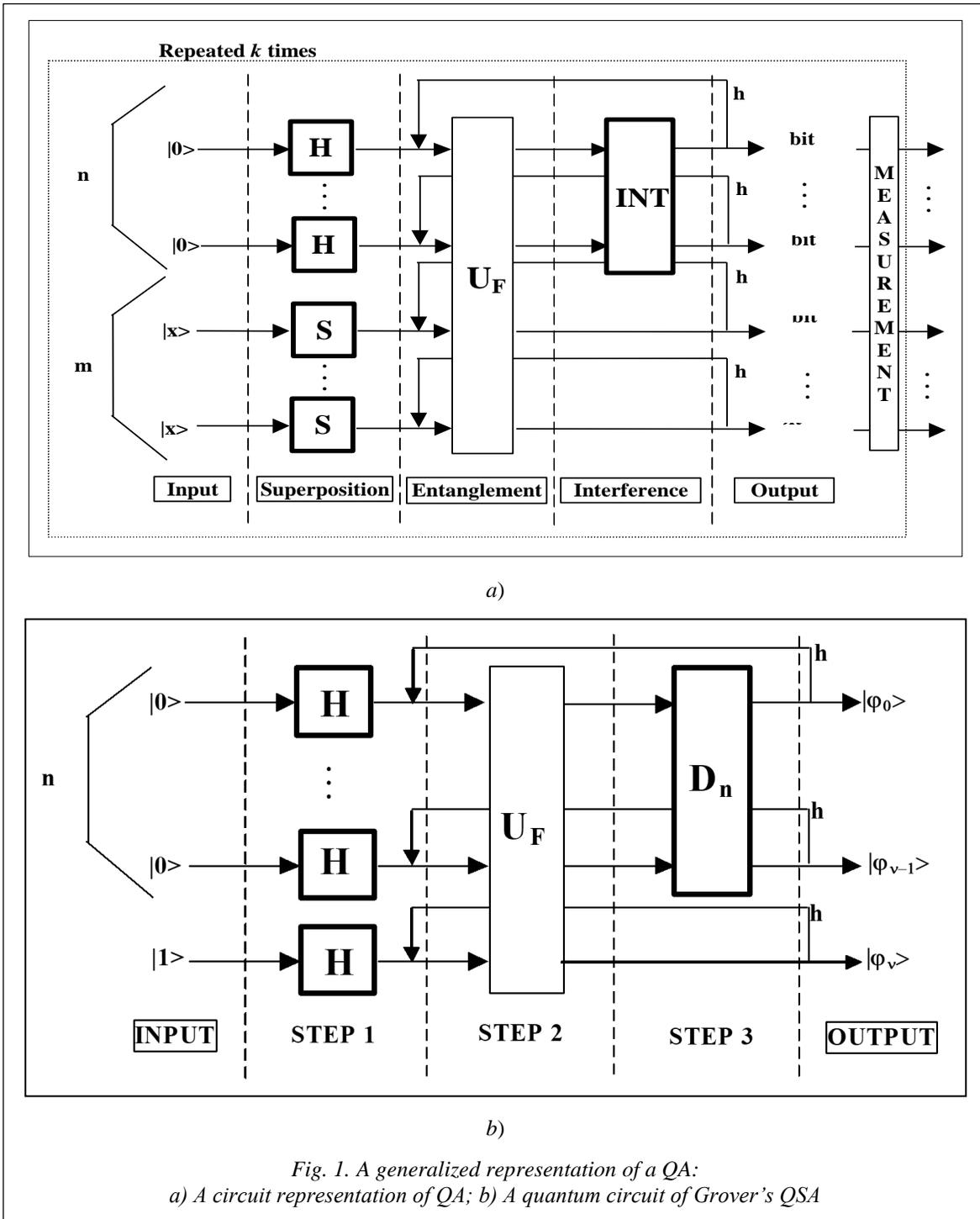


Fig. 1. A generalized representation of a QA:  
 a) A circuit representation of QA; b) A quantum circuit of Grover's QSA

makes it possible to develop appropriate hardware accelerators for QA simulation using classical gates.

### A generalized approach in QA simulation

In general, any QA can be represented as a circuit of smaller quantum gates as shown in Fig. 1. The circuit shown in Fig. 1a is divided into five general layers: (1) input; (2) superposition; (3) entanglement; (4) interference; and (5) output.

**Layer 1: Input.** The quantum state vector is set up to an initial value for this certain algorithm. For example, the input for Grover's QSA is a quantum state  $|\phi_0\rangle$  described as a tensor product:

$$\begin{aligned} |\phi_0\rangle &= a_1|0\rangle \otimes \dots \otimes |0\rangle \otimes |0\rangle + \\ &+ a_2|0\rangle \otimes \dots \otimes |0\rangle \otimes |1\rangle + \\ &+ a_3|0\rangle \otimes \dots \otimes |1\rangle \otimes |0\rangle + \dots \\ &\dots + a_n|1\rangle \otimes \dots \otimes |1\rangle \otimes |1\rangle = \\ &= 1|0\rangle \otimes \dots \otimes |0\rangle \otimes |1\rangle = |0\dots 01\rangle, \end{aligned} \tag{1}$$

where  $|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ ;  $|1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$ ;  $\otimes$  – denotes Kronecker tensor product operation.

Such quantum state can be presented as shown in Fig. 2a.

The coefficients  $a_i$  in Eq. (1) are called probability amplitudes. Probability amplitudes can take negative and/or complex values. However, the probability amplitudes must obey the following constraint:

$$\sum_i a_i^2 = 1. \tag{2}$$

The actual probability of the arbitrary quantum state  $a_i|i\rangle$  to be measured is calculated as a square

of its probability amplitude value  $p_i = |a_i|^2$ .

**Layer 2: Superposition.** The Walsh-Hadamard operator transforms the state of the quantum state vector so that probabilities are distributed uniformly among all basis states. The result of the superposition layer of Grover's QSA is shown in Fig. 2b as a probability amplitude representation, and in Fig. 3b as a probability representation.

**Layer 3: Entanglement.** Probability amplitudes of the basis vector corresponding to the current problem are flipped while rest basis vectors are left unchanged. Entanglement is typically provided by controlled-NOT (CNOT) operations. Figs 2c and 3c show the results of entanglement from the application of the operator to the state vector after superposition operation. An entanglement operation does not affect the probability of the state vector to

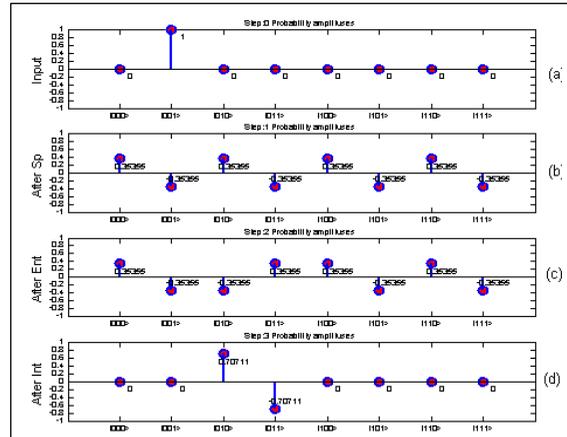


Fig. 2. Dynamics of Grover's QSA probability amplitudes of state vector on each algorithm step

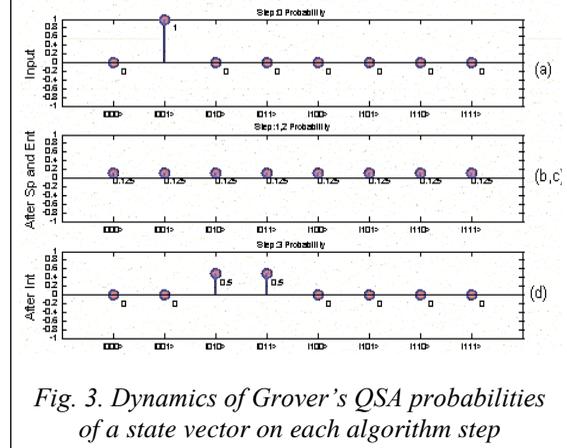


Fig. 3. Dynamics of Grover's QSA probabilities of a state vector on each algorithm step

be measured. Rather, entanglement prepares a state, which cannot be represented as a tensor product of simpler state vectors. For example, let us consider state  $\phi_1$  shown in Fig. 2b and state  $\phi_2$  presented in Fig. 2c:

$$\begin{aligned} \phi_1 &= 0.35355 \begin{pmatrix} |000\rangle - |001\rangle + |010\rangle - \\ -|011\rangle + |100\rangle - |101\rangle + \\ +|110\rangle - |111\rangle \end{pmatrix} = \\ &= 0.35355(|00\rangle + |01\rangle + |10\rangle + |11\rangle)(|0\rangle - |1\rangle), \\ \phi_2 &= 0.35355 \begin{pmatrix} |000\rangle - |001\rangle - |010\rangle + |011\rangle + \\ +|100\rangle - |101\rangle + |110\rangle - |111\rangle \end{pmatrix} = \\ &= 0.35355(|00\rangle - |01\rangle + |10\rangle + |11\rangle)|0\rangle - \\ &- 0.35355(|00\rangle + |01\rangle + |10\rangle + |11\rangle)|1\rangle. \end{aligned}$$

As shown above, the description of state  $\phi_1$  can be presented as a tensor product of simpler states, while state  $\phi_2$  (in the measurement basis  $\{|0\rangle, |1\rangle\}$ ) cannot.

**Layer 4: Interference.** Probability amplitudes are inverted about the average value. As a result,

the probability amplitude of states "marked" by entanglement operation will increase.

Figs 2d and 3d show the results of interference operator application.

Fig. 2d shows probability amplitudes and Fig. 3d shows probabilities.

**Layer 5: Output.** The output layer provides the measurement operation (extraction of the state with maximum probability) followed by interpretation of the result. For example, in the case of Grover's QSA, the required index is coded in the first  $n$  bits of the measured basis vector.

Since unitary quantum operators implement various QA layers, simulation of quantum operators depends on simulating such unitary operators. Thus, in order to develop an efficient simulation, it is useful to understand the nature of QA basic quantum operators.

### Basic QA operators

Let us consider the superposition, entanglement and interference operators from the simulation point of view. In this case, the superposition operators and the interference operators have more complicated structure and differ from algorithm to algorithm. Thus, it is first useful to consider the entanglement operators, since they have a similar structure for all QAs, and differ only by the analyzed function.

In general, the superposition operator is based on the combination of the tensor products Hadamard  $H$  operators:  $H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$  with the identity operator  $I : I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ .

*Remark.* As described in [1–3] the simulation system of quantum computation is based on quantum algorithm gates (QAG). The design process of QAG includes the matrix design form of three quantum operators: superposition ( $Sp$ ), entanglement ( $U_F$ ) and interference ( $Int$ ). In a general form, we can describe the structure of a QAG as follows:

$$QAG = \left[ (Int \otimes^n I) U_F \right]^{h+1} \left[ {}^n H \otimes^m S \right],$$

where  $I$  is the identity operator; the symbol  $\otimes$  denotes a tensor product;  $S$  is equal to  $I$  or  $H$  and depends on the problem description. One portion of the design process in QAG is the type-choice of the entanglement problem-dependent operator  $U_F$  that physically describes the qualitative properties of the function  $f$ .

The Hadamard Transform creates the superposition on classical states, and quantum operators

such as  $CNOT$  create robust entangled states. The Quantum Fast Fourier Transform (QFFT) produces interference. For most QAs the superposition operator can be expressed as

$$Sp = \left( \bigotimes_{i=1}^n H \right) \otimes \left( \bigotimes_{i=1}^m S \right), \tag{3}$$

where  $n$  and  $m$  are the numbers of inputs and of outputs respectively. The operator  $S$  depends on the algorithm and can be either the Hadamard operator  $H$  or the identity operator  $I$ . The numbers of outputs  $m$  as well as structures of the corresponding superposition and interference operators are presented in Table 1 for different QAs.

Table 1

### Parameters of superposition and interference operators of main quantum algorithms

Algorithm	Superposition	$m$	Interference
Deutsch's	$H \otimes I$	1	$H \otimes H$
Deutsch-Jozsa's	${}^n H \otimes H$	1	${}^n H \otimes I$
Grover's	${}^n H \otimes H$	1	$D_n \otimes I$
Simon's	${}^n H \otimes {}^n I$	$n$	${}^n H \otimes {}^n I$
Shor's	${}^n H \otimes {}^n I$	$n$	$QFT_n \otimes {}^n I$

**Superposition and interference** operators are often constructed as tensor powers of the Hadamard operator, which is called the Walsh-Hadamard operator. Elements of the Walsh-Hadamard operator can be obtained as

$$\begin{aligned} [{}^n H]_{i,j} &= \frac{(-1)^{i*j}}{\sqrt{2^n}} [{}^{n-1} H] = \\ &= \frac{1}{\sqrt{2^n}} \begin{pmatrix} {}^{(n-1)} H & {}^{(n-1)} H \\ {}^{(n-1)} H & -{}^{(n-1)} H \end{pmatrix}, \end{aligned} \tag{4}$$

where  $i = 0, 1, j = 0, 1, H$  denotes Hadamard matrix of order 3.

The rule in Eq. (4) provides a way to speed up the classical simulation of the Walsh-Hadamard operators, since the elements of the operator can be obtained by a simple replication described in Eq. (4) from the elements of the  ${}^{n-1}H$  order operator.

As an example, we consider the superposition operator of Grover's algorithm, for  $n = 2, m = 1, S = H$ :

$$\begin{aligned} [Sp]^{Grover's} &= {}^2 H \otimes H = \\ &= \left( \frac{1}{\sqrt{8}} \right)^3 H = \left( \frac{1}{\sqrt{8}} \right) \begin{pmatrix} {}^2 H & {}^2 H \\ {}^2 H & -{}^2 H \end{pmatrix} = \\ &= \left( \frac{1}{\sqrt{8}} \right) \begin{pmatrix} H & H & H & H \\ H & -H & H & -H \\ H & H & -H & -H \\ H & -H & -H & H \end{pmatrix}. \end{aligned} \tag{5}$$

Interference operators are calculated for each algorithm according to the parameters listed in Table 1. The interference operator is based on the interference layer of the algorithm, which is different for various algorithms, and from the measurement layer, which is the same or similar for most algorithms and includes  $m^{\text{th}}$  tensor power of the identity operator.

The interference operator of the Grover's algorithm can be written as a block matrix in the following form:

$$\begin{aligned} [Int^{Grover's}]_{i,j} &= D_n \otimes I = \\ &= \left( \frac{1}{\sqrt{2^n}} - I \right) \otimes I = \left( -1 + \frac{1}{\sqrt{2^n}} \right) \otimes I \Big|_{i,j}, \\ \left( \frac{1}{\sqrt{2^n}} \right) \otimes I \Big|_{i \neq j} &= \frac{1}{\sqrt{2^n}} \begin{cases} -I, & i = j \\ I, & i \neq j \end{cases}, \end{aligned} \tag{6}$$

where  $i = 0, \dots, 2^n - 1, j = 0, \dots, 2^n - 1, D_n$  refers to a diffusion operator:  $[D_n]_{i,j} = \frac{(-1)^{LAND(i=j)}}{\sqrt{2^n}}$ .

For example, the interference operator for Grover's QSA when  $n = 2, m = 1$  is:

$$\begin{aligned} [Int^{Grover's}]_{i,j} &= D_2 \otimes I = \left( \frac{1}{\sqrt{2^2}} - I \right) \otimes I = \\ &= \left( -1 + \frac{1}{2} \right) \otimes I \Big|_{i=j} = \\ &= \begin{pmatrix} \left( -1 + \frac{1}{2} \right) I & \frac{1}{2} I & \frac{1}{2} I & \frac{1}{2} I \\ \frac{1}{2} I & \left( -1 + \frac{1}{2} \right) I & \frac{1}{2} I & \frac{1}{2} I \\ \frac{1}{2} I & \frac{1}{2} I & \left( -1 + \frac{1}{2} \right) I & \frac{1}{2} I \\ \frac{1}{2} I & \frac{1}{2} I & \frac{1}{2} I & \left( -1 + \frac{1}{2} \right) I \end{pmatrix} = \\ &= \frac{1}{2} \begin{pmatrix} -I & I & I & I \\ I & -I & I & I \\ I & I & -I & I \\ I & I & I & -I \end{pmatrix}. \end{aligned} \tag{7}$$

As the number of qubits increases, the gain coefficient becomes smaller. The dimension of the matrix increases according to  $2^n$ , but each element can be extracted using Eq. (6), without allocation of the entire operator matrix.

*Remark.* Since  $D_n D_n^* = I, D_n$  is unitary and is therefore a possible quantum state transformation. While the matrix  $D_n$  is clearly unitary, it can have the decomposition form  $D_n = -H_n R_n^1 H_n$ , where  $R_n^1[i, j] = 0$ , if  $i \neq j, R_n^1[1, 1] = -1$ , and  $R_n^1[i, i] = +1$ , if  $1 < i \leq N$ .

In a specific form the operator  $D_n$  (*diffusion – inversion about average*) in Grover algorithm is decomposed as

$$D_n = \frac{1}{\sqrt{2^n}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}^{\otimes n} \begin{pmatrix} -1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & \dots & 1 \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}^{\otimes n},$$

and can be accomplished with  $O(n) = O(\log(n))$  quantum gates. It means that from the viewpoint of efficient computation the form in Eq. (6) is more preferable.

**An entanglement operator** is a sparse matrix. Using sparse matrix operations, it is possible to accelerate the simulation of the entanglement. Each row or column of an entanglement operation has only one position with a non-zero value. This is a result of the function  $F$  reversibility. For example, let us consider the entanglement operator for a binary function with two inputs and one output:

$$f : \{0,1\}^2 \rightarrow \{0,1\}^1, \text{ such that: } f(x) = 1|_{x=01} 0|_{x \neq 01}.$$

The reversible function  $F$  in this case is:  $F : \{0,1\}^3 \rightarrow \{0,1\}^3$ , such that:

(x, y)	(x, f(x) ⊗ y)
00,0	00,0 ⊗ 0 = 0
00,1	00,0 ⊗ 1 = 1
01,0	01,1 ⊗ 0 = 1
01,1	01,1 ⊗ 1 = 0
10,0	10,0 ⊗ 0 = 0
10,1	10,1 ⊗ 0 = 1
11,0	11,0 ⊗ 0 = 0
11,1	11,1 ⊗ 0 = 1

The corresponding entanglement block matrix can be written as:

$$U_F = \begin{matrix} \langle 00| & \langle 01| & \langle 10| & \langle 11| \\ \begin{matrix} |00\rangle \\ |01\rangle \\ |10\rangle \\ |11\rangle \end{matrix} \end{matrix} \begin{pmatrix} I & 0 & 0 & 0 \\ 0 & \boxed{C} & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & I \end{pmatrix}.$$

Figure 2c shows the result applying this operator in Grover's QSA.

### Command line simulation of the QAs

Let us present an example script of the Grover's algorithm (<http://www.swsys.ru/uploaded/image/2023-3/2023-3-dop/17.jpg>, <http://www.swsys.ru/uploaded/image/2023-3/2023-3-dop/18.jpg>, <http://www.swsys.ru/uploaded/image/2023-3/2023-3-dop/19.jpg>).

The algorithm-related script (<http://www.swsys.ru/uploaded/image/2023-3/2023-3-dop/17.jpg>) prepares the superposition (SP), entanglement (ENT) and interference (INT) operators of the Grover's algorithm with 3 qubits (including the measurement qubit). Then it assembles operators into the quantum gate  $G$ .

Then this script creates an input state  $|in\rangle = |001\rangle$  and calculates the output state  $|out\rangle = G|in\rangle$ . The result of this algorithm in Matlab is an allocation of operator matrices and state vectors in the memory.

Allocated quantum operator matrices, allocated input  $|in\rangle$  and output  $|out\rangle$  state vectors as well as quantum gate  $G$  are available at <http://www.swsys.ru/uploaded/image/2023-3/2023-3-dop/19.jpg>. In order to see the results, we applied visualization functions (<http://www.swsys.ru/uploaded/image/2023-3/2023-3-dop/18.jpg>). The presented code displays operator matrices in 3D visualization Fig. 4.

In this case, the vertical axis corresponds to the amplitudes of the corresponding matrix elements. Indexes of the elements are marked with the ket

notation. Input  $|in\rangle$  and the output  $|out\rangle$  states are demonstrated in Fig. 5. In this case, the vertical axis corresponds to the probability amplitudes of the state vector components. The horizontal axis corresponds to the index of the state vector component marked by the ket notation.

The title of the Fig. 5 contains the values of the Shannon and the von Neumann entropies of the corresponding visualized states.

We can formulate and execute other known QA using similar scripts and the corresponding equations taken from the previous section.

### Simulating QAs as dynamic systems

In order to simulate the behavior of dynamic systems with quantum effects, it is possible to represent QA as a dynamic system in the form of a block diagram and then to simulate its behavior in time. The example of a Simulink diagram of the quantum circuit for calculating the accuracy  $\langle a|a\rangle$  of the quantum state and the density matrix  $|a\rangle\langle a|$  of the quantum state is available at <http://www>.

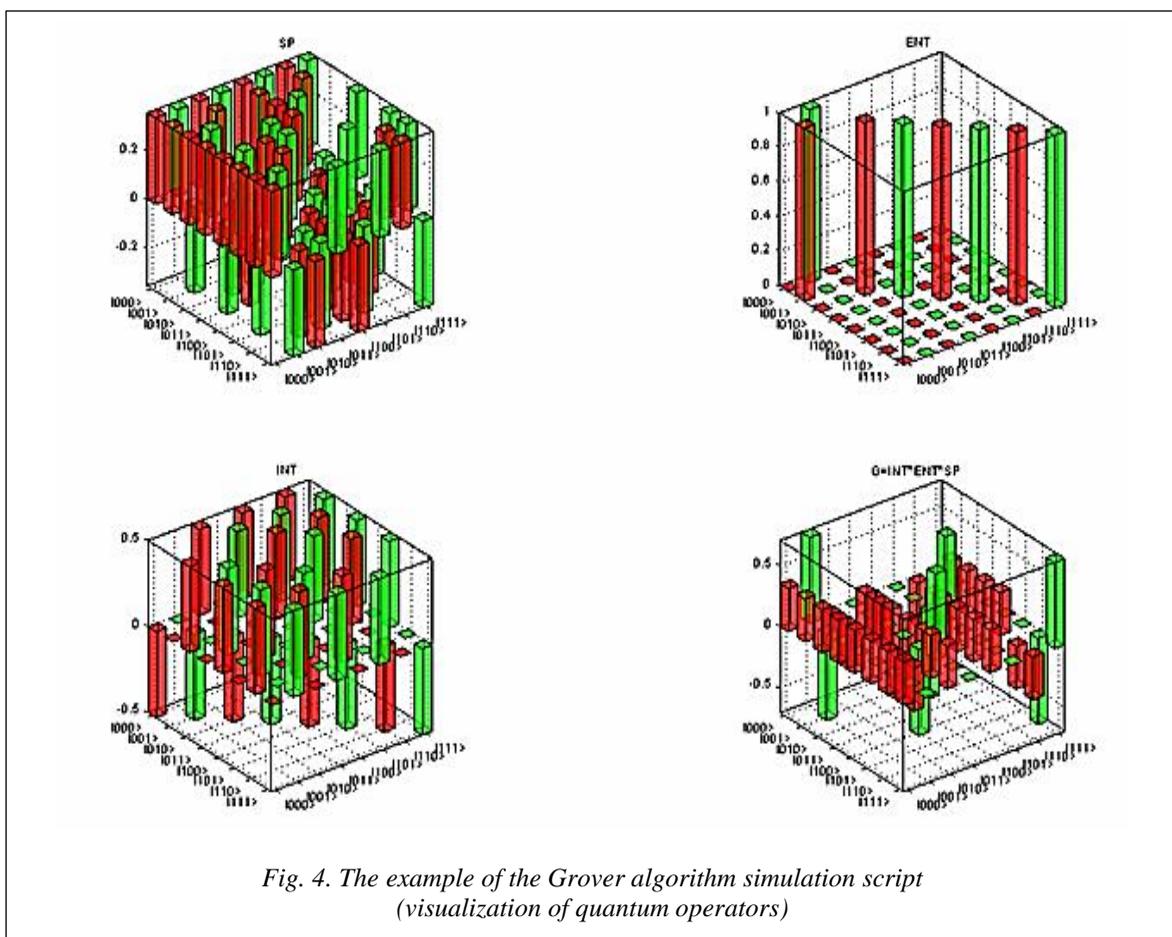


Fig. 4. The example of the Grover algorithm simulation script (visualization of quantum operators)

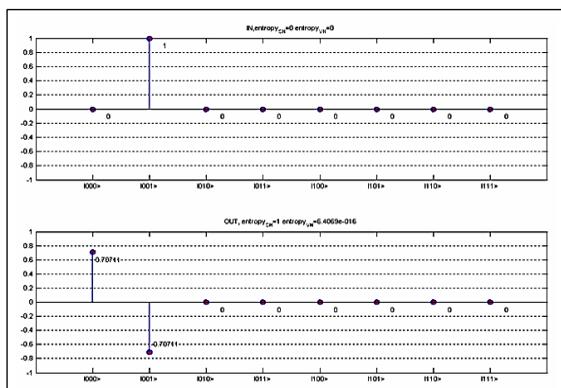


Fig. 5. The example of the Grover algorithm simulation script (visualization of input and output quantum states)

swwsys.ru/uploaded/image/2023-3/2023-3-dop/20.jpg. This example demonstrates using of the common functions for simulating the QA dynamics. Bra and ket functions are taken from the common library.

The ket function output goes to the first input of the matrix multiplier and as a second input of the matrix multiplier. The input also proceeds to the bra function. The output of the bra function goes to the second input of the matrix multiplier and as the first input of the matrix multiplier. The multiplier output is a density matrix of the input state. The multiplier output is the input state fidelity.

Figure 6 shows the Simulink structure of an arbitrary QA.

Such structure can be used to simulate a number of quantum algorithms in Matlab/Simulink environment.

### A dedicated QA emulator

The developed QA algorithmic representation is also applicable for designing QA software emulators. The key point is the reduction of multiple matrix operations to vector operations and the following replacement of multiplication operations. This may boost emulation performance, especially in the algorithms which do not require complex number operations, and when a quantum state vector has a relatively simple structure (for example, Grover’s QSA).

In the QC emulator launch window (<http://www.swwsys.ru/uploaded/image/2023-3/2023-3-dop/21.jpg>), we can choose creating a new QC model or continue modeling an existing one. If we choose creating a new model, then an algorithm selection dialog starts (<http://www.swwsys.ru/uploaded/image/2023-3/2023-3-dop/22.jpg>). Here a user may choose QA and its dimensions.

In fact, the system may operate with up to 50 qubits and more, however due to visualization problems, it is better to limit number of qubits to 10–11.

Once the algorithm initial parameters are set, the system draws an initial state vector and the selected algorithm structure in the system main window (<http://www.swwsys.ru/uploaded/image/2023-3/2023-3-dop/23.jpg>).

The main window contains all information of the emulated quantum algorithm and permits basic operations and analysis. There is an access to involved quantum operators from the menu, and it is possible to modify input functions (<http://www.swwsys.ru/uploaded/image/2023-3/2023-3-dop/>

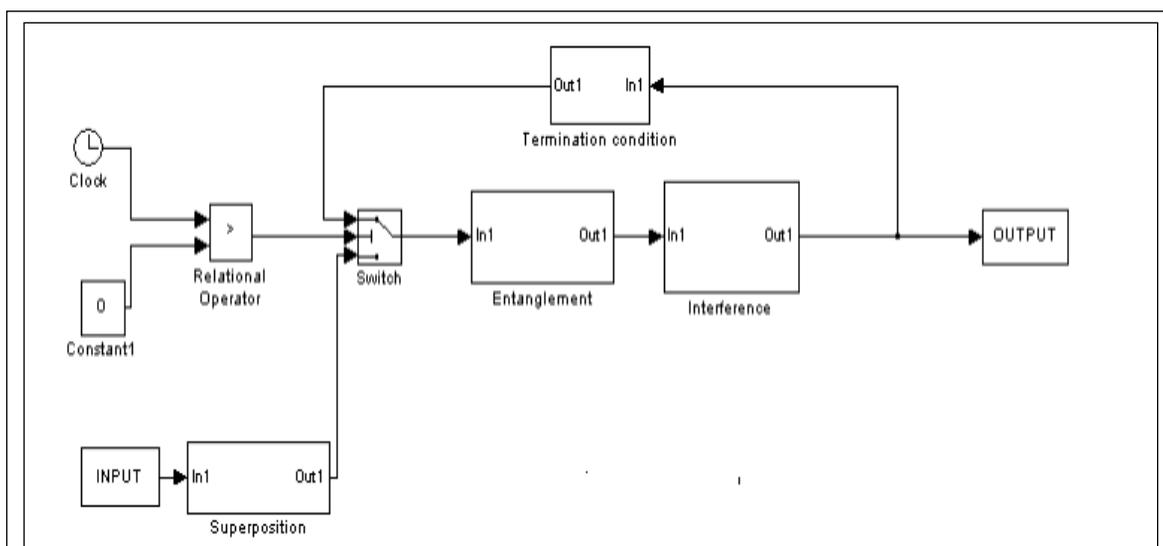


Fig. 6. A Simulink diagram for the simulating the arbitrary quantum algorithm

24.jpg, <http://www.swsys.ru/uploaded/image/2023-3/2023-3-dop/25.jpg>).

QAs have reversible nature, so it is possible to make forward and backward steps of the algorithm by clicking on arrows, and the currently applied algorithm step will be highlighted in the algorithm diagram.

The emulator menu consists of four components:

1. Item *File* provides basic operations like project save/load and an access to the new model creation interface.

2. Item *Model* permits an access to the input function editor.

3. Item *View* provides an access to operator matrix visualizers including Superposition, Entanglement and Interference operators. It is also possible to get a 3D preview of an algorithm state dynamics (Fig. 4).

4. There is an access to the program documentation from Help menu.

Tabbed interface in the lower part of the window gives an access to the Shannon entropy chart and to a 3D representation of the state vector dynamics, as well as to a usual, plain representation of the QA state. The tabbed area size can be modified by dragging a divider. A click on the middle point of divider hides the tabbed area from the screen.

The buttons in the middle part of the main window permit making steps of the currently parameterized QA. As it was mentioned above, the system can make forward and backward steps.

If the algorithm steps were enough, a click on the "!" button will extract an answer from the current state vector.

An appropriate result interpretation routine will be called depending on QA.

The quantum operator visualizer permits displaying a structure of involved quantum operator matrices in plain and in 3D representations.

If an operator consists of a tensor product of smaller operators, there is also a possibility to have an access to sub-blocks of the tensor products. A 3D visualizer permits zoom and rotation of the charts.

The input function editor enables automating the process of the entanglement operator coding as it was described previously. For Grover's QSA it is possible to code functions with more than one positive output (Fig. 7).

Figures 8, 9 show the results of Grover QSA simulation with entropy criteria termination.

The developed software can simulate 4 basic quantum algorithms, e.g. Deutsch-Jozsa's, Shor's,

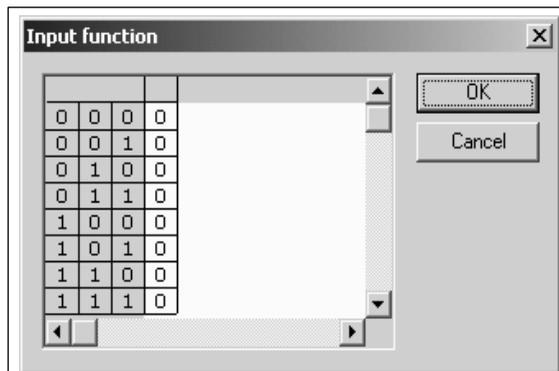


Fig. 7. An input function editor of QA emulator (3-Qubit Grover's QSA)

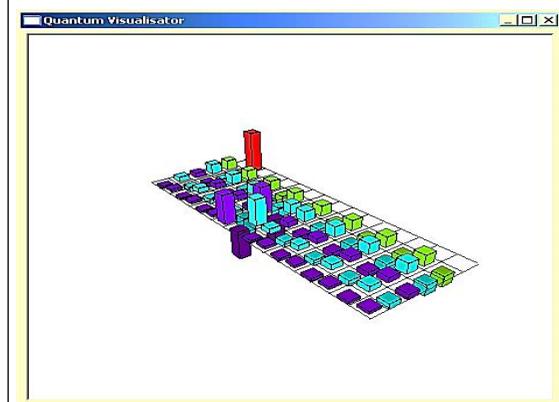


Fig. 8. A 3D view of the 3-qubit Grover's QSA state vector after two algorithm iterations

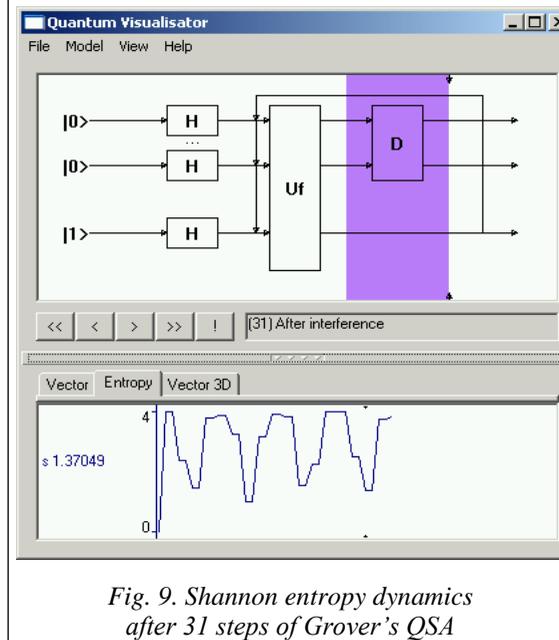


Fig. 9. Shannon entropy dynamics after 31 steps of Grover's QSA

Simon's and Grover's. The system uses a unified easy-to-understand interface for all algorithms, with the options of 3D visualization of state vector dynamics and quantum operators.

After analyzing quantum operators presented in the section 5, we can do the following simplification to increase the performance of classical QA simulations: a) all quantum operators are symmetrical around main diagonal matrices; b) a state vector is allocated as a sparse matrix; c) elements of quantum operators are not stored, but calculated when necessary using Eqs. (4), (5), (6) and (7); d) we consider minimum of Shannon entropy of the quantum state as a termination condition, calculated as:

$$H^{Sh} = -\sum_{i=0}^{2^{m+n}} p_i \log p_i . \tag{8}$$

The calculation of the Shannon entropy is applied to a quantum state after the interference operation [6, 7].

### The results of a classical quantum algorithmic gate simulation

Minimum of the Shannon entropy Eq. (8) corresponds to the state when there are few state vectors with high probability (states with minimum uncertainty). Selecting an appropriate termination condition is important since QAs are periodical. Figure 10 shows results of the Shannon information entropy calculation for Grover’s algorithm with 5 inputs.

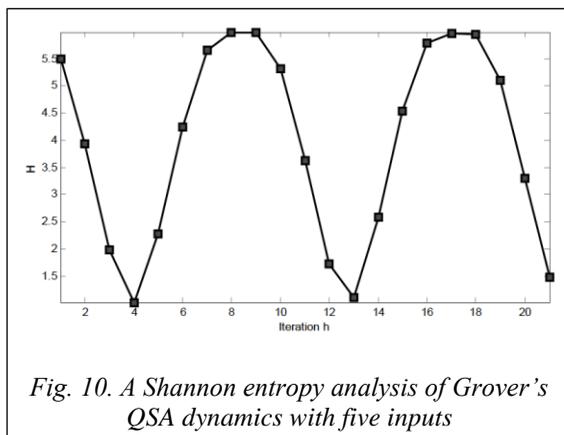


Fig. 10. A Shannon entropy analysis of Grover’s QSA dynamics with five inputs

Figure 10 shows that for five inputs of Grover’s QSA, an optimal number of iterations according to minimum of the Shannon entropy criteria for successful result is exactly four. After that, the probability of a correct answer decreases and algorithm may fail to produce a correct answer. Note that theoretical estimation for 5 inputs gives  $\frac{\pi}{4}\sqrt{2^5} = 4.44$  iterations.

Simulation results of fast Grover QSA are summarized in Table 2.

Table 2  
Temporal complexity of Grover’s QSA simulation on 1.2GHz computer with two CPUs

No.	Iterations number h	Temporal complexity, s	
		Approach 1 (one iteration)	Approach 2 (h iterations)
10	25	0.28	~0
12	50	5.44	~0
14	100	99.42	~0
15	142	489.05	~0
16	201	2060.63	~0
20	804	-	~0
30	25 375	-	0.016
40	853 549	-	4.263
50	26 353 589	-	12.425

Numbers of iterations for fast algorithm were estimated according to termination condition as the minimum of the Shannon entropy of a quantum state vector. The simulation involved the following approaches:

*Approach 1:* Quantum operators are applied as matrices; elements of quantum operator matrices are calculated dynamically according to Eqs. (5), (6), and (7). Classical Hardware limit of this approach is around 20 qubits caused by exponential temporal complexity.

*Approach 2:* Quantum operators are replaced with classical gates. Product operations are removed from simulation according to [8]. A state vector of probability amplitudes is stored in a compressed form (only different probability amplitudes are allocated in memory). The second approach makes it possible to perform classical efficient simulation of Grover’s QSA with an arbitrary large number of inputs (50 qubits and more).

When allocating the state vector in computer memory, this approach permits simulating 26 qubits on PC with 1GB of RAM. Figure 11 shows memory required for Grover algorithm simulation, when the whole state vector is allocated in memory.

Adding one qubit requires doubling the computer memory needed for simulating Grover’s QSA in case when a state vector is allocated completely in memory.

Temporal complexity of Grover’s QSA is presented in Fig. 12.

In this case, the state vector is allocated in memory and quantum operators are replaced with classical gates according to [8]. The fastest case is when we compress the state vector and replace quantum operator matrices with corresponding

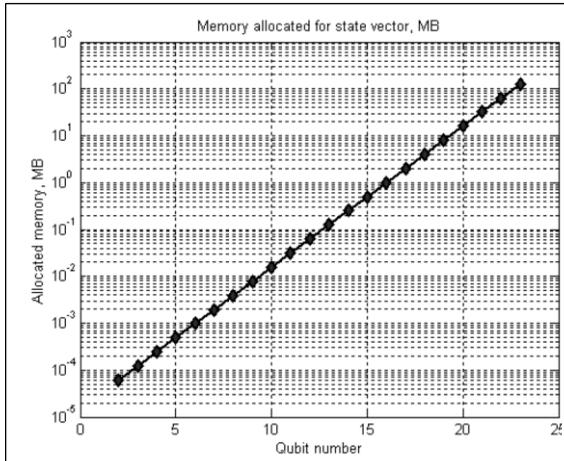


Fig. 11. Spatial complexity of Grover QA simulation

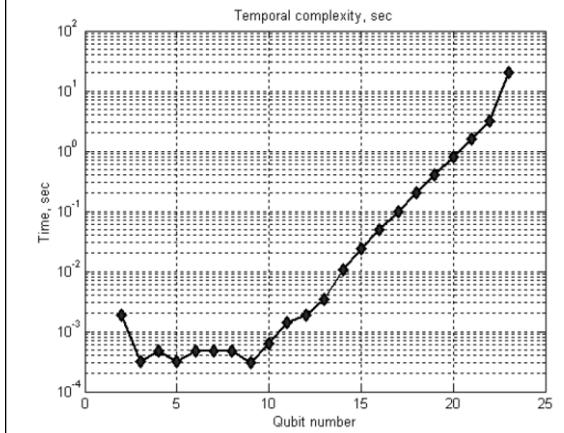


Fig. 12. Temporal complexity of Grover's QSA

classical gates according to [8]. In this case, we obtain speedup according to Approach 2.

**Fast QSA models: The structure and acceleration method of quantum algorithm simulation**

The analysis of the quantum operator matrices carried out in the previous sections forms the basis for specifying structural patterns that give the background for the algorithmic approach to QA modeling on classical computers. Allocating only a fixed set of tabulated (pre-defined) constant values in the computer memory *instead* of allocating huge matrices (even in sparse form) provides computational efficiency. Various elements of the quantum operator matrix can be obtained by applying an appropriate algorithm based on structural patterns and particular properties of the equations that define matrix elements. Each representation algorithm uses a set of table values for calculating

matrix elements. Calculation of the tables of the predefined values can be done as a part of the algorithm initialization.

*The algorithmic representation of the Grover's QA.* Figures 13a–c are flowcharts showing implementation of such approach for simulating superposition (Fig. 13a), entanglement (Fig. 13b) and interference (Fig. 13c) operators in Grover's QSA.

Here  $n$  is a number of qubits,  $i$  and  $j$  are the indexes of a requested element,  $hc = 2^{-(n+1)/2}$ ,  $dc1 = 2^{1-n} - 1$  and  $dc2 = 2^{1-n}$  are table values.

In Fig. 13a, the  $i, j$  values are specified and provided to an initialization block with loops control variables  $ii := i, jj := 0$ , and  $k := 0$  are initialized,

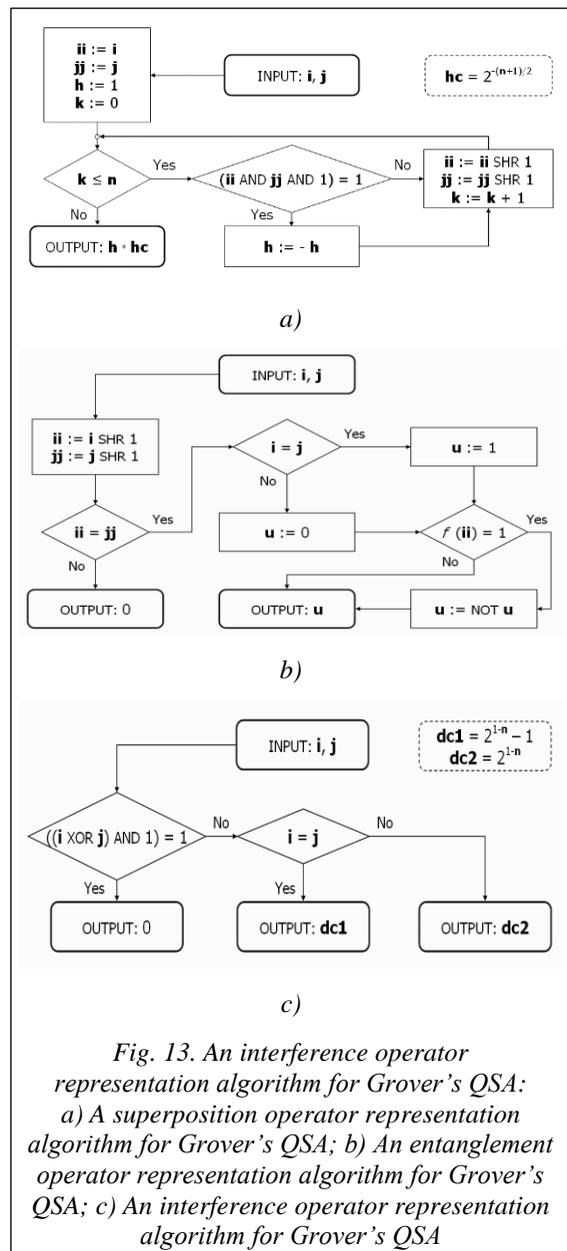


Fig. 13. An interference operator representation algorithm for Grover's QSA: a) A superposition operator representation algorithm for Grover's QSA; b) An entanglement operator representation algorithm for Grover's QSA; c) An interference operator representation algorithm for Grover's QSA

and a calculation variable  $h := 1$  is initialized. Then the process moves to a decision block. In the decision block, if  $k$  is less than or equal to  $n$ , then the process advances to another decision block; otherwise, the process advances to an output block where the output  $h*hc$  is computed (where  $hc = 2^{-(n+1)/2}$ ).

In the decision block, if  $(ii$  and  $jj$  and  $1) = 1$ , then the process advances to a block  $h := -h$ ; otherwise, the process advances to another block and passes to the next iteration without probability amplitude inversion. Alternatively, the process sets  $h := -h$  and proceeds to the next iteration. By setting  $ii := ii$  SHR 1,  $jj := jj$  SHR 1, and  $k := k + 1$  (where SHR is a shift right operation), and then the process continues until all probability amplitudes are assigned.

In Fig. 13, the inputs  $i, j$  in an input block are initialized as  $ii := i$  SHR 1, and  $jj := SHR$  1 and then are passed to the end test.

If the end test fail, it means that the inputs  $i$  and  $j$  are pointing to the marked elements; in this case the process of the probability amplitude inversion of the marked states is performed.

In Fig. 13c, the interference operator of Grover’s QSA can be substituted by a simple logic algorithm that outputs 0 if  $((i$  XOR  $j)$  AND  $1) = 1$ . Then regarding nonzero elements, if  $i = j$  then  $dc1$  outputs the process, otherwise  $dc2$ , where  $dc1 = 2^{1-n} - 1$  and  $dc2 = 2^{1-n}$  output the process.

The superposition and entanglement operators for Deutsch-Jozsa’s QA are the same as superposition and entanglement operators for Grover’s QSA (Figs 13a,b), respectively).

The time required for calculating the elements of an operator’s matrix during a process of applying a quantum operator is generally small in comparison to the total time of performing a quantum step. Thus, the live load created by exponentially increasing memory usage tends to be less or at least similar to the live load created by computing matrix elements as needed. Moreover, since the algorithms for computing matrix elements tend to be based on fast bit-wise logic operations, algorithms are amenable to hardware acceleration.

Table 3 shows comparisons of the traditional and required matrix calculation when the memory is used for the algorithm as required (*Memory\** is memory used for storing the quantum system state vector).

Table 3 shows that the algorithmic approach enables a significant speed-up compared with the prior art direct matrix approach. The use of algorithms for providing matrix elements allows considerable software optimization, including the

ability to optimize at the machine instruction level. However, as the number of qubits increases, there is an exponential increase in temporal complexity, which shows itself as an increase in time required for matrix product calculations.

Table 3

**Different approaches comparison:  
Standard (matrix based)  
and algorithmic-based approach**

Qubits	Standard		Calculated Matrices	
	Memory, MB	Time, s	Memory *	Time, s
1	1	0.03	≈0	≈0
8	18	3.4	0.008	0.0325
11	1048	1411	0.064	3.3
16	–	–	2	4573
24	–	–	512	3·10 <sup>8</sup>
64	–	–	–	–

\* The results shown in Table 3 are based on the results of testing the software implementation of the Grover QSA simulator on a personal computer with Intel Pentium III 1 GHz processor and 512 Mb memory. Only one iteration of the Grover QSA was performed.

Using structural patterns in the quantum system state vector and a problem-oriented approach for each particular algorithm can compensate this increase in temporal complexity. By way of explanation and not by way of limitation, the Grover algorithm is used below to explain the problem-oriented approach to simulating QA on a classical computer.

**The problem-oriented approach based on a structural pattern of QA state vector**

Let  $n$  be the input number of qubits. In the Grover algorithm, half of all  $2^{n+1}$  elements of a vector making up its even components always take values symmetrical to appropriate odd components and, therefore, it does not need to be computed.

Odd  $2^n$  elements can be classified into two categories:

- A set of  $m$  elements corresponding to truth points of an input function (or oracle);
- The remaining  $2^n - m$  elements.

The values of the same category elements are always equal.

As discussed above, Grover’s QA only requires two variables for storing element values. Its limitation in this sense depends only on a computer representation of the floating-point numbers used

for state vector probability amplitudes. For a double-precision software implementation of the state vector representation algorithm, the upper reachable limit of q-bit number is approximately 1024.

Figure 14 shows a state vector representation algorithm for Grover's QA.

In Fig. 14  $i$  is an element index,  $f$  is an input function,  $vx$  and  $va$  correspond to the element category, and  $v$  is a temporal variable. The number of variables for representing a state variable is constant. A constant number of variables for state vector representation allows reconsidering the traditional schema of quantum search simulation.

Classical gates are used not for simulating appropriate quantum operators with strict one-to-one correspondence but for simulating a quantum step that changes the system state. Matrix product operations are replaced by arithmetic operations with a fixed number of parameters irrespective of a qubit number.

Figure 15 shows a generalized schema for efficient simulation of the Grover QA built upon three blocks, a superposition block  $H$ , a quantum step block  $UD$  and a termination block  $T$ .

Figure 15 also shows input and output blocks. A  $UD$  block includes a  $U$  block and a  $D$  block. The input state from the input block transfers to the superposition block. A superposition of states from the superposition block transfers to the  $U$  block. An output from the  $U$  block transfers to the  $D$  block. An output from the  $D$  block transfers to the termination block. If the termination block terminates iterations, then the state is passed to the output block; otherwise, the state vector is returned to the  $U$  block for iteration.

As shown in Fig. 16, the superposition block  $H$  for Grover's QSA simulation changes the system state to the state obtained traditionally by using  $n + 1$  times the tensor product of Walsh-Hadamard transformations. In the process shown in Fig. 13,  $vx := hc$ ,  $va := hc$ , and  $vi := 0$ , where  $hc = 2^{-(n+1)/2}$  is a table value.

The quantum step block  $UD$  that emulates the entanglement and interference operators is shown on Figs 17a–c.

The  $UD$  block reduces the temporal complexity of the quantum algorithm simulation to a linear dependence on the number of executed iterations.

The  $UD$  block uses recalculated table values  $dc1 = 2^{1-n} - 1$  and  $dc2 = 2^{1-n}$ .

In the  $U$  block shown in Fig. 17,  $vx := -vx$  and  $vi := vi + 1$ .

In the  $D$  block shown in Fig. 17b,  $v := m*vx+dc1*va$ ,  $v := v/dc2$ ,  $vx := v - vx$ , and

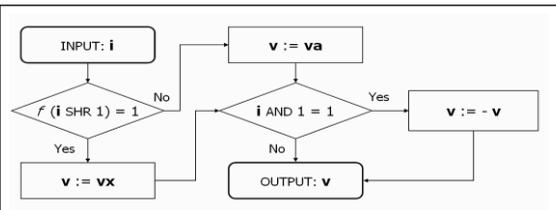


Fig. 14. A state vector representation algorithm for Grover's quantum search

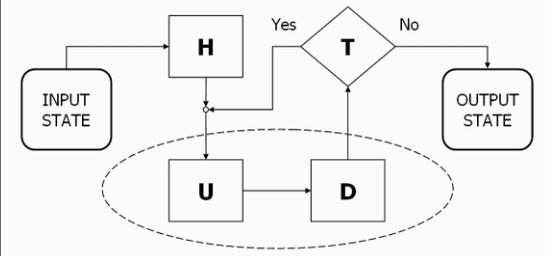


Fig. 15. A generalized simulation schema for Grover's QSA

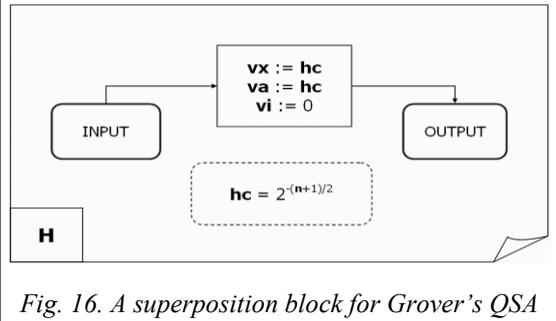
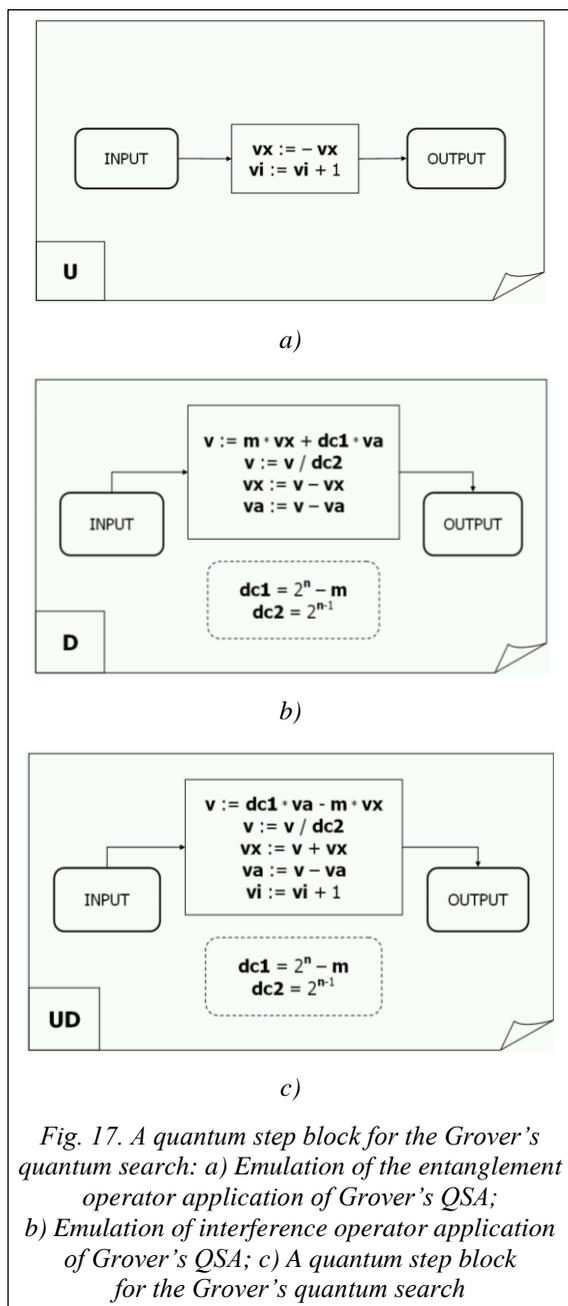


Fig. 16. A superposition block for Grover's QSA

$va := v - va$ ; in the  $UD$  block shown in Fig. 17c,  $v := dc1*va = m*vx$ ,  $v := v/dc2$ ,  $vx := v + vx$ ,  $va := v - va$ , and  $vi := vi + 1$ .

The termination block  $T$  is general for all QAs without regard to the operator matrix realization. Block  $T$  provides *intelligent termination condition* for a search process. Thus, the block  $T$  controls the number of iterations through the block  $UD$  by providing enough iteration to achieve a high probability of arriving at a correct answer to the search problem. The block  $T$  uses a rule based on observing the changing of the vector element values according to two classification categories. During a number of iterations,  $T$  block check that the values of same category elements monotonically increase or decrease while values of other category elements changed monotonically in reverse direction. If the direction is changed after some number of iterations, it means that an extremum point corresponding to a state with maximum or minimum uncertainty is passed. The process can use direct values of amplitudes instead of considering Shannon entropy value, thus, it significantly reduces the re-



quired number of calculations for determining the minimum uncertainty state that guarantees the high probability of a correct answer.

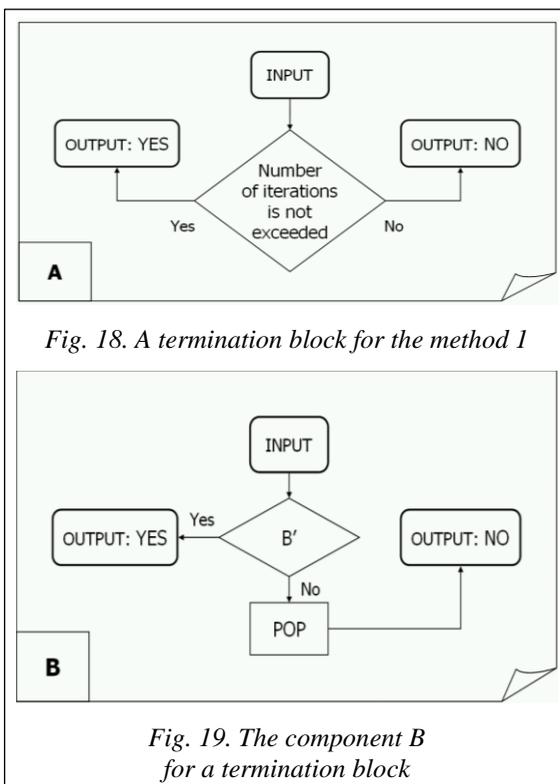
The termination algorithm implemented in *T* block can use one or more of five different termination models:

*Model 1:* Stop after a predefined number of iterations;

*Model 2:* Stop on the first local entropy minimum;

*Model 3:* Stop on the lowest entropy within a predefined number of iterations;

*Model 4:* Stop on a predefined level of acceptable entropy; and/or



*Model 5:* Stop on the acceptable level or lowest reachable entropy within the predefined number of iterations.

Note that models 1–3 do not require calculating an entropy value.

Figures 18–20 show the structure of termination condition blocks *T*.

Since time efficiency is one of the major demands on such termination condition algorithm, a separate module represents each part of the termination algorithm; and before the termination algorithm starts, links are built between the modules in correspondence to the selected termination model by initializing the appropriate functions' calls.

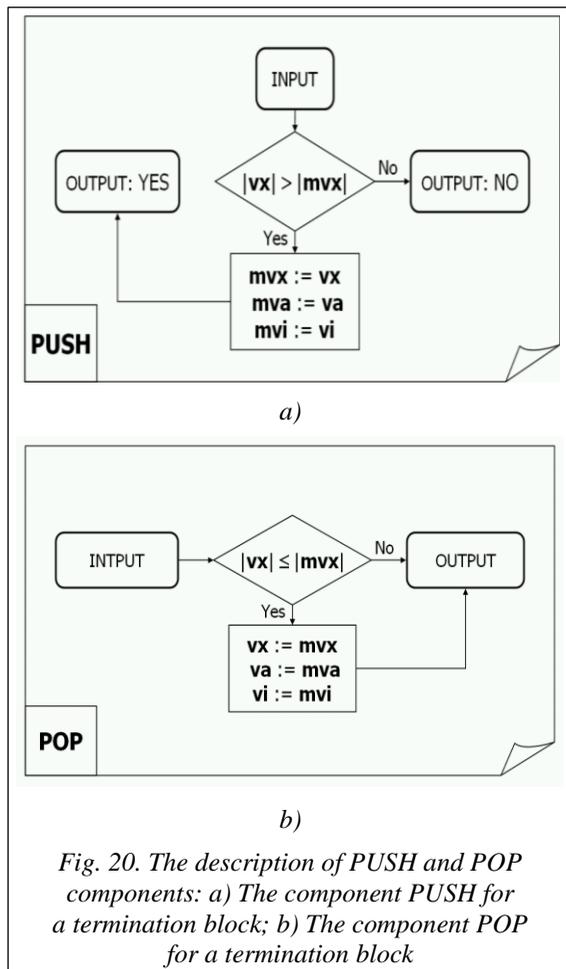
Table 4 shows components for the termination condition block *T* for the various models.

Table 4

**Termination block construction**

Model	T	B'	C'
1	A	–	–
2	B	PUSH	–
3	C	A	B
4	D	–	–
5	C	A	E

The elements *A*, *B*, *PUSH*, *C*, *D*, *E*, and *PUSH*-code in Table 4 correspond to the flowcharts (<http://www.swsys.ru/uploaded/image/2023-3/2023-3-dop/26.jpg>, <http://www.swsys.ru/uploaded/>



image/2023-3/2023-3-dop/27.jpg, <http://www.swsys.ru/uploaded/image/2023-3/2023-3-dop/28.jpg>).

In *model 1* requires only one test after each application of quantum step block *UD*, block *A* performs this test. Therefore, the initialization includes assuming *A* to be *T*, i.e., function calls to *T* are addressed to block *A* shown in Fig. 18.

As shown in Fig. 18, *A* block checks if the maximum number of iterations has been reached, if so, then the simulation is terminated, otherwise the simulation continues.

In *model 2*, simulation stops when the direction of category value modification is changed. Model 2 uses the comparison of the current value of *vx* category with *mvx* value that represents this category value obtained in previous iteration:

I. If *vx* is greater than *mvx*, its value is stored in *mvx*, *vi* value is stored in *mvi*, and the termination block proceeds to the next quantum step;

II. If *vx* is less than *mvx*, it means that *vx* maximum is passed and the process needs to set the current (final) value of  $vx := mvx$ ,  $vi := mvi$ , and stop the iteration process. So, the process stores the

maximum of *vx* in *mvx* and the appropriate iteration number *vi* in *mvi*. Here block *B*, shown in Fig. 19 is used as the main block of the termination process.

The block *PUSH* shown in the Fig. 20a is used for performing a comparison and for storing the *vx* value in *mvx* (*case a*). A *POP* block shown in Fig. 20b is used for restoring the *mvx* value (*case b*). In the *PUSH* block in Fig. 20a, if  $|vx| > |mvx|$ , then  $mvx := vx$ ,  $mva := va$ ,  $mvi := vi$ , and the block returns true; otherwise, the block returns false.

In the *POP* block in Fig. 17b, if  $|vx| \leq |mvx|$ , then  $vx := mvx$ ,  $va := mva$ , and  $vi := mvi$ .

The *model 3* termination block checks to see that a predefined number of iterations do not exceed (using block *A* in Fig. 20):

- If the check is successful, then the termination block compares the current value of *vx* with *mvx*. If *mvx* is less than, it sets the value of *mvx* equal to *vx* and the value of *mvi* equal to *vi*. If *mvx* is less using the *PUSH* block, then perform the next quantum step;

- If the check operation fails, then (if needed) the final value of *vx* equal to *mvx*, *vi* equal to *mvi* (using the *POP* block) and the iterations stop.

The *model 4*, the termination block uses a single component block *D* (<http://www.swsys.ru/uploaded/image/2023-3/2023-3-dop/27.jpg>).

The *D* block compares the current Shannon entropy value with a predefined acceptable level. If the current Shannon entropy is less than the acceptable level, then the iteration process is stopped; otherwise, the iterations continue.

The *model 5* termination block uses the *A* block to check that a predefined number of iterations do not exceeded. If the maximum number is exceeded, then the iterations are stopped. Otherwise, the *D* block is then used to compare the current value of the Shannon entropy with the predefined acceptable level. If an acceptable level is not achieved, then the *PUSH* block is called and the iterations continue. After the last performed iteration, the *POP* block is called to restore the *vx* category maximum and appropriate *vi* number and the iterations end.

Figure 21 shows measurement of the final amplitudes in the output state to determine success or failure of the search.

If  $|vx| > |va|$ , then the search was successful; otherwise the search was not successful.

Table 5 lists the results of testing the optimized version of Grover QSA simulator on a personal computer with Pentium 4 processor at 2GHz.

Using the above algorithm, a simulation of a 1000 qubit Grover QSA requires only 96 seconds for  $10^8$  iterations (<http://www.swsys.ru/uploaded/image/2023-3/2023-3-dop/29.jpg>).

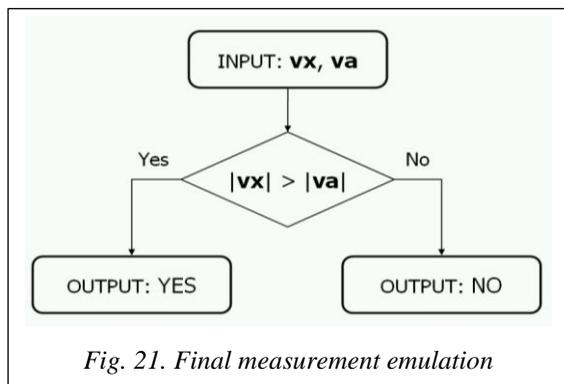


Fig. 21. Final measurement emulation

The theoretical boundary of this approach is not the number of qubits, but the representation of the floating-point numbers.

Table 5

**High probability answers for Grover QSA**

Qubits	Iterations	Time
32	51471	0.007
36	205887	0.018
40	823549	0.077
44	3294198	0.367
48	13176794	1.385
52	52707178	3.267
56	210828712	20.308
60	843314834	81.529
64	3373259064	328.274

The practical bound is limited by the front side bus frequency of a personal computer.

*Related works.* The presented approach was firstly suggested in [9–11] for efficient simulation

of quantum algorithms on classical computers with minimum Shannon entropy measure of termination of searching processes [7] and differ from results in [12–16] and [17–20].

**Conclusions**

In this paper we have: presented a design method of a modular system for realization of the Grover’s Quantum Search Algorithm; developed a design process of main quantum operators with algorithmic description for quantum algorithm gates simulation on a classical computer.

We also have: introduced model representations of quantum operators in fast QAs; described an algorithmic based approach when matrix elements are calculated on demand; demonstrated a problem-oriented approach, where we succeeded to run Grover’s algorithm with up to 64 and more qubits with Shannon entropy calculation (up to 1024 without termination condition) and considered it as a solution of a classically intractable problem.

These results are the background for efficient simulation of quantum soft computing algorithms on a classical computer, robust fuzzy control based on quantum genetic (evolutionary) algorithms and quantum fuzzy neural networks (that can implemented as modified Grover’s QSA), AI-problems as quantum game’s gate simulation approaches and quantum learning, quantum associative memory, quantum optimization.

**Reference List**

- Nielsen, M.A., Chuang, I.L. (2000) *Quantum Computation and Quantum Information*. UK: Cambridge University Press, 676 p.
- Serrano, M.A., Perez-Castillo, R., Piattini, M. (2022) *Quantum Software Engineering*. Springer Verlag Publ., 330 p.
- Korenkov, V.V., Reshetnikov, A.G., Ulyanov, S.V. (2022) *Quantum Software Engineering. Vol. 2*. Moscow, 452 p.
- Grover, L.K. (2001) *A Fast Quantum Mechanical Algorithms*, US, Pat. 6,317,766 B1.
- Ivancova, O.V., Korenkov, V.V., Ulyanov, S.V. (2020) *Intelligent Computing Technologies. Pt. 2. Quantum Computing and Algorithms. Quantum Self-Organization Algorithm. Quantum Fuzzy Inference*. Moscow: Kurs Publ., 296 p.
- Ulyanov, S.V., Panfilov, S.A., Kurawaki, I., Yazenin, A.V. (2001) ‘Information analysis of quantum gates for simulation of quantum algorithms on classical computers’, in *QCM&C*, pp. 207–214. doi: 10.1007/0-306-47114-0\_32.
- Ghisi, F., Ulyanov, S.V. (2000) ‘The information role of entanglement and interference in Shor quantum algorithm gate dynamics’, *J. of Modern Optics*, 47(12), pp. 2079–2090. doi: 10.1080/09500340008235130.
- Amato, P., Ulyanov, S., Porto, D., Rizzotto, G.G. (2003) ‘Hardware architecture system design of quantum algorithm gates for efficient simulation on classical computers’, *Proc. SCI*, 3, pp. 398–403.
- Panfilov, S.A., Ulyanov, S.V., Litvintseva, L.V., Yazenin, A.V. (2004) ‘Fast algorithm for efficient simulation of quantum algorithm gates on classical computer’, *Systemics, Cybernetics and Informatics*, 2(3), pp. 63–68.
- Ulyanov, S.V. (2003) *System and Method for Control Using Quantum Soft Computing*, US, Pat. 6,578,018 B1.
- Ulyanov, S.V., Panfilov, S.A. (2006) *Efficient Simulation of Quantum Algorithm Gates on Classical Computer Based on Fast Algorithm*, US, Pat. 2006/0224.547 A1.
- Nyman, P. (2007) ‘Simulation of quantum algorithms with a symbolic programming language’, *ArXiv*, art. 0705.3333v2, available at: <https://arxiv.org/abs/0705.3333v2> (accessed November 18, 2022).
- Juliá-Díaz, B., Burdis, J.M., Tabakin, F. (2009) ‘QDENSITY – A Mathematica quantum computer simulation’, *CPC*, 180(3), art. 474. doi: 10.1016/j.cpc.2008.10.006.
- Cumming, R., Thomas, T. (2022) ‘Using a quantum computer to solve a real-world problem – what can be achieved today?’, *ArXiv*, art. 2211.13080v1, available at: <https://arxiv.org/abs/2211.13080v1> (accessed November 18, 2022).

15. Ovide, A., Rodrigo, S., Bandic, M., Van Someren, H., Feld, S. et al. (2023) 'Mapping quantum algorithms to multi-core quantum computing architectures', *ArXiv*, art. 20232303.16125v1, available at: <https://arxiv.org/pdf/2303.16125.pdf> (accessed November 18, 2022).
16. Abhijith, J., Adedoyin, A., Ambrosiano, J., Anisimov, P. et al. (2022) 'Quantum algorithm implementations for beginners', *ArXiv*, art. 1804.03719v3, available at: <https://arxiv.org/abs/1804.03719v3> (accessed June 27, 2022).
17. Tezuka, H., Nakaji, K., Satoh, T., Yamamoto, N. (2021) 'Grover search revisited; application to image pattern matching', *ArXiv*, art. 2108.10854v2, available at: <https://arxiv.org/abs/2108.10854v2> (accessed Oct 1, 2021).
18. Vlastic, A., Certo, S., Pham, A. (2022) 'Complement Grover's search algorithm: An amplitude suppression implementation', *ArXiv*, art. 2209.10484v1, available at: <https://arxiv.org/abs/2209.10484> (accessed September 27, 2022).
19. Chattopadhyay, A., Menon, V. (2021) 'Fast simulation of Grover's quantum search on classical computer', *ArXiv*, art. 2005.04635, available at: <https://arxiv.org/pdf/2005.04635.pdf> (accessed September 27, 2022).
20. Toffano, Z., Dubois, F. (2020) 'Adapting logic to physics: The quantum-like eigenlogic program', *Entropy*, 22(2), art. 139. doi: 10.3390/e22020139.

УДК 512.6, 517.9, 519.6

doi: 10.15827/0236-235X.142.361-377

2023. Т. 36. № 3. С. 361–377

### Моделирование быстрого алгоритма квантового поиска на классических компьютерах: информационный анализ проблемы останова

С.В. Ульянов  
В.С. Ульянов

#### Ссылка для цитирования

Ульянов С.В., Ульянов В.С. Моделирование быстрого алгоритма квантового поиска на классических компьютерах: информационный анализ проблемы останова // Программные продукты и системы. 2023. Т. 36. № 3. С. 361–377. doi: 10.15827/0236-235X.142.361-377

#### Информация о статье

Поступила в редакцию: 31.03.2023

После доработки: 19.06.2023

Принята к публикации: 01.07.2023

**Аннотация.** Описана методика модификации моделирования квантового алгоритма, основанная на прямом (большого объема) матричном представлении квантовых операторов. Этот подход стабилен и точен, но требует размещения матриц оператора в памяти компьютера. Поскольку размер операторов растет экспоненциально, подход полезен для моделирования квантовых алгоритмов с относительно небольшим количеством кубитов (например, приблизительно 11 кубитов на типовом компьютере). Используя его, относительно просто смоделировать работу системы контроля качества решения и выполнить анализ достоверности. Более эффективный метод быстрого моделирования контроля качества основан на вычислении всех или части операторных матриц по мере необходимости на текущей вычислительной основе. Используя данный метод, можно избежать сохранения всех или части операторных матриц. В этом случае количество кубитов, которые могут быть смоделированы (например, количество входных кубитов или количество кубитов в регистре состояния системы), влияет на экспоненциальный рост числа операций, необходимых для вычисления результата матричных произведений, и на размер вектора состояния, выделяемого в памяти компьютера. В одном из вариантов применения этого подхода целесообразно моделировать до 19 или более кубитов на типичном настольном компьютере и даже больше на системе с векторной архитектурой. Из-за особенностей процессов адресации памяти и доступа к ней в типичном настольном компьютере (например, персональный компьютер на базе Pentium), когда количество кубитов относительно невелико, подход «вычисления по требованию», как правило, эффективнее, чем подход с прямым хранением. Подход «вычисления по требованию» выигрывает благодаря применению результатов детального изучения квантовых операторов и их структуры, что позволяет более эффективно вычислять матричные элементы. В работе рассмотрено эффективное моделирование алгоритма квантового поиска Гровера на примере компьютера с классической архитектурой.

**Ключевые слова:** алгоритм быстрого квантового поиска, квантовые операторы, вектор состояния, алгоритмическое представление, квантовые вычисления, минимум информационной энтропии Шеннона, критерии завершения **Благодарности.** Плодотворная дискуссия с П. Шором, Ч. Беннеттом, Л. Левитиным и В. Белавкиным помогла авторам сформулировать более четкие результаты для приложений квантовой программной инженерии

## Список литературы

1. Nielsen M.A., Chuang I.L. Quantum Computation and Quantum Information. UK, Cambridge University Press, 2000, 676 p.
2. Serrano M.A., Perez-Castillo R., Piattini M. Quantum Software Engineering. Springer Verlag Publ., 2022, 330 p.
3. Korenkov V.V., Reshetnikov A.G., Ulyanov S.V. Quantum Software Engineering. Moscow, 2022, vol. 2, 452 p.
4. Grover L.K. A Fast Quantum Mechanical Algorithms, US, Pat. 6,317,766 B1, 2001.
5. Иванцова О.В., Кореньков В.В., Ульянов С.В. Технологии интеллектуальных вычислений. Ч. 2. Квантовые вычисления и алгоритмы. Квантовый алгоритм самоорганизации. Квантовый нечеткий вывод. М., 2020, 296 с.
6. Ulyanov S.V., Panfilov S.A., Kurawaki I., Yazenin A.V. Information analysis of quantum gates for simulation of quantum algorithms on classical computers. In: QCM&C, 2001, pp. 207–214. doi: 10.1007/0-306-47114-0\_32.
7. Ghisi F., Ulyanov S.V. The information role of entanglement and interference in Shor quantum algorithm gate dynamics. J. of Modern Optics, 2000, vol. 47, no. 12, pp. 2079–2090. doi: 10.1080/09500340008235130.
8. Amato P., Ulyanov S., Porto D., Rizzotto G.G. Hardware architecture system design of quantum algorithm gates for efficient simulation on classical computers. Proc. SCI, 2003, vol. 3, pp. 398–403.
9. Panfilov S.A., Ulyanov S.V., Litvintseva L.V., Yazenin A.V. Fast algorithm for efficient simulation of quantum algorithm gates on classical computer. Systemics, Cybernetics and Informatics, 2004, vol. 2, no. 3, pp. 63–68.
10. Ulyanov S.V. System and Method for Control Using Quantum Soft Computing, US, Pat. 6,578,018 B1, 2003.
11. Ulyanov S.V., Panfilov S.A. Efficient Simulation of Quantum Algorithm Gates on Classical Computer Based on Fast Algorithm, US, Pat. 2006/0224.547 A1, 2006.
12. Nyman P. Simulation of quantum algorithms with a symbolic programming language. ArXiv, 2007, art. 0705.3333v2. URL: <https://arxiv.org/abs/0705.3333v2> (дата обращения: 18.11.2022).
13. Juliá-Díaz B., Burdis J.M., Tabakin F. QDENSITY – A Mathematica quantum computer simulation. CPC, 2009, vol. 180, no. 3, art. 474. doi: 10.1016/j.cpc.2008.10.006.
14. Cumming R., Thomas T. Using a quantum computer to solve a real-world problem – what can be achieved today? ArXiv, 2022, art. 2211.13080v1, URL: <https://arxiv.org/abs/2211.13080v1> (дата обращения: 18.11.2022).
15. Ovide A., Rodrigo S., Bandic M., Van Someren H., Feld S. et al. Mapping quantum algorithms to multi-core quantum computing architectures, ArXiv, 2023, art. 20232303.16125v1. URL: <https://arxiv.org/pdf/2303.16125.pdf> (дата обращения: 18.11.2022).
16. Abhijith J., Adedoyin A., Ambrosiano J., Anisimov P. et al. Quantum algorithm implementations for beginners. ArXiv, 2022, art. 1804.03719v3. URL: <https://arxiv.org/abs/1804.03719v3> (дата обращения: 27.06.2022).
17. Tezuka H., Nakaji K., Satoh T., Yamamoto N. Grover search revisited; application to image pattern matching. ArXiv, 2021, art. 2108.10854v2. URL: <https://arxiv.org/abs/2108.10854v2> (дата обращения: 01.10.2021).
18. Vlasic A., Certo S, Pham A. Complement Grover’s search algorithm: An amplitude suppression implementation, ArXiv, 2022, art. 2209.10484v1. URL: <https://arxiv.org/abs/2209.10484> (дата обращения: 27.09.2022).
19. Chattopadhyay A., Menon V. Fast simulation of Grover’s quantum search on classical computer. ArXiv, 2021, art. 2005.04635. URL: <https://arxiv.org/pdf/2005.04635.pdf> (дата обращения: 27.09.2022).
20. Toffano Z., Dubois F. Adapting logic to physics: The quantum-like eigenlogic program, Entropy, 2020, vol. 22, no. 2, art. 139. doi: 10.3390/e22020139.

## Авторы

**Ульянов Сергей Викторович**<sup>1,2</sup>,  
д.ф.-м.н., профессор,  
ulyanovsv46\_46@mail.ru  
**Ульянов Виктор Сергеевич**<sup>3</sup>,  
к.т.н., доцент,  
ulyanovik@mail.ru

## Authors

**Sergey V. Ulyanov**<sup>1,2</sup>,  
Dr.Sc. (Physics and Mathematics), Professor,  
ulyanovsv46\_46@mail.ru  
**Viktor S. Ulyanov**<sup>3</sup>,  
Ph.D. (Robotics and Mechatronics),  
Associate Professor, ulyanovik@mail.ru

<sup>1</sup> Государственный университет «Дубна» –  
Институт системного анализа и управления,  
г. Дубна, 141980, Россия

<sup>2</sup> Объединенный институт ядерных исследований –  
Лаборатория информационных технологий  
им. М.Г. Мещерякова, г. Дубна, 141980, Россия

<sup>3</sup> Московский государственный университет  
геодезии и картографии (МИИГАиК),  
г. Москва, 105064, Россия

<sup>1</sup> Dubna State University –  
Institute of System Analysis and Management,  
Dubna, 141980, Russian Federation

<sup>2</sup> Joint Institute for Nuclear Research –  
Meshcheryakov Laboratory of Information Technologies,  
Dubna, 141980, Russian Federation

<sup>3</sup> Moscow State University of Geodesy  
and Cartography (MIIGAiK),  
Moscow, 105064, Russian Federation

## Разработка алгоритма пропорционального разделения ГН-графа для формирования зон влияния объектов в сложных технических системах

Е.Р. Мунтян

### Ссылка для цитирования

Мунтян Е.Р. Разработка алгоритма пропорционального разделения ГН-графа для формирования зон влияния объектов в сложных технических системах // Программные продукты и системы. 2023. Т. 36. № 3. С. 378–387. doi: 10.15827/0236-235X.142.378-387

### Информация о статье

Поступила в редакцию: 22.03.2023

После доработки: 24.05.2023

Принята к публикации: 02.06.2023

**Аннотация.** В статье предложено одно из возможных решений задачи формирования зон влияния объектов в сложных технических системах. В качестве примера рассматривается система охраны протяженного периметра, исследуется взаимодействие ее объектов (элементов) – мобильных или стационарных объектов охраны, квадрокоптеров, лиц, принимающих решение, возможных потенциальных нарушителей периметра. Причем квадрокоптеры обладают различными техническими характеристиками, в частности, радиусами обзора видеокамер. Для моделирования процесса взаимодействия объектов системы охраны применяется модель на основе нечеткого графа с разнотипными вершинами и множественными и разнотипными связями (ГН-графа). В качестве множественных в ГН-графе используются связи в виде вектора, объединяющие несколько разнотипных связей в одну. Такая модель позволяет задать все необходимые отношения между элементами системы и при этом обладает преимуществом во времени вычисления расстояний по сравнению с графами, использующими только однотипные и разнотипные связи. Для решения поставленной задачи предлагаются алгоритмические средства моделирования ГН-графа, в том числе алгоритм пропорционального разделения графа и средства вычисления его метрик. В работе определена операция разделения ГН-графа, сформулированы критерии разделения – пропорциональность подграфов по заданному параметру и возможность пересечения подграфов. Выполнен синтез алгоритма пропорционального разделения ГН-графа в соответствии с данными критериями, результаты работы которого показаны на примере рассмотренной графовой модели. Использование предложенного алгоритма для разделения графа на пропорциональные подмножества и средств вычисления метрических характеристик полученных подграфов позволяет определить зоны влияния объектов системы в соответствии с их техническими параметрами. Рассмотрены возможности программной реализации предложенного алгоритма.

**Ключевые слова:** алгоритм разделения графа, нечеткий граф, ГН-граф, радиус, диаметр, разнотипные связи, множественные связи, система охраны периметра, сложная техническая система

В настоящее время отмечается интерес к исследованию *сложных технических систем* (СТС) [1], включая задачу планирования совместных действий объектов системы [2]. Для исследования взаимодействия объектов СТС применяют различные подходы, в том числе с использованием графовых моделей [3], метаграфов, гиперграфов и др. [4, 5]. Задача планирования совместных действий объектов СТС требует определения функциональных особенностей каждого из них, а также исследования отношений между объектами и их взаимного влияния.

В данной статье в качестве примера СТС рассмотрим систему охраны протяженного периметра. Ее элементами являются:

- стационарные или мобильные объекты охраны;
- необитаемые интеллектуальные взаимодействующие мобильные роботизированные платформы – квадрокоптеры, предназначенные для сбора данных об объектах охраны;
- потенциальные нарушители охраняемого периметра;

– компьютер (как ЛПП), обрабатывающий информацию о возможных действиях объектов охраны, квадрокоптеров и нарушителей территории.

Совместное действие всех элементов системы обеспечивает функцию охраны протяженного периметра. Для исследования взаимодействия элементов системы необходимо построить ее конкретную модель и выбрать алгоритмические средства для получения требуемых характеристик. В работе [6] предложен онтологический подход на основе нечетких графов с множественными и разнотипными связями (ГН-граф), в качестве предметной области используются СТС. Данное исследование посвящено развитию процесса моделирования систем на основе ГН-графов в части разработки алгоритмических средств для решения практических задач предметной области. ГН-граф задан нечеткими множествами вершин и связей. Веса вершин и/или связей соответствуют степеням принадлежности нечетким множествам и могут назначаться экспертом или вычисляться ЛПП путем фазификации [7, 8].

В таком графе вершины представляют элементы системы, а связи – отношения между ними. В ГН-графе вершины являются разнотипными, а множество связей допускает сочетание однотипных, разнотипных и множественных связей в виде вектора. Множественная связь в виде вектора объединяет несколько разнотипных связей, что позволяет уменьшить время вычисления характеристик в ГН-графах [9] по сравнению с графами, использующими только однотипные и разнотипные связи [10, 11].

В качестве алгоритмических средств модели предлагается синтез алгоритма разделения графа на подграфы и вычисления их метрических характеристик. Данная работа в большей степени посвящена реализации именно алгоритмических средств модели на основе ГН-графа. Для решения поставленной задачи кратко рассмотрим существующие методы и алгоритмы разделения графов.

Известны различные алгоритмы разделения графов с однотипными связями, которые можно разделить на несколько групп [12]: геометрические, генетические, спектральные, комбинаторные, а также иерархические алгоритмы с учетом многоуровневой оптимизации.

Примерами геометрических методов разделения графа служат рекурсивный инерционный метод, метод разбиения с помощью кривых, заполняющих пространство, и др. Такие методы позволяют использовать координаты узлов сети в зависимости от их пространственной близости, но не учитывают при этом связность узлов. Для геометрических методов разделения характерна более высокая скорость по сравнению с некоторыми другими методами, но при этом результаты разделения обычно хуже.

Генетические алгоритмы основаны на моделировании генетических процессов биологических организмов. Известно большое количество генетических алгоритмов, применяемых для решения задач дискретной оптимизации, в том числе для разделения графов [13]. Отмечаются универсальность и простота реализации таких алгоритмов, однако нет гарантии, что за приемлемое время будет найден оптимальный вариант разделения графа. Для спектральных методов разделения графа характерно значительно большее время работы по сравнению с алгоритмами многоуровневой оптимизации [12].

В отличие от геометрических методов группа комбинаторных алгоритмов позволяет

учитывать связность вершин, а не их пространственное положение. Среди комбинаторных методов можно выделить:

- 1) алгоритмы вложенных сечений;
- 2) метод деления с учетом связности (LND);
- 3) алгоритм Кернигана–Лина и его модификации [14];
- 4) алгоритм BLP [15] и др.

В работе [11] сравниваются результаты разделения заданного графа несколькими методами, при этом используются вышеперечисленные методы 2–4 и предложенный алгоритм разделения графа на пропорциональные подграфы на основе алгоритма свертки. Рассматривались и несколько алгоритмов многоуровневой оптимизации [16, 17]. В результате рекомендовано использование предложенного алгоритма разделения графа на основе алгоритма свертки. К преимуществам данного алгоритма можно отнести следующие возможности:

- разделение графа на равные или пропорциональные части;
- использование уже готового и реализованного алгоритма свертки графа;
- снижение времени работы разработанного алгоритма до 20 % в зависимости от полноты графа по сравнению с алгоритмами, использующими матрицы смежности или инцидентности и последовательный перебор вершин.

Однако предложенный в работе [12] алгоритм предназначен для разделения неориентированного графа с однотипными связями, а значит, необходима его модификация, если использовать ГН-граф. Для этого следует определить операцию разделения ГН-графа и сформулировать критерии работы алгоритма в зависимости от требований прикладной задачи. Сначала рассмотрим пример модели системы на основе ГН-графа.

### Модель системы на основе ГН-графа

Формально ГН-граф задан нечеткими конечными множествами  $G^v$  и  $G^e$  [6]:

$$G' = (G^v, G^e), \quad (1)$$

где  $G^v = \{g^v_i | i = 1, 2, \dots, n\}$  – множество разнотипных вершин, а  $G^e = \{g^e_j | j = 1, 2, \dots, m\}$  – множество связей. В качестве связей допускаются ребра и/или дуги (двунаправленные и/или однонаправленные связи) по трем категориям: однотипные, разнотипные (*tp – type*) и множественные связи в виде вектора  $v = \langle v_1, v_2, \dots, v_t \rangle$ , где  $t$  – размерность (кратность) вектора  $v$ . Граф представлен списковой структурой, в ко-

торой отражена информация не только о логических связях в графе, но и об атрибутах вершин и связей, в частности, типы и веса вершин связей, и др. (все атрибуты указаны в работе [6]).

Будем полагать, что система содержит десять элементов: объекты охраны (ОО<sub>1</sub>–ОО<sub>6</sub>), квадрокоптеры (К<sub>1</sub>, К<sub>2</sub>), компьютер (ЛПП), потенциальный нарушитель (ПН). При этом технические характеристики квадрокоптеров К<sub>1</sub> и К<sub>2</sub> различаются: радиус обзора видеокамеры квадрокоптера К<sub>1</sub> больше, чем у К<sub>2</sub>, в 1,5 раза. Тогда модель такой системы в соответствии с (1) будет задана графом  $G'_1 = (G'v_1, G'e_1)$ , где  $G'v_1 = \{gv_i | i = 1, 2, \dots, 10\}$  и  $G'e_1 = \{ge_j | j = 1, 2, \dots, 22\}$  (рис. 1). Соответствие элементов и отношений в системе вершинам и связям графовой модели отражено в таблицах 1 и 2.

Поскольку в качестве модели системы используется нечеткий граф, для оценки значения показателей весов вершин и связей графа  $G'_1$  необходимо привести их вербальное описание, в рассматриваемом случае это слабый (вес 0,01–0,3), умеренный (вес 0,31–0,7), сильный (вес 0,71–1,0) показатели. С учетом принятой оценки показателей весов вершин и связей графа целесообразно привести некоторые характеристики элементов системы к принятой шкале, например, для квадрокоптера К<sub>1</sub> радиус видимости  $r_1$  примет значение 0,9 (сильный показатель), а для квадрокоптера К<sub>2</sub> радиус  $r_2$  равен 0,6 (умеренный показатель), что соответ-

ствует заявленным техническим характеристикам этих квадрокоптеров.

Рассмотрим некоторые аспекты модели системы, предложенной на рисунке 1. В графе  $G'_1$  ребра типа  $tp_1$  соответствуют отношению «Удаленность объектов», например, между объектами ОО<sub>1</sub> и ОО<sub>2</sub> расстояние может быть определено как небольшое (слабый показатель), следовательно, в модели между вершинами  $gv_1$  и  $gv_2$  отражено ребро  $ge_1$  с весом  $\mu_1 = 0,3$ . При этом между объектами ОО<sub>1</sub> и ОО<sub>3</sub> не может быть определено расстояние (так как оно превышает принятую шкалу показателей), следовательно, в модели между вершинами  $gv_1$  и  $gv_3$  нет связи.

Например, наличие дуги ( $ge_9/tp_2$ ) между вершинами  $gv_8$  и  $gv_1$  на графе  $G'_1$  означает, что объект охраны ОО<sub>1</sub> находится в зоне видимости квадрокоптера К<sub>1</sub>. Таким образом, в зоне видимости квадрокоптера К<sub>1</sub> находятся элементы системы ОО<sub>1</sub>, ОО<sub>2</sub>, ОО<sub>3</sub>, ОО<sub>4</sub> и ПН, что отражено на графе при помощи соответствующих дуг типа  $tp_2$  ( $ge_9, ge_{10}, ge_{11}, ge_{12}, ge_{21}$ ). Аналогично в зоне видимости квадрокоптера К<sub>2</sub> должны находиться элементы системы ОО<sub>3</sub>, ОО<sub>4</sub>, ОО<sub>5</sub>, ОО<sub>6</sub> и ПН.

Так, предложенная модель системы на основе ГН-графа позволяет представить все ее необходимые элементы и отношения. Но для решения задачи формирования зон влияния определенных элементов системы (в данном

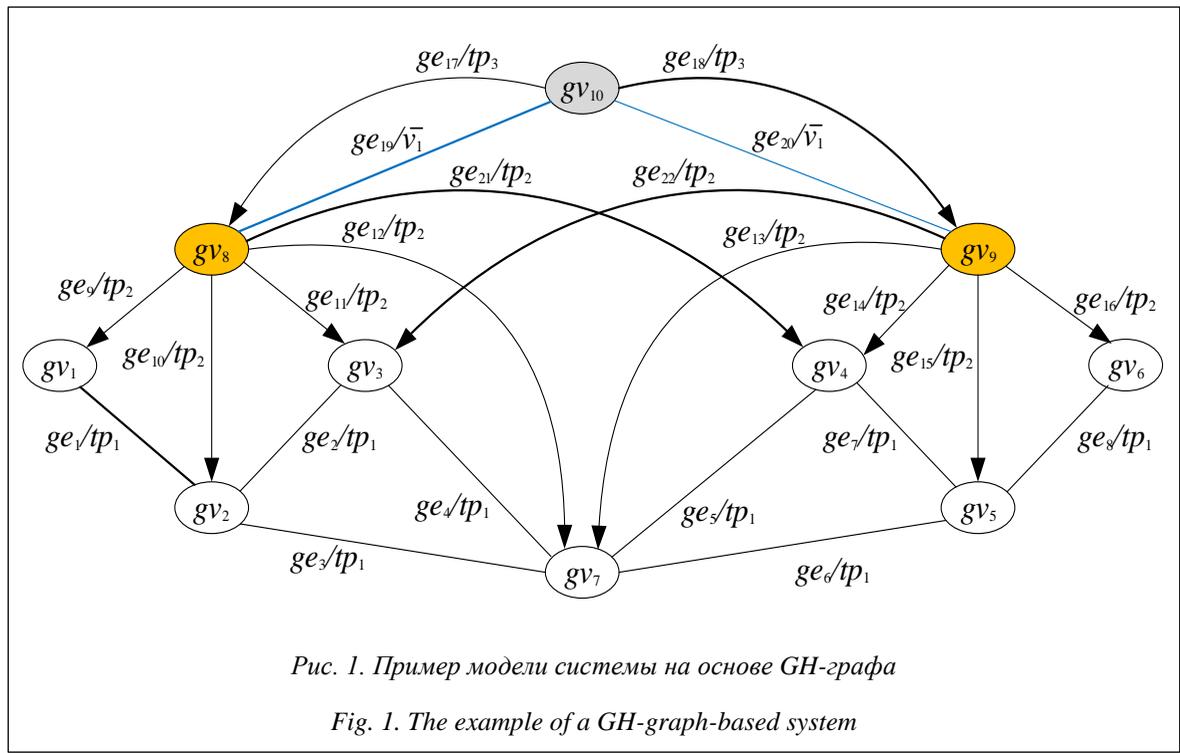


Рис. 1. Пример модели системы на основе ГН-графа

Fig. 1. The example of a GH-graph-based system

Таблица 1

Соответствие отношений между элементами системы связям графа  $G'_1$

Table 1

The conformity of the relations between system elements to the graph  $G'_1$  edges

Вид связи графа	Тип связи графа, $tp$	Отношение между элементами системы	Связи графа, $ge$	Вес связи, $\mu_j$	Интерпретация веса связи графа
Разнотипная связь	$tp_1$	«Удаленность объектов»	$ge_1$	0,3	Степень удаленности объектов охраны
			$ge_2$	0,5	
			$ge_3$	0,4	
			$ge_4$	0,7	
			$ge_5$	0,3	
			$ge_6$	0,5	
			$ge_7$	0,8	
			$ge_8$	0,5	
	$tp_2$	«Наблюдать»	$ge_9$	0,5	Степень удаленности квадрокоптера и объекта охраны
			$ge_{10}$	0,9	
			$ge_{11}$	0,8	
			$ge_{12}$	0,9	
			$ge_{13}$	0,6	
			$ge_{14}$	0,3	
			$ge_{15}$	0,5	
			$ge_{16}$	0,6	
			$ge_{21}$	0,8	
			$ge_{22}$	1,0	
$tp_3$	«Управлять движением квадрокоптера»	$ge_{17}$	0,5	Степень соответствия заданному направлению движения	
		$ge_{18}$	0,5		
Связь в виде вектора	$\bar{v}_1$	Передача информации по каналам: $v_{1.1}$ – Wi-Fi; $v_{1.2}$ – Bluetooth; $v_{1.3}$ – УКВ	$ge_{19}$	0; 1; 0	Степень достоверности использования канала передачи данных
			$ge_{20}$	0; 1; 0	

Таблица 2

Соответствие элементов системы вершинам графа  $G'_1$

Table 2

The conformity of the system elements to the graph  $G'_1$  vertexes

Элемент системы	Вершина графа	Тип вершины	Вес вершины, $\eta_i$	Интерпретация веса вершины графа
ОО <sub>1</sub> –ОО <sub>6</sub>	$gv_1 - gv_6$	$tpv_1$	по 0,3	Степень значимости элемента системы
ПН	$gv_7$	$tpv_2$	0,8	
К <sub>1</sub> , К <sub>2</sub>	$gv_8, gv_9$	$tpv_3$	по 0,5	
ЛПР	$gv_{10}$	$tpv_4$	1,0	

случае – квадрокоптеров) необходимо применение алгоритма разделения ГН-графа и средств вычисления его метрик.

**Алгоритм пропорционального разделения ГН-графа**

Для начала определим понятие «разделение ГН-графа». Будем полагать, что разделение

ГН-графа на  $n$  подграфов – это процесс представления исходного графа  $G'$  в виде множества вершин  $Gv = \{Gv_1, Gv_2, \dots, Gv_n\}$  с инцидентными им связями  $ge_j \in Ge$  в соответствии с принятыми критериями, где  $Gv_1, Gv_2, \dots, Gv_n$  – множества вершин  $n$  подграфов.

Сформулируем критерии разделения ГН-графа, обусловленные требованиями к решению прикладных задач:

- пропорциональность подграфов по заданному параметру (количество или вес вершин, связей и др.);
- наличие или отсутствие пересечений подграфов.

Признак пересечения подграфов  $Pa$  определяется следующим образом:

$$Pa = \begin{cases} 0, & \text{если нет пересечения подграфов;} \\ 1, & \text{если есть пересечение подграфов.} \end{cases}$$

Обоснованность сформулированных критериев разделения GH-графа рассмотрим впоследствии на примере разделения графа  $G'_1$ . На рисунке 2 приведена блок-схема алгоритма пропорционального разделения GH-графа на подграфы, в основе которого лежит использование алгоритма свертки графа. В соответствии

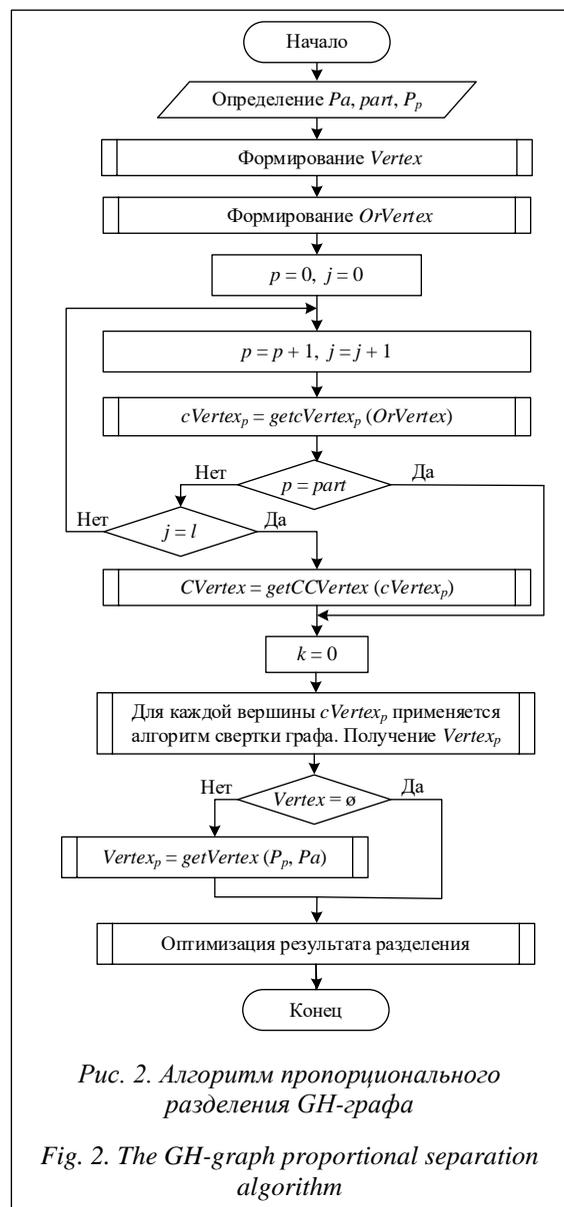


Рис. 2. Алгоритм пропорционального разделения GH-графа

Fig. 2. The GH-graph proportional separation algorithm

с данным алгоритмом разделение GH-графа выполняется с учетом сформулированных критериев. Разработанный алгоритм основан на предложенном в работе [12] алгоритме пропорционального разделения неориентированного графа с сохранением значения вычислительной сложности и всех преимуществ оригинального алгоритма.

На предварительном этапе работы алгоритма определяются следующие параметры: количество подграфов  $part$ , их пропорциональные доли  $P_p \in ]0,1[$  и признак пересечения подграфов  $Pa$  (на соответствие критериям). Затем выполняется коррекция списка графа, которая заключается в фильтрации списка по заданному типу или весу вершин и/или связей. После этого формируются множество доступных вершин графа в текущий момент времени  $Vertex = \{Vertex_i \mid i = 1, 2, \dots, (n - 1)\}$  и подмножество выбранных вершин  $OrVertex = \{orVertex_j \mid j = 1, 2, \dots, l\}$ . В качестве выбранных вершин могут быть вершины заданного типа или принадлежащие группе центральных (или периферийных) вершин и др.

Из подмножества  $OrVertex$  при помощи функции  $cVertex_p = getCVertex_p(OrVertex)$  каждый раз определяется одна исходная вершина, пока не будет сформировано подмножество исходных вершин как  $CVertex = getCVertex(cVertex_p)$ . Полученное подмножество  $CVertex$  необходимо для работы алгоритма свертки графа, подробно описанного в работе [12]. Алгоритм свертки использует одну вершину в качестве исходной и позволяет свернуть ее и смежные ей вершины вместе с инцидентными ребрами в одну новую вершину за одну итерацию. Итерации повторяются по необходимости. При этом свернутая часть хранится в виде отдельного списка, что позволяет вернуть граф в первоначальное состояние. В рассматриваемом случае для каждой исходной вершины  $cVertex_p$  применяется алгоритм свертки графа в параллельном режиме. При этом вместе с исходной вершиной  $cVertex_p$  поглощаются смежные ей вершины из подмножества  $Neighbors$ . Таким образом формируются текущие подмножества вершин подграфов  $Vertex_p$ .

Рассмотрим пример использования предложенного алгоритма разделения GH-графа и средств вычисления его метрик для решения задачи формирования зон влияния квадрокоптеров. Для вычисления метрик графа рекомендовано использовать стандартные средства, например, описанные в работе [18].

### Формирование зон влияния элементов системы при помощи алгоритма разделения ГН-графа

Система содержит квадрокоптеры с различными техническими характеристиками, поэтому их зоны влияния должны быть пропорциональными (60 % и 40 %). Рассмотрим реализацию предложенного алгоритма разделения ГН-графа в программном модуле на примере модели системы, которая представлена в виде графа  $G'_1$  [19].

Шаг 1. На начальном этапе определим параметры:

- количество подграфов  $part = 2$  (по количеству квадрокоптеров в системе –  $K_1$  и  $K_2$ );
- пропорциональные доли подграфов  $P_1 = 0,6$ ,  $P_2 = 0,4$  (по атрибуту веса связи);
- признак пересечения подграфов  $Pa = 1$ .

Шаг 2. Выполним фильтр вершин и связей в списке графа  $G'_1$ , оставим вершины типа  $tpv_1$ – $tpv_3$  (соответствуют объектам охраны, квадрокоптерам и потенциальному нарушителю) и связи типа  $tp_2$  (отношение «наблюдать»), таким образом сформируем текущее множество вершин  $Vertex$ .

Шаг 3. В качестве исходных вершин для алгоритма выберем вершины, которые соответствуют квадрокоптерам  $K_1$ ,  $K_2$  ( $cVertex_1 = gv_8$  и  $cVertex_2 = gv_9$ ). Тогда подмножество исходных вершин примет вид  $CVertex = \{cVertex_1, cVertex_2\}$ .

Шаг 4. Сформируем подмножества смежных вершин для  $ge_8$  и  $ge_9$ , получим, соответственно,  $Neighbors_1 = \{gv_1, gv_2, gv_3, gv_4, gv_7\}$  и  $Neighbors_2 = \{gv_3, gv_4, gv_5, gv_6, gv_7\}$ . При этом очевидно, что  $Neighbors_1 \cap Neighbors_2 = \{gv_3, gv_4, gv_7\}$ .

Шаг 5. Выполним алгоритм свертки графа в параллельном режиме, начиная с исходных вершин  $ge_8$  и  $ge_9$ . Поскольку  $Neighbors_1 \cap Neighbors_2 \neq \emptyset$ , отработывается функция распределения вершины из множества  $Vertex$  в подмножества  $Vertex_1$  и  $Vertex_2$  в соответствии с критериями разделения, как показано на рисунке 2,  $Vertex_p = getVertex(P_p, Pa)$ .

Шаг 6. Сформируем подмножества вершин подграфов  $Vertex_1 = \{ge_1, ge_2, ge_3, ge_4, ge_7, ge_8\}$  и  $Vertex_2 = \{ge_3, ge_4, ge_5, ge_6, ge_7, ge_9\}$ . Алгоритм свертки заканчивает свою работу, так как смежных вершин для вершин из этих подмножеств больше нет. Поскольку все вершины из множества  $Vertex$  распределены в подмножества  $Vertex_1$  и  $Vertex_2$ , переходим на шаг 7.

Шаг 7. Оптимизируем полученный результат. Поясним этот шаг подробнее. Вес связи типа  $tp_2$  (отношение «Наблюдать») характеризует степень удаленности квадрокоптера и объекта охраны. Для вершин из подмножества  $Neighbors_1$  веса связей распределяются следующим образом:  $\mu_9 = 0,5$ ,  $\mu_{10} = 0,9$ ,  $\mu_{11} = 0,8$ ,  $\mu_{12} = 0,9$ ,  $\mu_{21} = 0,8$  (максимальное числовое значение – 0,9), а для вершин из подмножества  $Neighbors_2$  веса связей  $\mu_{13} = 0,6$ ,  $\mu_{14} = 0,3$ ,  $\mu_{15} = 0,5$ ,  $\mu_{16} = 0,6$ ,  $\mu_{22} = 1,0$  (максимальное числовое значение – 1,0). Очевидно, что в соответствии с принятыми значениями радиусов обзора видеокамер квадрокоптеров  $K_1$  и  $K_2$  ( $r_1 = 0,9$  и  $r_2 = 0,6$ ) объект охраны  $OO_3$  не попадает в зону видимости квадрокоптера  $K_2$ , поэтому вершина  $gv_3$  должна быть исключена из подмножества  $Vertex_2$  в соответствии с принятыми критериями.

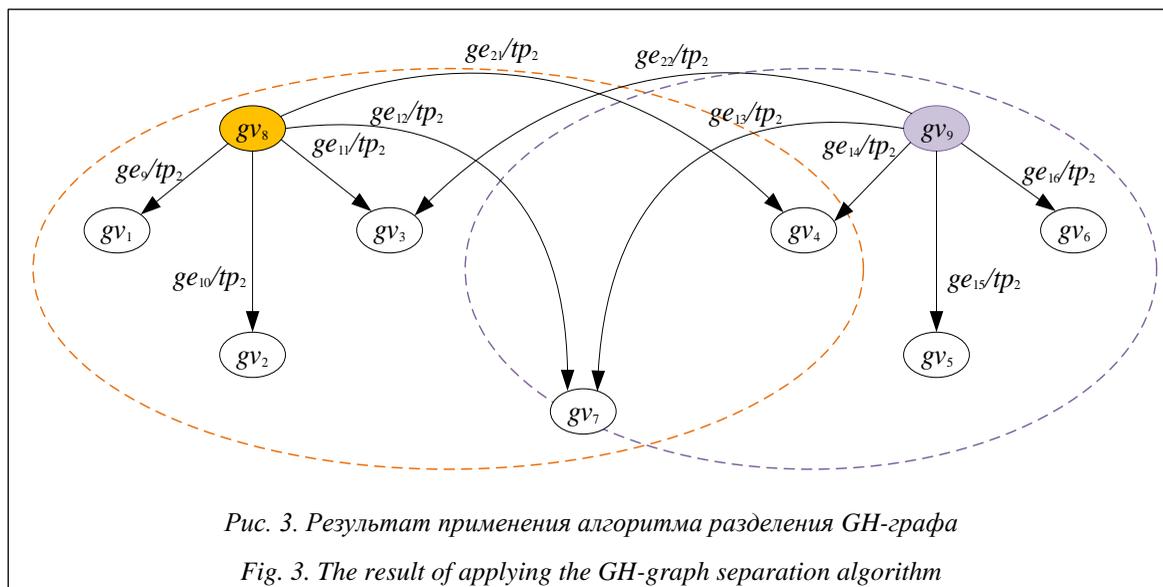
Тогда подмножества вершин подграфов примут значения  $Vertex_1 = \{ge_1, ge_2, ge_3, ge_4, ge_7, ge_8\}$  и  $Vertex_2 = \{ge_4, ge_5, ge_6, ge_7, ge_9\}$ . Результат разделения графа  $G'_1$  представлен на рисунке 3.

Полученные таким образом подмножества вершин подграфов  $Vertex_1$  и  $Vertex_2$  позволяют сформировать зоны влияния квадрокоптеров  $K_1 = \{OO_1, OO_2, OO_3, OO_4, ПН\}$  и  $K_2 = \{OO_4, OO_5, OO_6, ПН\}$  в заданных пропорциях. Вычисление метрик полученных подграфов (радиус, диаметр) позволяет определить соответствие зоны влияния техническим параметрам элементов системы, в рассматриваемом случае диаметр подграфа не должен превышать радиус обзора видеокамеры квадрокоптера (в принятой шкале). Пересечение зон влияния квадрокоптеров в системе обусловлено необходимостью соблюдения жизненного цикла каждого устройства, например, временные рамки работоспособности, необходимость в подзарядке и т.д. Очевидно, что с учетом жизненного цикла для обеспечения функций охраны заданного периметра достаточно трех квадрокоптеров.

### Программная реализация

Программный модуль [19], реализующий предложенный алгоритм разделения и средства вычисления характеристик графа, встроен в программный комплекс моделирования системы на основе ГН-графа (ПК), как показано на рисунке 4.

Для создания графического изображения используется блок редактирования графа.



Сразу после загрузки (посредством блока загрузки графа) или по мере создания графа определяются его основные параметры. Полученный граф сохраняется в блоке хранения графа в виде .xml-файла, в котором хранится его списковое представление, и посредством блока отображения выводится на экран. Блок вычисления характеристик графа и блок реализации алгоритма разделения графа составляют программный модуль реализации алгоритмов графа.

ПК разработан на языке C++ с использованием библиотеки Qt. Все используемые средства реализации являются кроссплатформенными, что обуславливает возможность функционирования ПК в различных операционных системах. Разработанный программный модуль планируется использовать с применением SQL или NoSQL БД, например, графовой СУБД Neo4j и других БД [20].

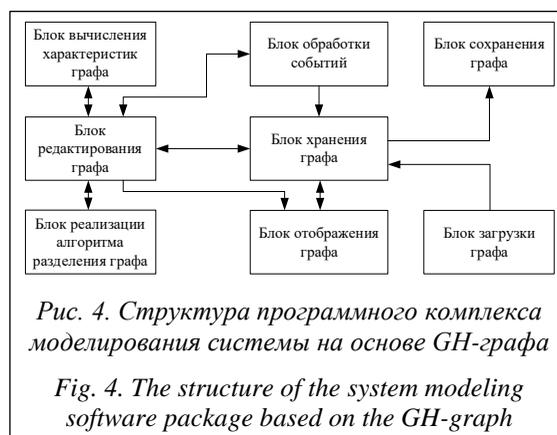
### Заключение

В работе поставлена задача синтеза формирования зон влияния объектов в сложных технических системах, для решения которой предложен алгоритм пропорционального разделения графа с множественными и разнотипными связями (GH-графа). Приведено формальное описание такого графа, рассмотрен пример модели на основе GH-графа фрагмента конкретной системы охраны протяженного периметра, которая содержит ряд объектов (элементов) – это объекты охраны, квадрокоптеры, ЛПР (компьютер) и нарушитель периметра, причем квадрокоптеры обладают различными техни-

ческими характеристиками, в частности, различаются радиусы обзора их видеокамер. Такая модель позволяет задать все необходимые отношения между элементами системы и при этом обладает преимуществом во времени вычисления расстояний по сравнению с графами, использующими только однотипные и разнотипные связи.

Определена операция разделения GH-графа, сформулированы критерии разделения, в том числе пропорциональность подграфов по заданному параметру (количество или вес вершин, связей и др.) и возможность пересечения подграфов.

В соответствии с сформулированными критериями выполнен синтез модифицированного алгоритма пропорционального разделения GH-графа, результаты работы которого показаны на примере рассмотренной графовой модели. В качестве оригинального алгоритма для модификации использован разработанный ранее алгоритм пропорционального разделения



неориентированного графа, который позволяет уменьшить время разделения графа по сравнению с алгоритмами, использующими матрицы смежности или инцидентности и последовательный перебор вершин в зависимости от полноты графа до 20 %. При этом для модифицированного алгоритма пропорционального разделения ГН-графа сохраняются значения вычислительной сложности и всех преимуществ оригинального алгоритма.

Пропорциональность подграфов обусловлена весами связей между вершинами определенного типа для соответствия техническим характеристикам некоторых элементов системы. Необходимость пересечения подграфов может быть обусловлена требованиями к со-

блюдению жизненных циклов объектов системы.

Выполнена программная реализация модуля, содержащего алгоритм разделения графа на пропорциональные подмножества и средства вычисления метрик, интегрированного в программный комплекс моделирования системы на основе ГН-графа. Реализованная в программном комплексе модель СТС на примере системы охраны, а также использование предложенного алгоритма для разделения графа на пропорциональные подмножества и средства вычисления метрических характеристик полученных подграфов (радиус, диаметр) позволяют определить зоны влияния объектов системы в соответствии с их техническими параметрами.

### Список литературы

1. Коноваленко С.А., Шабля В.О., Титов Г.О. Анализ методов контроля состояния процесса функционирования сложных технических систем // *Наукофера*. 2021. № 12-2. С. 224–230.
2. Дубровина А.И. Состав инженерной защиты и технической охраны объектов // *Modern Sci*. 2020. № 12-3. С. 236–241.
3. Буравцев А.В., Щенников А.Н. Проектирование данных для компьютерной обработки в фискальных системах // *ИТНОУ*. 2018. № 1. С. 3–14.
4. Плесневич Г.С., Тарасов В.Б. Метаграфы с темпорально-логическими ограничениями // *ИММВ-2021*: сб. науч. тр. 2021. С. 201–219.
5. Миков А.И., Миков А.А. Динамические гиперграфы процессов восстановления мобильных сетей // *Информатизация и связь*. 2021. № 5. С. 31–38. doi: 10.34219/2078-8320-2021-12-5-31-38.
6. Еремеев А.П., Мунтян Е.Р. Разработка онтологии на основе графов с множественными и разнотипными связями // *Искусственный интеллект и принятие решений*. 2021. № 3. С. 3–18. doi: 10.14357/20718594210301.
7. Minaev Y.N., Filimonova O.Y., Minaeva J.I., Filimonov A. Fuzzy mathematics with limited possibilities for assigning membership functions. *Cybernetics and Systems Analysis*, 2020, vol. 56, no. 1, pp. 29–39. doi: 10.1007/s10559-020-00218-9.
8. Дандыбаев С.Т. Нечеткие множества с нечеткими функциями принадлежности // *Теория и практика современной науки*. 2021. № 1. С. 130–133.
9. Мунтян Е.Р. Реализация нечеткой модели взаимодействия объектов сложных технических систем на основе графов // *Программные продукты и системы*. 2019. Т. 32. № 3. С. 411–418. doi: 10.15827/0236-235X.127.411-418.
10. Smirnov A.V. Spanning tree of a multiple graph. *J. of Combinatorial Optimization*, 2021, vol. 43, no. 4, pp. 850–869. doi: 10.1007/s10878-021-00810-5.
11. Кальней А.М. Модели многоуровневых сетей (краткий обзор) // *Проблемы информатики*. 2021. № 3. С. 5–20.
12. Сергеев Н.Е., Мунтян Е.Р. Применение алгоритма свертки для разделения графа на пропорциональные подграфы // *Вестн. УГАТУ*. 2018. Т. 22. № 1. С. 121–130.
13. Akhand M.A.H., Ayon S.I., Shahriar S.A., Siddique N. Discrete spider monkey optimization for travelling salesman problem. *Applied Soft Computing*, 2020, vol. 86, art. 105887. doi: 10.1016/j.asoc.2019.105887.
14. Fiduccia C.M., Mattheyses R.M. A linear-time heuristics for improving network partitions. *Proc. 19th Design Automation Conf.*, 1982, pp. 175–181. doi: 10.1109/DAC.1982.1585498.
15. Moussawi A.El., Ruiz R.R., Seghouani N.B. Sampling-based label propagation for balanced graph partitioning. *Proc. ACM/SPEC on ICPE*, 2022, pp. 223–230. doi: 10.1145/3489525.3511698.
16. Курейчик В.В., Заруба Д.В. Двухуровневый алгоритм разбиения графа на части // *Изв. ЮФУ. Технич. науки*. 2019. № 2. С. 6–15.
17. Курейчик В.В., Курейчик Вл.Вл. Программная подсистема для решения NP-сложных комбинаторно-логических задач на графах // *Изв. ЮФУ. Технич. науки*. 2021. № 2. С. 39–50.
18. Muntyan E.R., Melnik E.V. The graph-based analysis of structural delays in distributed multiprogram systems of information processing. *JPCS*, 2020, vol. 1661, no. 1, art. 012061. doi: 10.1088/1742-6596/1661/1/012061.
19. Мунтян Е.Р. Программный модуль для моделирования взаимодействия акторов и групп акторов на основе графов: Свид. о регистр. ПрЭВМ № 2018665593. Рос Федерация, 2018.
20. Еремеев А.П., Панявин Н.А. Унификация модели представления данных и преобразование форматов на основе нереляционной СУБД Neo4j // *Программные продукты и системы*. 2022. Т. 35. № 4. С. 549–556. doi: 10.15827/0236-235X.140.549-556.

## Developing a GH-graph proportional separation algorithm to form of object influence zones in complex technical systems

Evgeniya R. Muntyan

**For citation**

Muntyan, E.R. (2023) 'Developing a GH-graph proportional separation algorithm to form of object influence zones in complex technical systems', *Software & Systems*, 36(3), pp. 378–387 (in Russ.). doi: 10.15827/0236-235X.142.378-387

**Article info**

Received: 22.03.2023

After revision: 24.05.2023

Accepted: 02.06.2023

**Abstract.** The article suggests one of the possible solutions to the problem the object influence zones in complex technical systems (CTS). It considers the extended perimeter security system as a CTS example, investigates the interaction of its objects (elements). The objects of such a system are mobile or stationary security objects, quadcopters, decision makers, possible potential perimeter violators; provided that quadcopters have different technical characteristics, in particular, the their video camera surveillance radii differ. To simulate the process of interaction of security system objects, the authors use a model based on a fuzzy graph with different types of vertices and multiple and different types of connections (GH-graph). Multiple connections in the GH-graph are vector connections that combine several different types of connections into one. Such model allows setting all necessary relationships between system elements and at the same time, it has an advantage in the time of calculating distances compared to graphs using only same type and different types of connections. To solve this problem, the authors propose using GH-graph algorithmic modeling means, including the graph proportional separation algorithm and the means of calculating graph metrics. The paper defines the GH-graph splitting operation, defines the separation criteria: the proportionality of subgraphs by a given parameter and the possibility of subgraph intersection. The authors have carried out the synthesis of the GH-graph proportional separation algorithm in accordance with the criteria and shown the algorithm results using the example of the graph model considered. It is shown that using the proposed graph separation algorithm into proportional subsets and the means of calculating the metric characteristics of the obtained subgraphs make it possible to determine the system object influence zones in accordance with their technical parameters. The paper considers the possibilities of software implementation of the proposed algorithm.

**Keywords:** graph separation algorithm, fuzzy graph, GH-graph, radius, diameter, different type edges, multiple edges, perimeter security system, complex technical system

**Reference List**

1. Konovalenko, S.A., Shablya, V.O., Titov, G.O. (2021) 'Analysis of methods for control of the condition of the functioning process of complex technical systems', *Naukosfera*, (12-2), pp. 224–230 (in Russ.).
2. Dubrovina, A.I. (2020) 'A composition of engineering protection and technical protection of facilities', *Modern Sci.*, (12-3), pp. 236–241 (in Russ.).
3. Buravtsev, A.V., Schennikov, A.N. (2018) 'Designing data for computer processing in fiscal systems', *ITNOU*, (1), pp. 3–14 (in Russ.).
4. Plesnevich, G.S., Tarasov, V.B. (2021) 'Metagraphs with temporal-logical constraints', *Proc. Integrated Models and Soft Computing in Artificial Intelligence*, pp. 201–219 (in Russ.).
5. Mikov, A.I., Mikov, A.A. (2021) 'Dynamic hypergraphs of renewal processes in mobile networks', *Informatization and Communication*, (5), pp. 31–38 (in Russ.). doi: 10.34219/2078-8320-2021-12-5-31-38.
6. Ereemeev, A.P., Muntyan, E.R. (2021) 'Development of an ontology based on graphs with multiple edges of different types', *Artificial Intelligence and Decision Making*, (3), pp. 3–18 (in Russ.). doi: 10.14357/20718594210301.
7. Minaev, Y.N., Filimonova, O.Y., Minaeva, J.I., Filimonov, A. (2020) 'Fuzzy mathematics with limited possibilities for assigning membership functions', *Cybernetics and Systems Analysis*, 56(1), pp. 29–39. doi: 10.1007/s10559-020-00218-9.
8. Dandybaev, S.T. (2021) 'Fuzzy sets with fuzzy membership functions', *Theory and Practice of Modern Sci.*, (1), pp. 130–133 (in Russ.).
9. Muntyan, E.R. (2019) 'Implementation of a fuzzy model of interaction between objects in complex technical systems based on graphs', *Software & Systems*, 32(3), pp. 411–418 (in Russ.). doi: 10.15827/0236-235X.127.411-418.
10. Smirnov, A.V. (2021) 'Spanning tree of a multiple graph', *J. of Combinatorial Optimization*, 43(4), pp. 850–869. doi: 10.1007/s10878-021-00810-5.
11. Kalney, A.M. (2021) 'Layered network models (overview)', *Problems of Informatics*, (3), pp. 5–20 (in Russ.).
12. Sergeev, N.E., Muntyan, E.R. (2018) 'Using convolution algorithm to separate a graph on the proportional subgraphs', *Vestn. USATU*, 22(1), pp. 121–130 (in Russ.).
13. Akhand, M.A.H., Ayon, S.I., Shahriar, S.A., Siddique, N. (2020) 'Discrete spider monkey optimization for traveling salesman problem', *Applied Soft Computing*, 86, art. 105887. doi: 10.1016/j.asoc.2019.105887.
14. Fiduccia, C.M., Mattheyses, R.M. (1982) 'A linear-time heuristics for improving network partitions', *Proc. 19th Design Automation Conf.*, pp. 175–181. doi: 10.1109/DAC.1982.1585498.
15. Moussawi, A.El., Ruiz, R.R., Seghouani, N.B. (2022) 'Sampling-based label propagation for balanced graph partitioning', *Proc. ACM/SPEC on ICPE*, pp. 223–230. doi: 10.1145/3489525.3511698.

16. Kureychik, V.V., Zaruba, D.V. (2019) 'Two-level graph partitioning algorithm', *Izv. SFedU. Engineering Sci.*, (2), pp. 6–15 (in Russ.).
17. Kureychik, V.V., Kureychik, V.I. (2021) 'Software subsystem for solving np-complex combinatorial logic problems on graphs', *Izv. SFedU. Engineering Sci.*, (2), pp. 39–50 (in Russ.).
18. Muntyan, E.R., Melnik, E.V. (2020) 'The graph-based analysis of structural delays in distributed multiprogram systems of information processing', *JPCS*, 1661(1), art. 012061. doi: 10.1088/1742-6596/1661/1/012061.
19. Muntyan, E.R. (2018) *A Software Module for Modeling the Interaction of Actors and Actors Groups Based on Graphs*, Pat. RF, № 2018665593.
20. Eremeev, A.P., Paniavin, N.A. (2022) 'Unification of a data presentation model and format conversion based on a non-relational Neo4j DBMS', *Software & Systems*, 35(4), pp. 549–556 (in Russ.). doi: 10.15827/0236-235X.140.549-556.

**Авторы**

**Мунтян Евгения Ростиславна**<sup>1</sup>, к.т.н., доцент,  
ermuntyan@sfedu.ru

<sup>1</sup> Южный федеральный университет,  
г. Таганрог, 347900, Россия

**Authors**

**Evgeniya R. Muntyan**<sup>1</sup>, Ph.D. (Engineering),  
Associate Professor, ermuntyan@sfedu.ru

<sup>1</sup> Southern Federal University,  
Taganrog, 347900, Russian Federation

## Об одной тенденции развития алгоритмов, реализуемых в системах поддержки принятия решений

О.В. Саяпин  
О.В. Тиханычев  
А.А. Безвесильная  
С.В. Чискидов

### Ссылка для цитирования

Саяпин О.В., Тиханычев О.В., Безвесильная А.А., Чискидов С.В. Об одной тенденции развития алгоритмов, реализуемых в системах поддержки принятия решений // Программные продукты и системы. 2023. Т. 36. № 3. С. 388–397. doi: 10.15827/0236-235X.142.388-397

### Информация о статье

Поступила в редакцию: 27.01.2023

После доработки: 15.06.2023

Принята к публикации: 20.06.2023

**Аннотация.** В статье рассмотрены вопросы реализации эффективных алгоритмов управления сложными человеко-машинными системами в современных условиях. Как показывает анализ предметной области, в практике управления все чаще используются алгоритмы, основанные на простых, но надежных методах, использование которых раньше было проблематичным из-за высоких вычислительных затрат. С развитием вычислительной техники, ростом пропускной способности сетей передачи данных и совершенствованием пользовательских интерфейсов практическое использование подобных методов стало возможным. В качестве примера реализации таких алгоритмов рассмотрены методы формирования управляющих воздействий, использующие визуализацию процесса управления с применением технологий виртуальной реальности. Актуальность работы определяется тем, что в условиях управления динамичными пространственно распределенными эргатическими системами существующие методы и технологии управления не всегда обеспечивают требуемую эффективность. В то же время эффективное управление – залог успешного применения любых систем. В статье на основе анализа типового цикла управления и недостатков используемых в настоящее время алгоритмов управления синтезированы предложения по применению моделей, основанных на прямых вычислениях и логических правилах. Как пример рассмотрено использование моделей на основе средств виртуальной реальности для применения визуального алгоритма сведения текущего состояния управляемой системы с требуемым состоянием, рассчитанным для достижения цели. Предложенный алгоритм логически дополняет алгоритмы решения оптимизационных задач и поиска информации, находясь в общем тренде развития алгоритмизации систем поддержки принятия решений в пользу упрощения используемых методов.

**Ключевые слова:** поддержка принятия решений, дополненная реальность, виртуальная реальность, алгоритмическое обеспечение управления, повышение эффективности управления

В теории управления принято оперировать понятием «типовой цикл управления», включающим оценку ситуации, целеполагание (формулирование или уяснение задачи), выработку решения, планирование, постановку задач исполнителям и контроль их выполнения. В ходе данного цикла его участники реализуют широкий спектр логико-аналитических и расчетно-моделирующих функций. Существенная часть задач в рамках реализации этих функций, в первую очередь информационных и расчетно-моделирующих, в настоящее время обеспечивается программными средствами автоматизации управления, или прикладным ПО.

Для решения данных задач в составе прикладных программ реализуется разнообразный математический аппарат преимущественно оптимизационного или прогнозного типа. Для решения могут применяться аналитические, численные или статистические методы, а также методы эвристической самоорганизации, например, основанные на генетических алгоритмах. Специалисты признают, что при ис-

пользовании указанных методов возникают проблемы формализации исходных данных, интерпретации результатов расчетов и настройки вычислительных алгоритмов, влияющие на точность получаемых результатов.

В то же время известен простейший вариант получения гарантированного оптимального (рационального) решения распределительных задач, которым является метод полного перебора. Однако в задачах сколь-нибудь большой размерности применение полного перебора проблематично. Например, при решении задачи коммивояжера для  $n$  агентов размерность полного перебора определяется формулой перестановок  $P_n = n!$ .

С ростом размерности задачи наступает эффект комбинаторного взрыва. Поэтому при решении подобных задач приходится либо использовать усеченные алгоритмы оптимизации, либо снижать размерность входной информации, применяя методы кластеризации данных, ассоциативные правила, генетические алгоритмы и т.п. Кроме оптимизационных

задач, достаточно большие вычислительные мощности требуются для реализуемых в алгоритме типового цикла управления информационных задач поиска данных, решаемых на этапах планирования и управления.

Возможности применения методов решения перечисленных задач существенно зависят от вычислительной мощности средств автоматизации управления. На начальном этапе развития теории управления, когда такие задачи решались без применения средств автоматизации, использовались максимально простые алгоритмы, оптимизированные относительно вычислительных затрат: северо-западного угла, потенциалов, венгерский метод и др. По мере внедрения средств информатизации доступные методы усложнялись, появились градиентные методы поиска экстремума, вероятностные модели, а впоследствии и эвристические методы – генетические алгоритмы, нейросети, оптимизированные уже относительно точности получаемого результата. Однако по-прежнему все эти методы имеют те или иные ограничения, которые понижают требования по вычислительным затратам, но отрицательно влияют на точность получаемых результатов. Использовать методы с гарантированным результатом, такие как полный перебор, до последнего времени было проблематично из-за ограничений по возможностям вычислительных средств.

Впрочем, уже с появлением первых компьютеров наиболее простые алгоритмы полного перебора (типа линейного всплывающего пузырька, его модифицированных версий «шейкер» и «расческа») стали применяться на практике. Впоследствии более мощные компьютеры позволили решать перебором более сложные задачи.

Таким образом, дилемма между точностью расчетов и возможностью реализации вычислительных методов, постоянно возникающая при разработке алгоритмов поддержки принятия решений и ранее однозначно разрешаемая в пользу огрубления результатов, в современных условиях получила перспективу уточнения.

Последнее определяется непрерывным ростом вычислительных мощностей и пропускной способности компьютерных сетей, которые позволяют снять часть ограничений на вычислительные потребности реализуемых алгоритмов. Примером могут служить методы Data Mining и Big Data, основанные на известных методах кластеризации, по-новому реализуемых на современных больших вычислительных мощностях [1]. При этом широко при-

меняемые в настоящее время нейросети при решении ряда задач используют алгоритмы интерполяции, обеспечивающие динамический, а не задаваемый пользователем статично, как это было раньше, выбор вида аппроксимационной функции. В отличие от классических методов это позволяет выбирать функцию, на основе которой проводится интерполяция, не вручную, а автоматически из множества ранее использованных при решении подобных задач. Раньше подбор аппроксимирующей функции осуществлялся пользователем вручную с использованием математических методов, основанных на критериях близости, а теперь – за счет подбора на примерах так называемого обучения. Такой же подход используется при распознавании нейросетями образов, когда в рамках обучения настраиваются параметры коэффициентов близости заданных характеристик изображения к эталонным. И подобных примеров нового использования классических алгоритмов множество.

Таким образом, как показал анализ предметной области, по мере развития информационных технологий постепенно появилась возможность использовать максимально простые, но дающие гарантированный результат алгоритмы, назовем их прямыми. Важно, что реализуются эти алгоритмы за счет многократно возросших возможностей средств обработки данных, а не упрощения процесса или снижения точности результата, как раньше.

В соответствии с законом Мура считается, что производительность процессоров возрастает вдвое каждые 24 месяца. Сначала рост шел вследствие увеличения количества транзисторов в кристалле микросхем, в настоящий момент – за счет многоядерности и распараллеливания процессов. Практика показывает, что значительную часть этого роста съедают затраты на визуализацию, реализуемую для взаимодействия программ с пользователем, однако тенденция непрерывного прироста мощности существует, что подтверждает возможность и перспективы использования прямых алгоритмов.

Таким образом, на основании сложившейся ситуации можно сформулировать гипотезу о том, что перспективным направлением развития математического аппарата поддержки принятия решений является переход к прямым алгоритмам.

В рамках углубления детализации прямые алгоритмы предлагается разделить на две группы:

– классические, обеспечивающие поиск конечного решения на основе прямых вычислений и логических операций, например, полного перебора, и предоставляющие готовый результат пользователю;

– комбинированные, позволяющие вычислять количественные параметры управления до значения, которое можно получить без потери точности (далее на основании представленных результатов решение принимает пользователь с использованием качественных оценок и на основе логических выводов).

Практика показывает, что классические прямые алгоритмы в той или иной степени уже реализуются современными технологиями: в методах перебора с ограничениями, в технологиях Big Data. В статье детально рассмотрена возможность реализации комбинированных подходов, применяемых в условиях неполной и динамически изменяющейся исходной информации, наиболее сложных для реализации прямых алгоритмов. На основе анализа сделан вывод о том, могут ли прямые алгоритмы применяться практически на всех диапазонах решения управленческих задач или данная тенденция является частным случаем.

### **Анализ требований к алгоритмам поддержки принятия решений**

Выясним, возможна ли при современном уровне развития технологий практическая реализация прямых алгоритмов для решения управленческих задач.

Известно, что начальные и последний этапы типового цикла управления предполагают формирование формализованного описания управляемой системы в состояниях «как есть», «ожидается к началу действий» и «требуемое для достижения цели».

Формирование подобного описания и поддержание его в актуальном состоянии – достаточно затратный по времени как обработки, так и передачи информации процесс. Для построения модели управляемой системы в рамках подобного описания требуется реализация разнообразных алгоритмов и методов агрегирования информации, ее формализации, визуализации и прогнозирования изменений. Современные средства автоматизации управления в большинстве случаев не обеспечивают формирование подобных моделей в динамичном режиме [2].

Сложность данного процесса усугубляется тем, что при управлении крупными распределенными эргатическими (организационно-

техническими) системами ЛПР практически никогда не наблюдает их визуально и вынуждено формировать модель управляемой системы теми или иными методами и средствами. На начальных этапах развития управления подобные модели были преимущественно логическими, основой их формирования являлись результаты сбора данных об обстановке, предоставляемые ЛПР в текстовом и графическом видах. По мере развития автоматизации визуальные модели стали оформляться в виде графических схем и электронных карт, дополняемых таблицами, графиками, диаграммами и другими компонентами, что, впрочем, не решало проблему создания моделей управляемых систем с требуемой функциональностью, вынуждая ЛПР по-прежнему достраивать их мысленно [3]. Как показывает практика, при использовании подобного алгоритма управления в определенных ситуациях часть информации может просто теряться в ходе формирования ЛПР мысленной модели, что вносит в нее искажения и может привести к принятию ошибочных решений. Из примеров подобных ситуаций можно сделать вывод, что проблема алгоритмизации визуализированного представления управляемых распределенных систем для выработки решений по управлению ими в полном объеме не решена и остается актуальной.

Следует отметить, что речь идет о процессе управления распределенными системами, размещенными на местности за пределами визуального обзора. Управляемые системы такого типа имеются во всех областях управления: экономического, военного, государственного. Для систем меньшего масштаба уже сейчас можно отметить практические примеры реализации аналогов комбинированных прямых алгоритмов управления на основе построения моделей визуализации с использованием технологий дополненной реальности (augmented reality, AR). С применением технологий AR осуществляется управление техническими системами, находящимися в прямой видимости оператора. Например, обеспечение ситуационной осведомленности водителя проецированием показателей состояния и параметров движения управляемого автомобиля на лобовое стекло, как это делается в системах HUD (Head-up display). Или более сложный алгоритм: когда программа строит и проецирует на экран или лобовое стекло коридор разрешенных траекторий, оператор просто выдерживает его, управляя техническим средством (автомобильный парктроник, авиационная система

обеспечения посадки и т.п.). Вполне возможно, что такой алгоритм найдет применение для управления более крупными и сложными системами.

### Некоторые варианты реализации прямых алгоритмов управления

В рамках анализа проблемы управления распределенными системами отметим, что к таким системам можно отнести любое крупное предприятие, группу предприятий или отрасль как управляемый объект, недоступный для постоянного визуального контроля. Для управления такой системой, как отмечалось ранее, приходится организовывать распределенный сбор информации, ее обобщение и визуализацию в виде графиков, таблиц, схем, условных знаков на картах (схемах). На основе анализа этих данных управленцами строится виртуальная модель, отображающая состояние и тренд поведения системы, с помощью которых оцениваются риски и угрозы, формируются управляющие воздействия и прогнозируются последствия их реализации. В доинформационную эпоху все эти операции выполнялись вручную, с привлечением большого числа управленцев и все равно были недостаточно эффективны с точки зрения как времени, так и полноты описания управляемой системы. По мере развития автоматизации управления, с внедрением АСУ различных типов ситуация начала меняться. Появилась возможность использовать затратные алгоритмы оптимизации на основе градиентных методов поиска экстремума, а также методы эвристической самоорганизации, в том числе генетические алгоритмы [4, 5]. Но даже в настоящее время, когда различные автоматизированные и информационные системы используются достаточно активно, проблема динамического формирования модели управляемой системы решена не в полном объеме. Значительная доля работы по формированию набора показателей состояния управляемой системы остается за ЛПР и проводится методом мысленного моделирования.

Более того, в рамках существующих концепций автоматизации управления автоматизации подлежат прежде всего расчетные и рутинные функции: сбор и агрегирование информации, формирование электронных управляющих и отчетных документов. Более сложные с точки зрения формализации задачи назначаются для выполнения человеком. Такой подход прагматично решает задачу автоматизации, но не поз-

воляет в полной мере использовать возможности современных информационных технологий для поддержки принятия решений, в том числе для автоматизации аналитических функций организации управления [6, 7].

В то же время существующие технологии потенциально позволяют существенно расширить перечень автоматизируемых функций, в том числе с реализацией прямых алгоритмов управления. Например, как отмечено ранее, для небольших по масштабу динамических систем повышение информационной осведомленности управленцев с одновременным выполнением ряда аналитических функций достаточно успешно решается с помощью технологий AR [1].

Однако многие сложные системы из-за своего масштаба не могут быть полноценно описаны средствами AR и требуют использования более мощных технологий, например, виртуальной реальности (virtual reality, VR). Прежде всего это распределенные человеко-машинные системы, к которым относится большинство современных систем управления предприятием, логистикой, распределением ресурсов и им подобных.

Возможные технологии реализации прямых алгоритмов комбинированного типа с применением средств VR в таких системах определяются структурой типового цикла управления [8, 9]: целеполагание, оценка обстановки, принятие решения, планирование, постановка задач, контроль выполнения, корректировка управляющих воздействий. В ходе реализации каждого цикла неизбежно возникают ошибки управления, требующие периодической корректировки управляющих параметров, определение которых является сложной научно-практической задачей.

При решении указанной задачи текущее положение управляемой системы в фазовом пространстве состояний чаще всего определяется вектором состояния вида  $C = \{ \{c_c\} \{c_m\} \{c_f\} \}$ , где  $C$  – семейство множеств, описывающих текущее состояние системы;  $c_c$  – множество параметров состояний, соответствующих контрольным параметрам вектора цели (параметрам, приводящим управляемую систему в требуемое состояние);  $c_m$  – множество управляемых параметров;  $c_f$  – множество свободных параметров [10].

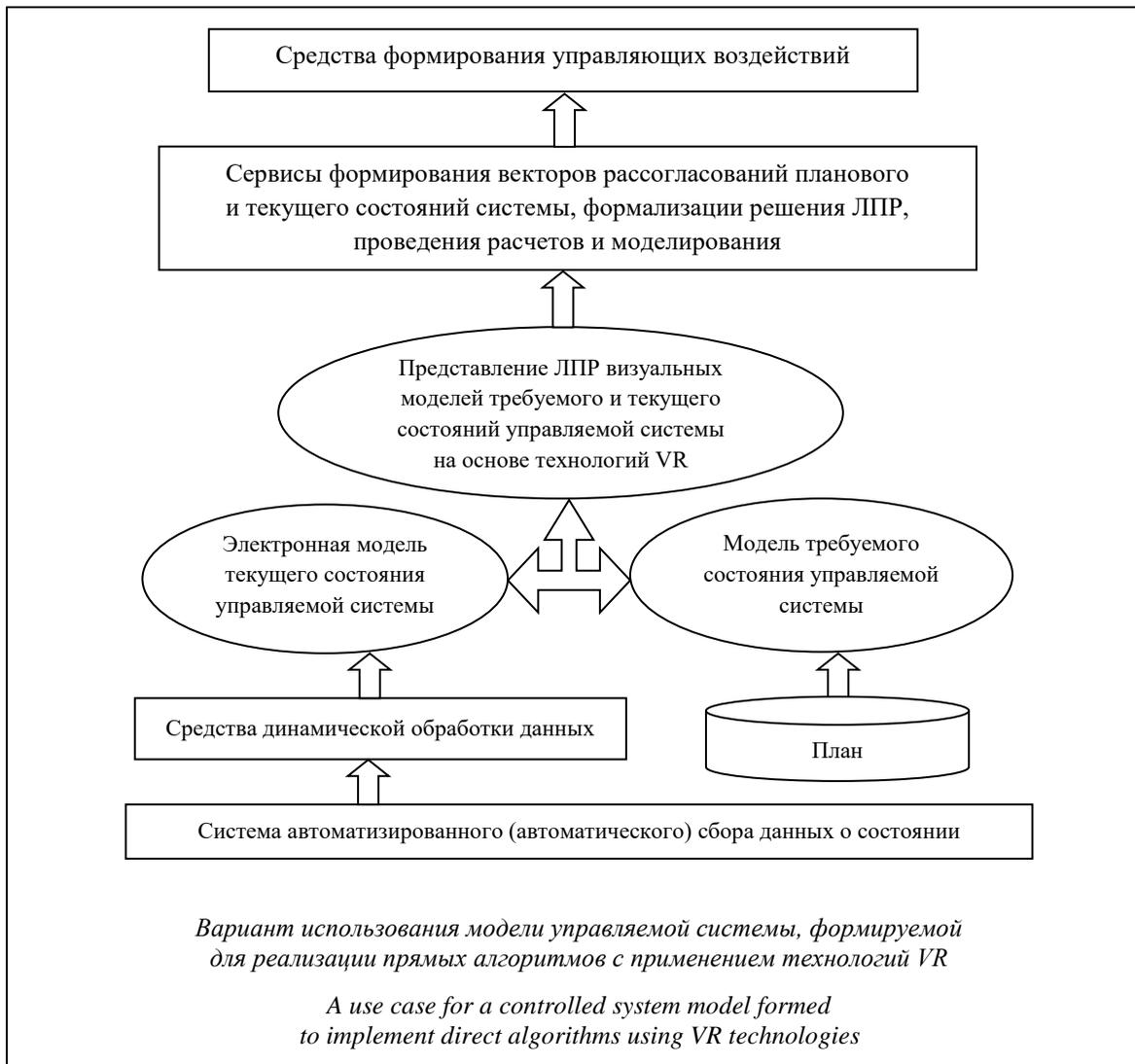
В такой постановке процесс управления понимается как последовательность формирования управленческих решений с параметрами  $c_m$ , обеспечивающими вывод системы в состояние  $c_c$  к установленному времени и с затра-

тами ресурса в рамках заданных ограничений. Основными задачами управления являются формирование и периодическая корректировка управляющих параметров, реализация которых приведет управляемый объект (систему) в требуемую точку фазового пространства, то есть обеспечит достижение цели.

В рамках реализации прямых алгоритмов управления комбинированного вида для обеспечения этого процесса с использованием технологий VR могут быть созданы несколько виртуальных моделей управляемой системы, наглядно представляющих ее текущее и требуемое состояния и визуализирующих ошибки управления. Такой подход обеспечит логичное разделение процесса формирования решений на этапы количественных расчетов, выполняемых автоматизированной системой, и качественных оценок, формируемых ЛПР (см. рисунок).

Представленная на рисунке последовательность реализации цикла управления обеспечивает выполнение несложного, но потенциально эффективного прямого алгоритма комбинированного вида:

- формирование в ходе планирования визуальной модели требуемого конечного состояния управляемой системы;
- описание набора моделей требуемого состояния управляемой системы с дискретностью по времени;
- описание на базе технологий VR текущего состояния управляемой модели и поддержание ее актуальности в автоматическом режиме на каждый момент времени  $t_m$ ;
- представление моделей ЛПР с наглядным указанием отклонений состояния;
- оценка, принятие решения ЛПР;
- расчет с использованием разработанных моделей новых значений управляемых параметров  $c_m^*$ .



Предлагаемый прямой алгоритм управления распределенными системами с применением технологий VR можно определить еще как сведение векторов состояния на визуальной модели, или алгоритм сведения. Он достаточно прост математически, хотя и затратен с точки зрения обработки графической информации и обмена данными. Но с учетом того, что большинство людей примерно две трети информации получают от органов зрения, данный алгоритм управления должен существенно упростить процесс принятия решений, оставив расчетные и моделирующие функции за системой и предоставив ЛПР информацию для принятия решений на обобщенном качественном уровне.

С учетом данных факторов использование прямого алгоритма типа «визуальный алгоритм сведения» потенциально обеспечит повышение динамичности этого процесса и адекватности реакции органов управления на изменение обстановки.

При оценке возможности реализации предлагаемого алгоритма можно отметить, что сами технологии его реализации уже существуют и используются в различных областях управления [11]. С точки зрения оценки эффективности любой реализуемый в *системах поддержки принятия решений* (СППР) алгоритм должен быть проверен на устойчивость, сходимость, точность и экономичность.

Сравнение по указанным параметрам прямых и обычных оптимизационных алгоритмов показывает следующее.

- В отличие от обычных алгоритмов, критично зависящих от чувствительности применяемых методов поиска решения к изменению исходных данных, прямые алгоритмы всех типов обладают высокой устойчивостью, что теоретически строго обоснования не имеет, но подтверждается обширной практикой [12, 13].

- Математически точного подтверждения гарантированной сходимости прямых алгоритмов, особенно комбинированных, нет, но на практике не отмечено ни одного случая, когда сходимость не была бы достигнута на любых конечных множествах в отличие от других алгоритмов, зависящих от используемых методов поиска решения [14, 15]. Следует отметить, что на практике скорость сходимости прямых алгоритмов в большинстве случаев ниже, чем у обычных.

- Точность классических прямых алгоритмов при корректном завершении является абсолютной в отличие от всех других алгоритмов, погрешность которых определяется задаваем-

мой заранее точностью округления формализуемых входных, промежуточных и выходных данных, количеством итераций вычисления и принятыми границами области завершения поиска решения. В комбинированных алгоритмах точность корректно может быть оценена только относительно их вычислительной части, и эта оценка совпадает с показателями классических алгоритмов. В части обычных алгоритмов существуют ситуации стопроцентной точности алгоритмов на основе аналитических методов вычисления, но это, скорее, исключение, чем правило. Обычно погрешность расчетов устанавливается с учетом затрат времени на поиск решения и принимается в пределах от долей до нескольких единиц процентов [16].

- Экономичность прямых и обычных алгоритмов определяется их структурой и должна оцениваться для каждого конкретного случая. В большинстве ситуаций для задач большой размерности асимптотический анализ алгоритмов показывает значительно большие затраты вычислительных ресурсов для варианта использования прямых алгоритмов, вычислительная емкость которых выше [15].

Таким образом, прямые алгоритмы в подавляющем большинстве случаев будут более затратными, чем все остальные. Впрочем, как отмечалось ранее, во-первых, в связи с непрерывным ростом вычислительных мощностей этот недостаток не является критичным, во-вторых, на задачах малой размерности даже классические прямые алгоритмы сопоставимы по затратам с обычными [16], не говоря уже о комбинированных. Кроме того, затратность компенсируется высокой точностью, устойчивостью и сходимостью.

Данные выводы подтверждаются опытом практического использования алгоритма сведения на виртуальной модели управляемой системы в группе компаний «Техносервь». Использование данного алгоритма при разработке программных средств оперативного управления логистикой показало его реализуемость и работоспособность [3]. Это подтверждает потенциальную применимость прямых алгоритмов управления в системах поддержки принятия решений, реализующих управление во всех диапазонах условий применения управляемой системы.

В целом же алгоритм сведения, относящийся к комбинированному типу прямых алгоритмов, является одним из направлений развития алгоритмизации СППР, отражающим общую тенденцию и дополняющим ее (см. таблицу).

**Тенденции развития алгоритмизации СППР**  
**Trends in the DMSS algorithmization development**

Тип алгоритмов	Ранее используемый подход	Современное состояние	Перспектива с учетом использования прямых алгоритмов
Алгоритмы поиска информации	Оптимизация относительно вычислительных мощностей с выполнением ограничений по максимальному времени поиска и минимально достаточной полноте найденных данных	Оптимизация по скорости поиска с обеспечением соотношения найденной и пропущенной информации не ниже заданного	Получение максимального количества необходимой информации за время поиска не выше заданного
Классические алгоритмы решения оптимизационных задач	Оптимизация относительно вычислительных мощностей с выполнением ограничений по максимальному времени расчета и максимально допустимой погрешности результата	Оптимизация по точности результата с ограничениями по времени решения и требованиям к вычислительной мощности	Оптимизация относительно точности результата с выполнением ограничений по времени его получения
Алгоритмы решения оптимизационных задач методами эвристической самоорганизации	Не были автоматизированы	Оптимизация относительно располагаемого времени для приемлемой точности результата	Оптимизация относительно точности результата с выполнением ограничений по времени его получения
Комплексные алгоритмы формирования управляющих воздействий	Не были автоматизированы	Формирование визуальных моделей на основе технологий AR с формированием реакции оператора	Формирование визуальных моделей на основе технологий AR и VR. Автоматическое формирование управляющих воздействий на основе качественного решения оператора

### Выводы

В таблице отражены особенности прямых алгоритмов управления всех типов относительно обычных:

- использование для принятия решений максимально возможного количества исходной информации без упрощения и отсека лишней информации;
- управление объектами и процессами по реальным параметрам с их минимальным преобразованием и формализацией;
- определение общих параметров управления практически без использования математических вычислений, на основе формальной логики.

Сравнительная оценка особенностей разных классов алгоритмов позволяет сделать вывод о том, что перспективой развития алгоритмического обеспечения СППР является использова-

ние именно прямых методов поиска решений, реализуемых в следующих группах алгоритмов:

- оптимизационные алгоритмы на основе полного перебора с условиями;
- алгоритмы поиска информации на основе полного перебора с кластеризацией;
- комбинированные количественно-качественные алгоритмы управления, примером которых являются алгоритмы сведения.

В таблице отражены только оптимизационные алгоритмы, которые невозможно свести к задачам, имеющим гарантированное решение [17]. Алгоритмы решения задач, которые могут быть сведены к аналитическим методам с гарантированным результатом, например, к линейной оптимизации, вероятно, без изменений будут использоваться в СППР и впредь. То есть речь идет не о замене математических методов алгоритмами полного перебора, а о пе-

реходе к использованию исключительно методов получения точного решения и только там, где их применение невозможно, к использованию прямых алгоритмов.

Таким образом, прямые алгоритмы не исключают, а дополняют применяемые в настоящее время методы управления, расширяя перечень инструментария поддержки принятия решения, во-первых, учитывая то, что практическая реализация указанных алгоритмов потенциально обеспечивает получение гарантированного решения практически для любых условий решения задач управления, во-вторых, исходя из того, что одной из существенных проблем использования современных методов формирования управляющих воздействий является проблема формализации: от выбора упрощенной модели для описания управляемой системы или процесса до регулируемой потери точности результата при формализации входных и выходных параметров управления.

Учитывая указанные положения, вероятно, в перспективе приоритет будет именно за прямыми алгоритмами, упрощающими разработку систем управления, устраняющими потери

точности при формализации и обеспечивающими максимальную вероятность получения гарантированного решения.

### Заключение

Таким образом, принятие гипотезы о том, что перспективой развития алгоритмов СППР являются прямые алгоритмы, позволяет обозначить пути решения научно-практической задачи выбора направлений развития средств автоматизированной поддержки принятия решений. Само предлагаемое решение по использованию прямых алгоритмов в СППР находится в современном тренде развития автоматизации управления – реализации относительно простых, но эффективных методов с гарантированной результативностью. Методы, которые раньше не использовались из-за недостаточных возможностей вычислительной техники, теперь могут быть успешно применены в практике управления сложными распределенными системами.

В случае, если сформулированная гипотеза подтвердится, данная дилемма может быть в значительной степени разрешена.

### Список литературы

1. Rodriguez M.Z., Comin C.H., Casanova D., Bruno O.M. et al. Clustering algorithms: A comparative approach. PLoS ONE, 2019, vol. 14, no. 1, art. e0210236. doi: 10.1371/journal.pone.0210236.
2. Тиханычев О.В. Виртуальная реальность и поддержка принятия решений // Прикладная информатика. 2019. Т. 14. № 4. С. 56–64.
3. Tikhanychev O.V. On the virtual reality technologies application to support decision-making. AIP Conf. Proc., 2021, vol. 2402, art. 050003. doi: 10.1063/5.0071321.
4. Свиридова Е.А. Генезис понятия искусственного интеллекта с точки зрения права в условиях реализации национального проекта «Цифровая экономика Российской Федерации» // Экономика. Налоги. Право. 2020. Т. 13. № 3. С. 152–160.
5. Morisse T. Intelligence artificielle, un nouvel horizon: pourquoi la france a besoin d'une culture du numérique? Comprendre et débattre ses enjeux technologiques, économiques, légaux et éthiques. Paris, Les Cahiers Lysias Publ., 2017, 117 p.
6. LaValle S.M. Virtual Reality. Cambridge University Press, 2017, 424 p.
7. Barocas S., Felten E., Huey J., Kroll J., Reidenberg J., Robinson D., Yu H. Accountable algorithms. University of Pennsylvania Law Review, 2017, vol. 165, pp. 633–705.
8. Гладков Л.А., Лейба С.Н., Тарасов В.Б. Разработка и программная реализация гибридного алгоритма решения оптимизационных задач автоматизированного проектирования // Программные продукты и системы. 2018. Т. 31. № 3. С. 569–580. doi: 10.15827/0236-235X.123.569-580.
9. Ульянов С.В., Рябов Н.В. Квантовый генетический алгоритм в задачах моделирования интеллектуального управления и суперкомпьютеринг // Программные продукты и системы. 2019. Т. 32. № 2. С. 181–189. doi: 10.15827/0236-235X.126.181-189.
10. Lai C.H., Lin S.H. Systems theory. In: The International Encyclopedia of Organizational Communication, 2017, pp. 1–18. doi: 10.1002/9781118955567.wbieoc203.
11. Варшавский П.Р., Ар Кар Мью, Шункевич Д.В. Применение методов классификации и кластеризации для повышения эффективности работы прецедентных систем // Программные продукты и системы. 2017. Т. 30. № 4. С. 625–631. doi: 10.15827/0236-235X.120.625-631.
12. Култыгин О.П., Лохтина И.В. Бизнес-аналитика как инструмент системы поддержки принятия решений // Прикладная информатика. 2021. Т. 16. № 1. С. 52–58. doi: 10.37791/2687-0649-2021-16-1-52-58.
13. Tengö M., Andersson E. Solutions-oriented research for sustainability: Turning knowledge into action. Ambio, 2022, vol. 51, no. 1, pp. 25–30. doi: 10.1007/s13280-020-01492-9.

14. Шевченко А.А., Шевченко А.В., Зрелова Д.П., Ульянов С.В. Когнитивные регуляторы: технологии мягких вычислений и информационно-термодинамический закон самоорганизации интеллектуального управления // Программные продукты и системы. 2023. Т. 36. № 1. С. 13–25. doi: 10.15827/0236-235X.141.0.

15. Емельянов В.Н. Численные методы: введение в теорию разностных систем. М.: Юрайт, 2018. 189 с.

16. Higham N.J. Accuracy and Stability of Numerical Algorithms. Philadelphia, SIAM Publ., 2016, 675 p.

17. Бибило П.Н., Ланкевич Ю.Ю. Экспериментальное сравнение эффективности алгоритмов оптимизации BDD-представлений систем булевых функций // Программные продукты и системы. 2020. Т. 33. № 3. С. 449–463. doi: 10.15827/0236-235X.131.449-463.

Software &amp; Systems

doi: 10.15827/0236-235X.142.388-397

2023, vol. 36, no. 3, pp. 388–397

### On one trend in the development of algorithms implemented in decision support systems

Oleg V. Sayapin  
Oleg V. Tikhanychev  
Angela A. Bezvesilnaya  
Sergey V. Chiskidov

#### For citation

Sayapin, O.V., Tikhanychev, O.V., Bezvesilnaya, A.A., Chiskidov, S.V. (2023) 'On one trend in the development of algorithms implemented in decision support systems', *Software & Systems*, 36(3), pp. 388–397 (in Russ.). doi: 10.15827/0236-235X.142.388-397

#### Article info

Received: 27.01.2023

After revision: 15.06.2023

Accepted: 20.06.2023

**Abstract.** The article considers the issues of implementing effective control algorithms for complex human-machine systems in modern conditions. The subject area analysis shows that recently, algorithms based on simple but reliable methods are increasingly used in management practice; their use was previously problematic due to high computational costs. The practical use of such methods has become possible due to the computer technology development and the growth of the bandwidth of data transmission networks. As an example of implementing such algorithms, the authors consider the methods for forming control actions using the control process visualization using virtual reality technologies. The relevance of the work is determined by the fact that in the conditions of control of dynamic spatially distributed ergodic systems, existing control methods and technologies do not always provide the required efficiency. At the same time, effective management is the key to the successful application of any systems. Based on the analysis of a typical control cycle and the shortcomings of the currently used control algorithms, the article synthesizes proposals for using models based on direct calculations and exhaustive search methods. As an example, it considers the use of models based on virtual reality tools for applying a visual algorithm reducing the current state of the controlled system to the required state to achieve a goal. The proposed algorithm logically complements the algorithms for solving optimization problems and information retrieval, being in the general trend of the development of algorithmization of decision support systems in favor of simplifying the methods used.

**Keywords:** decision support, augmented reality, virtual reality, algorithmic management support, management efficiency improvement

#### Reference List

1. Rodriguez, M.Z., Comin, C.H., Casanova, D., Bruno, O.M. et al. (2019) 'Clustering algorithms: A comparative approach', *PLoS ONE*, 14(1), art. e0210236. doi: 10.1371/journal.pone.0210236.

2. Tikhanychev, O.V. (2019) 'Virtual reality and decision support', *J. of Applied Informatics*, 14(4), pp. 56–64 (in Russ.).

3. Tikhanychev, O.V. (2021) 'On the virtual reality technologies application to support decision-making', *AIP Conf. Proc.*, 2402, art. 050003. doi: 10.1063/5.0071321.

4. Sviridova, E.A. (2020) 'The genesis of the concept of artificial intelligence from the legal perspective in the context of the national project "Digital economy of the Russian Federation"', *Economics. Taxes. Law*, 13(3), pp. 152–160 (in Russ.).

5. Morisse, T. (2017) *Intelligence artificielle, un nouvel horizon: pourquoi la france a besoin d'une culture du numérique? Comprendre et débattre ses enjeux technologiques, économiques, légaux et éthiques*. Paris: Les Cahiers Lysias Publ., 117 p.

6. LaValle, S.M. (2017) *Virtual Reality*. Cambridge University Press, 424 p.
7. Barocas, S., Felten, E., Huey, J., Kroll, J., Reidenberg, J., Robinson, D., Yu, H. (2017) 'Accountable algorithms', *University of Pennsylvania Law Review*, 165, pp. 633–705.
8. Gladkov, L.A., Leyba, S.N., Tarasov, V.B. (2018) 'A hybrid algorithm for solving optimization problems of computer-aided design and its software implementation', *Software & Systems*, 31(3), pp. 569–580 (in Russ.). doi: 10.15827/0236-235X.123.569-580.
9. Ulyanov, S.V., Ryabov, N.V. (2019) 'The quantum genetic algorithm in the problems of intelligent control modeling and supercomputing', *Software & Systems*, 32(2), pp. 181–189 (in Russ.). doi: 10.15827/0236-235X.126.181-189.
10. Lai, C.H., Lin, S.H. (2017) 'Systems theory', in *The International Encyclopedia of Organizational Communication*. Wiley-Blackwell Publ., pp. 1–18. doi: 10.1002/9781118955567.wbieoc203.
11. Varshavsky, P.R., Ar Kar Myo, Shunkevich, D.V. (2017) 'Classification and clustering methods for improving efficiency of case-based systems', *Software & Systems*, 30(4), pp. 625–631 (in Russ.). doi: 10.15827/0236-235X.120.625-631.
12. Kulygin, O., Lokhtina, I. (2021) 'Business intelligence as a decision support system tool', *J. of Applied Informatics*, 16(1), pp. 52–58 (in Russ.). doi: 10.37791/2687-0649-2021-16-1-52-58.
13. Tengö, M., Andersson, E. (2022) 'Solutions-oriented research for sustainability: Turning knowledge into action', *Ambio*, 51(1), pp. 25–30. doi: 10.1007/s13280-020-01492-9.
14. Shevchenko, A.A., Shevchenko, A.V., Zrelva, D.P., Ulyanov, S.V. (2023) 'Cognitive regulators: Soft computing technologies and the information-thermodynamic law of intelligent control self-organization', *Software & Systems*, 36(1), pp. 13–25 (in Russ.). doi: 10.15827/0236-235X.141.0.
15. Emelyanov, V.N. (2018) *Numerical Methods: An Introduction to the Theory of Difference Systems*. Moscow, 189 p. (in Russ.).
16. Higham, N.J. (2016) *Accuracy and Stability of Numerical Algorithms*. Philadelphia: SIAM Publ., 675 p.
17. Bibilo, P.N., Lankevich, Yu.Yu. (2020) 'Experimental investigation of effectiveness of algorithms for minimizing BDD representations of Boolean function syst', *Software & Systems*, 33(3), pp. 449–463 (in Russ.). doi: 10.15827/0236-235X.131.449-4633.

#### Авторы

**Саяпин Олег Викторович**<sup>1</sup>, д.т.н., профессор, o.saiapin@amchs.ru  
**Тиханычев Олег Васильевич**<sup>2</sup>, к.т.н., заместитель начальника отдела управления перспективных разработок, otihanychev@technoserv.com  
**Безвесильная Анжела Александровна**<sup>1</sup>, к.п.н., зав. кафедрой, a.bezvesilnaia@amchs.ru  
**Чискидов Сергей Васильевич**<sup>1</sup> к.т.н., профессор кафедры, доцент, s.chiskidov@amchs.ru

#### Authors

**Oleg V. Saiapin**<sup>1</sup>, Dr.Sc. (Engineering), Professor, o.saiapin@amchs.ru  
**Oleg V. Tikhanychev**<sup>2</sup>, Ph.D. (Engineering), Deputy Head of Department, otihanychev@technoserv.com  
**Angela A. Bezvesilnaya**<sup>1</sup>, Ph.D. (Education), Head of Chair, a.bezvesilnaia@amchs.ru  
**Sergey V. Chiskidov**<sup>1</sup>, Ph.D. (Engineering), Professor of Chair, Associate Professor, s.chiskidov@amchs.ru

<sup>1</sup> Академия гражданской защиты МЧС России, г. Химки, 141435, Россия

<sup>2</sup> «Т1 Интеграция», г. Москва, 111395, Россия

<sup>1</sup> Civil Defence Academy EMERCOM of Russia, Khimki, 141435, Russian Federation

<sup>2</sup> "T1 Integration Group", Moscow, 111395, Russian Federation

## Исследование свойств конкурентного двоичного дерева

В.Г. Грачев  
А.С. Ушаков

### Ссылка для цитирования

Грачев В.Г., Ушаков А.С. Исследование свойств конкурентного двоичного дерева // Программные продукты и системы. 2023. Т. 36. № 3. С. 398–413. doi: 10.15827/0236-235X.142.398-413

### Информация о статье

Поступила в редакцию: 10.05.2023

После доработки: 05.06.2023

Принята к публикации: 30.06.2023

**Аннотация.** В разрабатываемом программном обеспечении навигационных тренажеров «Система моделирования морских операций» реализована технология автоматического построения трехмерной модели глобального рельефа на базе алгоритма рекурсивного деления с локально адаптивным управлением детализацией. Алгоритм реализован полностью на графическом адаптере (GPU) и использует возможности массивной параллельной обработки данных в вычислительных шейдерах. Алгоритм рекурсивного деления строится на использовании двоичного дерева, однако классические двоичные деревья, явно описанные с использованием указателей, неприменимы для GPU-реализаций из-за архитектурных особенностей. Для обеспечения возможности параллельного выполнения алгоритма применена специализированная параллельная структура данных – конкурентное двоичное дерево. В статье описаны предпосылки к созданию структуры конкурентного двоичного дерева и рассматриваются этапы его построения от использования неявного двоичного дерева до представления в виде бинарного поля с последующим дополнением его редуцированной суммой листовых узлов. Теоретически обоснован объем оперативной памяти, необходимый для размещения конкурентного двоичного дерева заданной глубины. Выполнен анализ алгоритмической сложности построения конкурентного двоичного дерева и итерации по его листовым узлам. Приведены и проанализированы результаты сравнительных синтетических тестов производительности конкурентного двоичного дерева, выполненные на CPU и GPU, а также результаты его практического применения в системе моделирования морских операций.

**Ключевые слова:** конкурентное двоичное дерево, параллельная структура данных, алгоритм рекурсивного деления, оптимизация объема памяти, массивный GPU-параллелизм, система моделирования морских операций

В разрабатываемом перспективном отечественном кроссплатформенном ПО навигационных тренажеров для практической подготовки специалистов ВМФ «Система моделирования морских операций» (СММО) [1] принята парадигма глобального района тренировки, которая подразумевает возможность проведения тренировок в любом районе земного шара [2]. При этом конкретный географический район может быть детализирован до необходимой степени визуального соответствия реальному. Основной составной частью любого района тренировки, в том числе и глобального, является массив суши, на котором располагаются другие объекты – строения, причальное оборудование, навигационные знаки и др. Если конкретные районы тренировок детализируются, как правило, вручную с использованием труда дизайнеров, то для глобального района, то есть для всего остального пространства земной поверхности вне конкретных районов, ввиду его обширной протяженности необходимо реализовать процесс автоматического или автоматизированного создания и отображения недетализированных массивов суши.

Эта задача решается в СММО при помощи алгоритма локально адаптивного управления

детализацией протяженного рельефа на базе рекурсивного деления с использованием вычислительных возможностей современных графических адаптеров (GPU). Практическая реализация этого алгоритма требует использования специализированного представления данных, допускающих их параллельную обработку.

### Структурирование данных для алгоритмов рекурсивного деления

Алгоритмы рекурсивного деления демонстрируют быстрый рост вычислительной сложности в зависимости от глубины рекурсии. Из-за фундаментально экспоненциального характера этих алгоритмов вычислительные затраты с увеличением глубины подразделения могут быстро стать ограничением. Один из путей амортизации возрастающей вычислительной сложности – выполнять рекурсивное деление не только адаптивно, но и параллельно. Хотя адаптивное подразделение легко реализовать последовательно, в общем случае совместить его с параллельной обработкой непросто. Решить эту проблему можно путем внедрения структуры данных, подходящей для параллельной обработки.

Канонический алгоритм рекурсивного деления строится на использовании двоичного дерева, количество узлов которого удваивается на каждом шаге рекурсии. Листовые узлы дерева формируют его изменяющуюся топологию. Если добиться возможности параллельной обработки листовых узлов двоичного дерева, то можно ускорить любой алгоритм подразделения.

В технической литературе описаны способы параллельного построения BVH-деревьев. В работе [3] представлен алгоритм построения иерархий ограничивающих объемов на многоядерных GPU. Авторы [4] устранили для этого алгоритма необходимость синхронизации данных между CPU и GPU, а автор [5] улучшил масштабируемость построения дерева в целом, тем самым максимизировав параллелизм. В работе [6] предложено повышение утилизации GPU путем кодирования листовых узлов Мортон-кодами для кривой Z-порядка. Однако специфические структуры данных BVH-деревьев неприменимы в алгоритмах рекурсивного деления, так как не обеспечивают изменение своей топологии с течением времени.

Классическая реализация двоичного дерева использует указатели для хранения связей между его узлами. Однако архитектура GPU не предусматривает работу с ними.

Известно, что конкурентное двоичное дерево (КДД) – это структура данных, предназначенная для построения и параллельного обновления топологии полного двоичного дерева,

в том числе с использованием возможностей GPU [7, 8]. В оригинальной статье [7] КДД описаны постфактум без рассмотрения этапов создания и строгого математического анализа их свойств. В настоящей работе теоретически исследованы все этапы построения КДД и алгоритмы его использования.

### Совершенные двоичные деревья как двоичная куча

**Свойства и определения.** Двоичное дерево, в котором все внутренние узлы имеют по два дочерних узла, а все листовые находятся на одном уровне, формирует так называемое совершенное двоичное дерево. В нем суммарно  $2^{D+1} - 1$  узлов, где  $D \geq 0$  – уровень листовых узлов (глубина дерева). На рисунке 1 представлена структура совершенного двоичного дерева глубиной  $D = 4$ .

Двоичное дерево тесно связано с таким понятием, как двоичная куча – полным двоичным деревом, в котором значение в любом узле не меньше, чем значения его потомков, а глубина всех узлов отличается не более чем на единицу. Топология совершенного двоичного дерева может быть представлена в виде двоичной кучи. Индексация двоичной кучи выполняется по следующему принципу: корневому узлу дерева присваивается индекс 1, и далее выполняется обход дерева в ширину с инкрементальным увеличением индексов узлов.

**Алгебраические зависимости.** Такой подход к индексации обладает следующими свой-

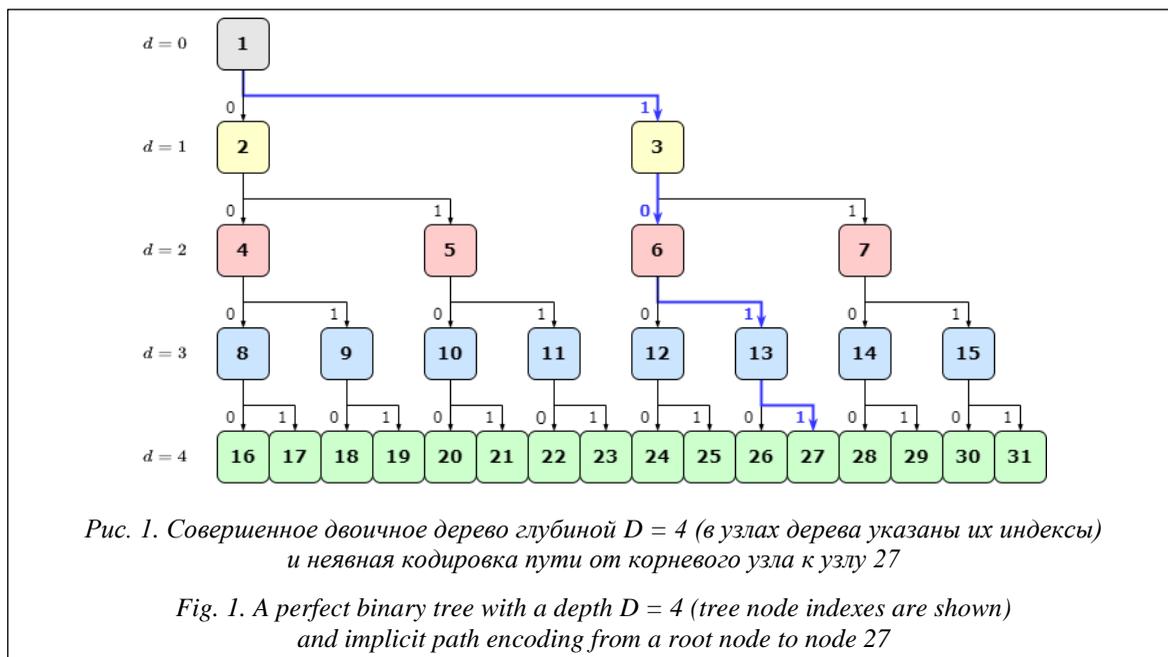


Рис. 1. Совершенное двоичное дерево глубиной  $D = 4$  (в узлах дерева указаны их индексы) и неявная кодировка пути от корневого узла к узлу 27

Fig. 1. A perfect binary tree with a depth  $D = 4$  (tree node indexes are shown) and implicit path encoding from a root node to node 27

ствами. Для узла дерева с индексом  $k \geq 1$  индекс его родительского узла  $k_p$ , левого  $k_L$  и правого  $k_R$  потомков будут следующими:

$$\begin{aligned} k_p &= \lfloor k/2 \rfloor, \\ k_L &= 2k, \\ k_R &= 2k+1. \end{aligned} \tag{1}$$

Для узлов дерева на уровне  $d \geq 0$  диапазон их индексов будет находиться в промежутке  $[2^d, 2^{d+1} - 1]$ .

Из индекса узла можно извлечь дополнительную информацию – номер уровня дерева, на котором узел находится:

$$d_k = \lfloor \log_2(k) \rfloor. \tag{2}$$

Для машинного двоичного представления положительного целого числа эта операция по результату эквивалентна нахождению в нем позиции старшего значащего бита:

$$\lfloor \log_2(k) \rfloor = \text{FindMSB}(k),$$

где *FindMSB* – функция, возвращающая позицию старшего значащего бита (справа налево, считая крайнюю правую позицию нулевой).

**Неявная кодировка пути к узлу.** Индекс узла двоичного дерева неявно кодирует путь к нему. Бинарное представление любого индекса двоичной кучи содержит весь путь от корневого узла до узла с этим индексом. При этом ноль обозначает переход от узла к его левому потомку, а единица – к правому. Путь предвзят установленным в единицу старшим битом, причем позиция бита равна уровню дерева, на котором находится этот узел (2). Например, узел с индексом 27 имеет бинарное представление индекса 11011. Смещение старшего бита  $d_{27} = 4$ , а путь к узлу 1011 или «направо-налево-направо-направо» от корневого узла (рис. 1).

**Неявная двоичная куча.** Двоичная куча может быть эффективно закодирована с использованием неявной структуры данных, в которой структурная информация неявно определяется способом хранения данных, а не прямым образом через указатели [9]. Для случая двоичной кучи это одномерный массив, содер-

жащий значения узлов дерева. Размер этого массива фиксированный и равен  $2^{D+1}$ . В первом элементе массива с индексом 0 хранится глубина дерева  $D$ . Оставшиеся элементы массива ассоциированы с каждым узлом дерева в соответствии с индексом. Корневой узел дерева хранится в элементе массива с индексом 1, за ним следуют два узла первого уровня, затем четыре узла второго уровня и т.д. На рисунке 2 совершенное двоичное дерево глубиной  $D = 4$  закодировано неявным образом в одномерный массив.

Благодаря такому представлению двоичное дерево может быть размещено в непрерывной области оперативной памяти с тривиальной итерацией по его узлам. Отсутствие в используемой неявной структуре данных указателей позволяет реализовать алгоритмы работы с ней на GPU.

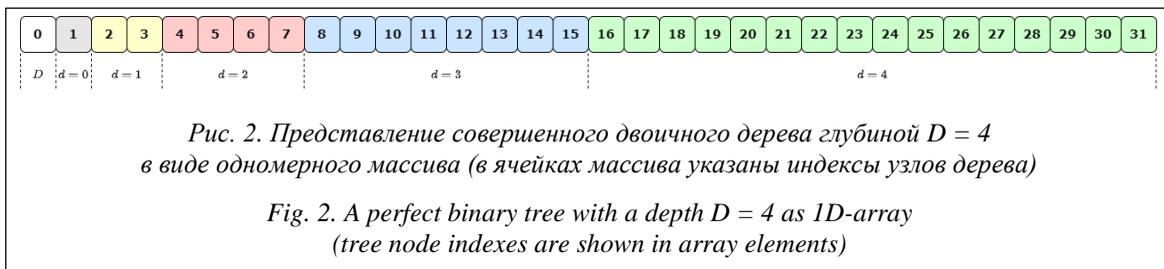
Фундаментальный недостаток двоичной кучи состоит в том, что ее применение ограничено совершенными двоичными деревьями. Несмотря на то, что можно изменять значения любого узла двоичного дерева, представленного в виде двоичной кучи, топология такого дерева остается неизменной.

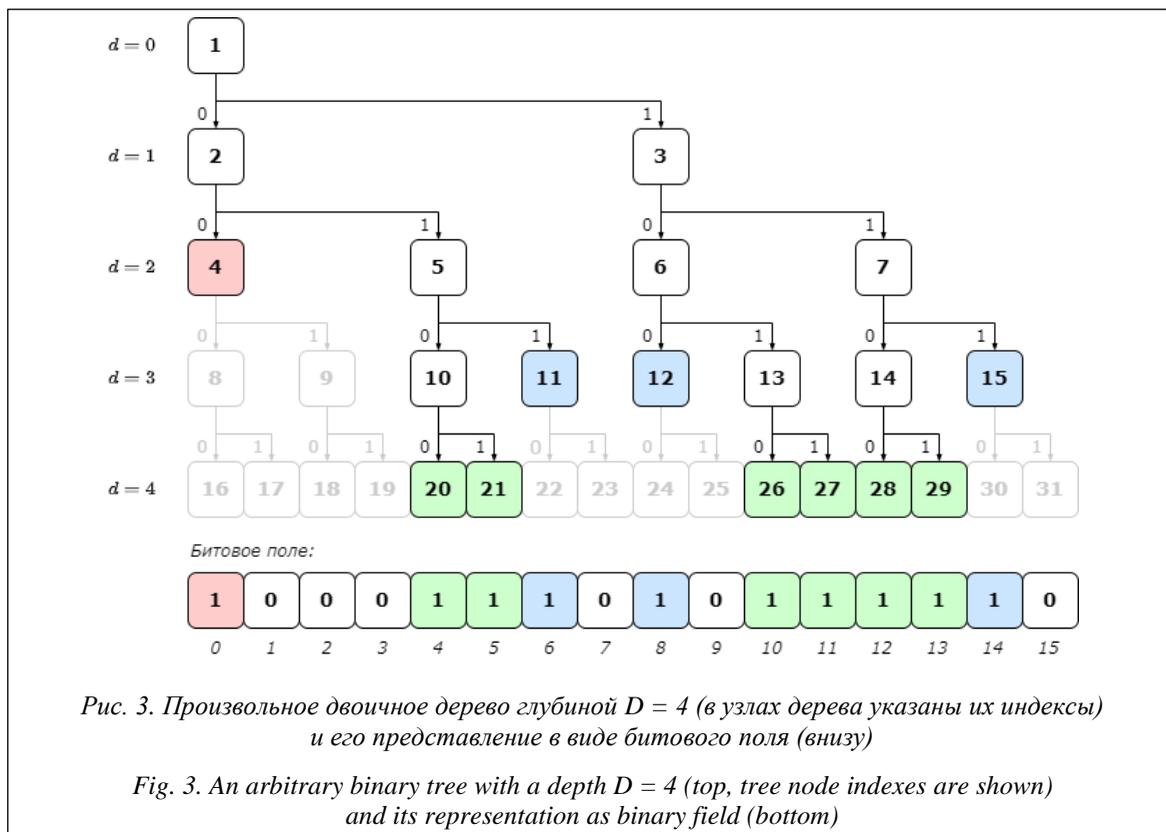
КДД предоставляет возможность обойти это ограничение путем представления произвольного двоичного дерева без использования указателей таким образом, что топология этого дерева может изменяться после его первичного формирования.

### Структура КДД

**Представление бинарных деревьев в виде битового поля. Создание битового поля.** Двоичное дерево глубиной  $D$  может быть представлено в виде битового поля размером  $2^D$ , содержащего по одному биту для каждого узла двоичного дерева на максимальном уровне  $d = D$ . На рисунке 3 представлены произвольное двоичное дерево глубиной  $D = 4$  и соответствующее ему битовое поле.

В таком битовом поле каждый бит, установленный в единицу, обозначает лиственный узел





дерева. При этом уровень листового узла определяется количеством нулей, следующих за соответствующим ему битом.

Раскодирование узлов дерева из битового поля представляет собой процесс получения индексов этих узлов  $k \in [1, 2^{D+1} - 1]$ . Бинарное представление индекса каждого узла двоичного дерева неявно кодирует путь к нему – таким образом достигается доступ ко всей топологии двоичного дерева.

*Индекс узла дерева из индекса бита.* Существует однозначное преобразование индекса бита  $x$  в битовом поле, кодирующем листовый узел произвольного двоичного дерева глубиной  $D$ , в индекс этого узла  $k$ .

Рассмотрим бит, находящийся в битовом поле по индексу  $x \in [0, 2^D - 1]$ . Если бит имеет значение ноль, то соответствующий ему узел дерева не является листовым и не существует в топологии дерева. Когда значение бита равно единице, он может кодировать до  $N_x$  листовых узлов дерева.

Для узлов дерева на уровне  $d$  соответствующие им биты в битовом поле располагаются через каждые  $2^{D-d}$  позиций. Следовательно, в зависимости от позиции бит может кодировать несколько узлов дерева. Если последовательно рассматривать битовое поле справа налево, то

можно увидеть, что каждый бит кодирует узел уровня  $d = D$ , каждый второй бит дополнительно кодирует узел на уровне  $d = D - 1$ , каждый четвертый бит дополнительно к ним кодирует узел на уровне  $d = D - 2$  и т.д. Таким образом, один и тот же бит  $x$  битового поля может кодировать до  $N_x = D + 1$  узлов дерева.

Для определения значения  $N_x$  для бита с индексом  $x$  необходимо найти максимальное значение  $n \in [0, D]$ , при котором выполняется условие

$$(2^D - x) \bmod 2^n = 0, \tag{3}$$

то есть определить максимальную степень двойки, на которую без остатка делится дополнение индекса бита до размера битового поля. Тогда количество кодируемых битом узлов дерева будет

$$N_x = n + 1. \tag{4}$$

Учитывая, что размер битового поля представляет собой степень двойки, при нахождении остатка от деления индекса бита на ту или иную степень двойки можно в (3) вместо дополнения индекса до размера битового поля использовать непосредственно сам индекс:

$$x \bmod 2^n = 0. \tag{5}$$

Если рассматривать двоичное машинное представление положительного целочислен-

ного индекса бита  $x$ , то максимальное значение  $n$  в (5) будет соответствовать позиции младшего значащего бита. Тогда

$$N_x = \begin{cases} D+1, & x = 0, \\ \text{FindLSB}(x)+1, & x > 0, \end{cases} \quad (6)$$

где  $\text{FindLSB}$  – функция, возвращающая позицию младшего значащего бита (справа налево, считая крайнюю правую позицию нулевой).

Индексы  $N_x$  узлов дерева, кодируемых битом  $x$ , определяются как

$$\{k_1, \dots, k_{N_x}\} = (2^D + x)2^{-n_x}, \quad (7)$$

$$n_x \in [0, N_x - 1].$$

Множество  $\{k_1, \dots, k_{N_x}\}$  содержит индексы узлов, находящихся в зависимости «потомок–родитель», начиная с узла на уровне  $d_k = D$  до узла на уровне

$$d_k = D - (N_x - 1).$$

С учетом (4)

$$d_k = D - (n_x + 1 - 1) = D - n_x. \quad (8)$$

Для приведенного на рисунке 3 примера бит с индексом 4 имеет машинное представление 100. Наименьший значащий бит находится на позиции 2. Следовательно, в соответствии с (6) этот бит может кодировать три узла дерева, находящиеся на различных уровнях. Индексы узлов определяются по (7) и равны  $\{20, 10, 5\}$ . Аналогично биты с индексами 0 и 14 могут соответственно кодировать до пяти и до двух узлов с индексами  $\{16, 8, 4, 2, 1\}$  и  $\{30, 15\}$ .

Чтобы определить, какой именно узел дерева из допустимого диапазона индексов узлов кодирует бит, необходимо установить уровень дерева, на котором располагается кодируемый узел.

Если узел  $k$  на уровне дерева  $d_k$  является листовым и кодируется единичным битом с индексом  $x_k$ , то следующий единичный бит  $x_{k'}$  в битовом поле будет находиться на позиции

$$x_{k'} = x_k + 2^{D-d_k}$$

Отсюда следует, что в битовом поле за единичным битом, кодирующим листовую узел на уровне дерева  $d_k$ , будут следовать нулевые биты в количестве, равном  $N_0 = 2^{D-d_k} - 1$ .

Тогда

$$N_0 + 1 = 2^{D-d_k},$$

$$\log_2(N_0 + 1) = D - d_k$$

и уровень кодируемого узла в двоичном дереве определяется как

$$d_k = D - \log_2(N_0 + 1). \quad (9)$$

Из (8) выразим

$$n_x = D - d_k.$$

С учетом этого выражение (7) преобразуется:

$$k_x = (2^D + x)2^{-(D-d_k)}. \quad (10)$$

Принимая во внимание (9), индекс узла дерева, кодируемый битом  $x$ , можно определить как

$$k_x = (2^D + x)2^{-\log_2(N_{0x}+1)}.$$

Так как каждый бит в битовом поле может кодировать только по одному узлу на каждом уровне дерева, битовое поле однозначно отображает индексы битов в индексы узлов двоичного дерева произвольной топологии.

*Индекс бита из индекса узла дерева.* Для любого узла дерева  $k$  можно определить кодирующий его бит, расположенный в битовом поле на позиции  $x_k$ .

Решая уравнение (10) относительно  $x$  с учетом известного из (2)  $d_k$ , получаем

$$x_k = k2^{D-d_k} - 2^D.$$

Смысл этого выражения в следующем. Так как бит  $x$  в битовом поле может кодировать  $N_x$  узлов, требуется привести индекс узла  $k$  к максимальному из значений множества индексов узлов, которые могут быть закодированы соответствующим битом. Для этого необходимо рекурсивно вычислять индекс левого потомка узла  $k$  вплоть до последнего уровня дерева  $d = D$ , на котором для кодирования узла требуется один бит. Это можно сделать, последовательно используя выражение (1)  $D - d_k$  раз. Затем полученный индекс следует уменьшить на индекс первого узла дерева на уровне  $D$ , то есть на  $2^D$ .

Для приведенного на рисунке 3 примера узел с индексом 5 имеет самого глубокого потомка на уровне  $D$  с индексом 20. Этот индекс необходимо уменьшить на индекс первого узла на уровне  $D$ , то есть на 16. Соответствующий узлу бит будет иметь индекс 4.

*Изменение топологии дерева.* Представление двоичного дерева в виде битового поля позволяет реализовать операции разделения и слияния узлов дерева напрямую.

Разделение заданного узла двоичного дерева с индексом  $k$  на левый и правый потомки означает установку в единицу бита битового поля, соответствующего правому потомку этого узла с индексом  $2k + 1$ .

Правый и левый потомки одного родительского узла с индексом  $k$  могут быть объединены в родительский узел. Эта операция означает установку в ноль бита битового поля, со-

ответствующего правому потомку родительского узла с индексом  $2k + 1$ .

Пример изменения топологии двоичного дерева и его представления в виде битового поля в результате операций слияния и разделения листовых узлов показан на рисунке 4.

Этот метод инвариантен к типу обрабатываемого узла. Допускается разделение нелистового узла. В этом случае для бита, соответствующего правому потомку такого узла, будет повторно установлено значение единица. При слиянии несуществующих узлов для бита, соответствующего несуществующему узлу, будет повторно установлено значение ноль.

**Итерация по листовым узлам.** Для получения из битового поля всех индексов листовых узлов дерева необходимо последовательно проверить значения всех его битов. Сложность такого алгоритма составляет  $O(2^D)$  [10]. Она не зависит от количества листовых узлов  $L \in [1, 2^D]$  в закодированном двоичном дереве. Блок-схема алгоритма представлена на рисунке 5.

Проверка значений битов в битовом поле должна выполняться последовательно, начиная с нулевого индекса, что исключает возможность параллельного поиска индексов листовых узлов.

**Тесты производительности декодирования битового поля.** Для сравнительной оценки производительности структуры КДД на CPU и GPU была выполнена его программная реализация на языках C++ и GLSL [11].

Тесты выполнялись на ПК, оснащенный центральным процессором Intel Core i7-11700 и графическим адаптером NVIDIA 3080 RTX

с 8704 универсальными процессорами, под управлением Windows 10.

Замеры производительности декодирования битового поля для КДД глубиной от 10 до 30 приведены на рисунке 6.

Архитектурные особенности GPU, выполнение алгоритма с использованием одного потока, а также эффективное задействование кеша центрального процессора и упреждающей выборки данных при последовательном обращении к линейному участку памяти битового поля приводят к существенному преимуществу CPU перед GPU [12]. Разница во времени декодирования составляет примерно 50 раз.

Полученные результаты существенно ограничивают практическое применение метода итерации по листовым узлам дерева путем декодирования битового поля.

Дополнительным ограничивающим фактором является необходимость отдельно от самого КДД хранить получаемые индексы декодированных листовых узлов для дальнейшего использования в алгоритме рекурсивного деления. Максимальное количество листовых узлов КДД равно  $2^D$ , номер каждого узла на GPU хранится в переменной типа uint, занимающей в памяти четыре байта (32 бита), то есть для записи всех возможных индексов листовых узлов необходимо дополнительно выделить  $4(2^D)$  байт, что в 32 раза больше, чем объем памяти самого битового поля. Это контрпродуктивно в условиях существующих ограниченных ресурсов видеопамати.

**Дополнение битового поля суммой листовых узлов.** Редуцированная сумма листовых

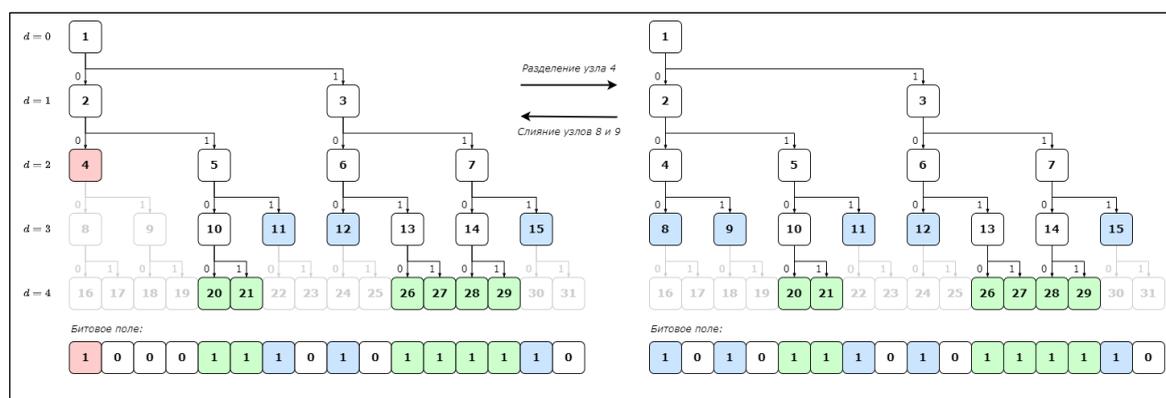


Рис. 4. Изменение битового поля при операциях разделения (слева направо) и слияния (справа налево) узлов двоичного дерева (в ячейках указаны значения редуцированной суммы листовых узлов, в верхнем левом углу ячеек – индексы узлов)

Fig. 4. Bitfield variation during binary tree nodes split (left-to-right) and merge (right-to-left) operations. Leaf nodes sum reduction values are shown in elements and node indexes are shown in elements left top corners

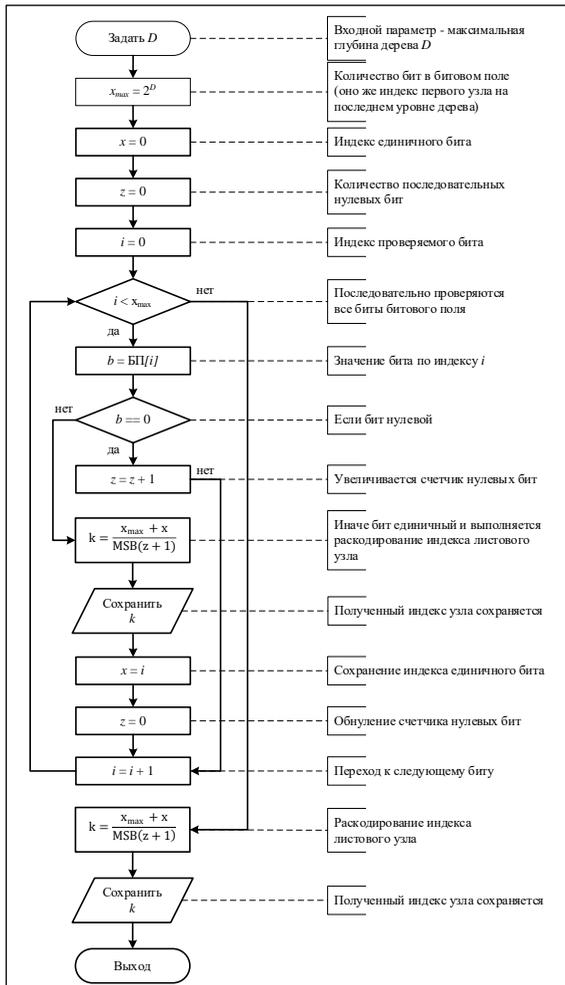


Рис. 5. Блок-схема алгоритма поиска индексов листовых узлов двоичного дерева путем декодирования битового поля

Fig. 5. A block diagram of a bitfield decoding algorithm for findumn binary tree leaf node indexes

узлов. Представление двоичного дерева в виде битового поля позволяет полностью закодировать топологию дерева и выполнять операции разделения и слияния узлов. Тем не менее битовому полю недостает эффективных методов итерации по закодированным в нем узлам дерева. Эта возможность особенно важна для больших битовых полей, так как их размер экспоненциально растет с ростом глубины двоичного дерева и простой однопоточный последовательный перебор битов на GPU будет выполняться за время, недопустимо большое при практическом использовании.

Для обеспечения эффективной итерации по листовым узлам двоичного дерева представляющее это дерево битовое поле дополняется его

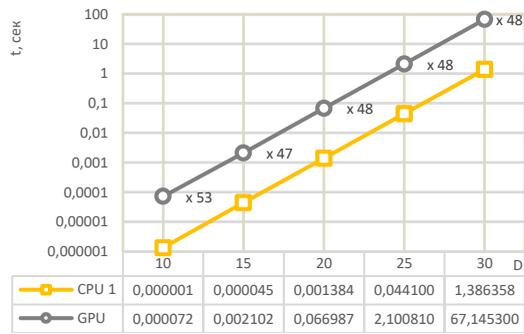


Рис. 6. Время декодирования битового поля КДД в зависимости от его глубины

Fig. 6. Concurrent binary tree bitfield decoding time depending on the tree depth

редуцированной суммой. Этот процесс может быть рассмотрен как построение снизу вверх совершенного двоичного дерева глубиной  $D$ , в котором каждый листовой узел соответствует биту битового поля (рис. 7).

По значению редуцированной суммы левого и правого потомков узла такого дерева можно определить, какое количество листовых узлов содержится в левом и правом поддеревьях узла.

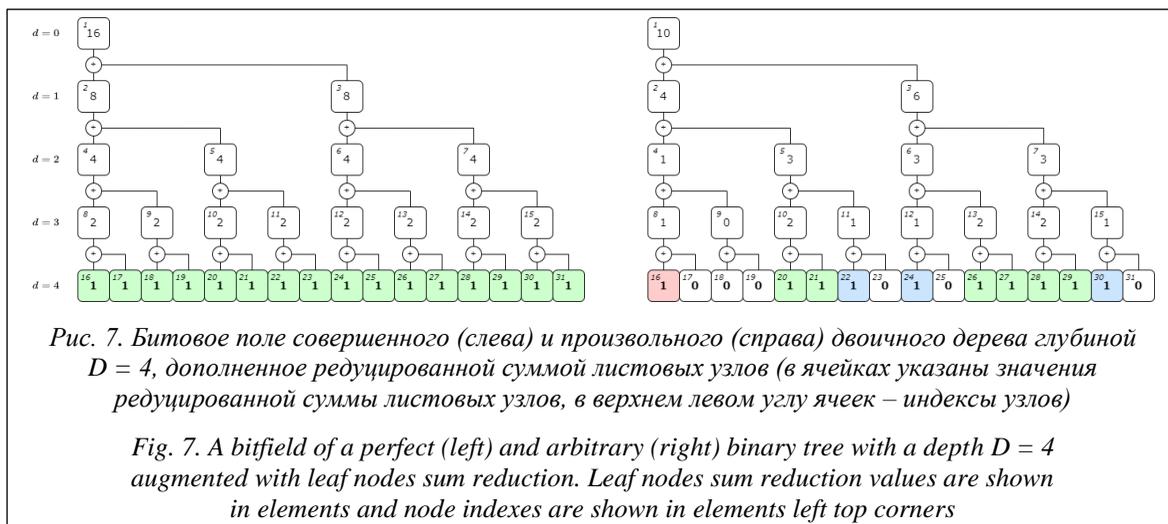
В свою очередь, такое совершенное дерево может быть представлено в виде двоичной кучи, в которой значения узлов хранятся в порядке обхода дерева в ширину начиная от корневого узла. Эта двоичная куча, включая битовое поле, являющееся его последним уровнем, представляет собой структуру данных КДД.

В элементе двоичной кучи с индексом 0 хранится глубина КДД  $D$ . Значение глубины КДД принимается в момент его построения и в дальнейшем не может быть изменено.

Второй элемент кучи с индексом 1 хранит общее количество листовых узлов дерева  $L \in [1, 2^D]$ , полученное в результате вычисления редукции суммы. Это свойство структуры КДД является отправным для выполнения итераций по листовым узлам дерева.

Индекс листового узла дерева кучи по его порядковому номеру. Индекс узла КДД равен индексу соответствующего ему элемента двоичной кучи. В КДД, содержащем  $L$  листовых узлов, для каждого  $l$ -го листового узла, где  $l \in [0, L - 1]$ , можно определить индекс соответствующего ему элемента двоичной кучи.

Алгоритм поиска индекса узла представляет собой рекурсивный переход от корневого узла дерева к его левому или правому поддереву в зависимости от того, превышает ли по-



рядковый номер листового узла количество листовых узлов в левом поддереве. Блок-схема алгоритма приведена на рисунке 8.

**Итерация по листовым узлам.** Сложность алгоритма поиска индекса одного листового узла КДД по его порядковому номеру в худшем случае составляет  $O(D)$ . При поиске всех листовых узлов совершенного дерева  $L_D \in [1, 2^D]$  эта сложность возрастает до  $O(DL_D) = O(D2^D)$ .

Полученная оценка сложности алгоритма поиска всех листовых узлов КДД в  $D$  раз превосходит сложность поиска листовых узлов по бинарному дереву. Однако КДД является параллельной структурой данных и допускает использование до  $P \in [1, L]$  потоков, одновременно выполняющих поиск индексов листовых узлов. С учетом этого сложность многопоточного алгоритма составит  $O(D2^D/P)$ . Для современных GPU  $P \gg D$ .

Условием завершения алгоритма поиска индекса  $l$ -го листового узла является количество потомков  $k$ -го узла, меньшее или равное единице. В свою очередь, как было показано ранее, изменение топологии КДД приводит к изменению значения необходимого бита битового поля на ноль (в случае слияния узлов) либо на единицу (в случае разделения узла). При этом происходит неявное изменение значений сумм листовых узлов только на последнем уровне дерева  $d = D$ , где расположены биты битового поля, непосредственно кодирующие листовые узлы. Так как это значение для последнего уровня КДД не может превышать единицу, условие завершения алгоритма поиска листовых узлов продолжает выполняться. Это дает возможность одновременно с поиском листовых узлов выполнять их слияние или разделе-

ние, таким образом изменяя топологию дерева. К тому же отпадает необходимость промежуточного хранения индексов листовых узлов и выделения для этого дополнительного объема памяти.

**Размещение структуры данных в памяти.** Потребный объем памяти КДД. Определим минимальный размер памяти, необходимый для размещения КДД глубиной  $D$ . Для этого найдем верхнюю границу значения редуцированной суммы для узлов дерева на произвольном уровне  $d$  и, как следствие, минимальное количество бит для кодирования этого значения.

Любое значение узла КДД (редуцированная сумма его потомков, являющихся листовыми узлами) на глубине  $d \in [0, D]$  требует не более  $N_{b_d} = D - d + 1$  бит для записи.

На последнем уровне дерева в битовом поле признак листового узла кодируется одним битом. Каждые два смежных узла имеют общий родительский узел на предыдущем уровне, который может принять максимальное значение 2. Дальнейшая редуция суммы будет удваивать это значение для каждого предшествующего уровня двоичного дерева. Следовательно, максимальное значение редуцированной суммы листовых узлов на уровне  $d$  будет

$$S_{MAX_d} = 2^{D-d}.$$

Количество бит для записи значения  $S_{MAX_d}$  равно

$$N_{b_d} = \lceil \log_2(S_{MAX_d}) \rceil + 1, \tag{11}$$

$$N_{b_d} = D - d + 1.$$

Количество узлов двоичного дерева на уровне  $d \in [0, D]$  составляет

$$N_{l_d} = 2^d.$$

Общее количество памяти, необходимое для хранения узлов КДД глубиной  $D$ , будет равно сумме произведений количества узлов на каждом уровне на количество бит, необходимых для записи их значений:

$$N_b = \sum_{d=0}^D N_{l_d} N_{b_d}$$

или

$$N_b = \sum_{d=n}^D 2^d (D - d + 1). \tag{12}$$

Для определения этого значения найдем частную сумму числового ряда (12). Представим сумму ряда как

$$N_b = D \sum_{d=n}^D 2^d - \sum_{d=n}^D d 2^d + \sum_{d=n}^D 2^d. \tag{13}$$

Обозначим входящие в это выражение частные суммы как

$$S_1 = \sum_{d=n}^D 2^d,$$

$$S_2 = \sum_{d=n}^D d 2^d$$

и последовательно определим их.

Частная сумма ряда  $S_1$  от  $n$  до  $D$  равна разности между частной суммой этого ряда от нуля до  $D$  и частной суммой от нуля до  $n - 1$ :

$$S_1 = \sum_{d=0}^D 2^d - \sum_{d=0}^{n-1} 2^d. \tag{14}$$

Обозначим уменьшаемое в (14) как

$$S_{11} = \sum_{d=0}^D 2^d,$$

а вычитаемое как

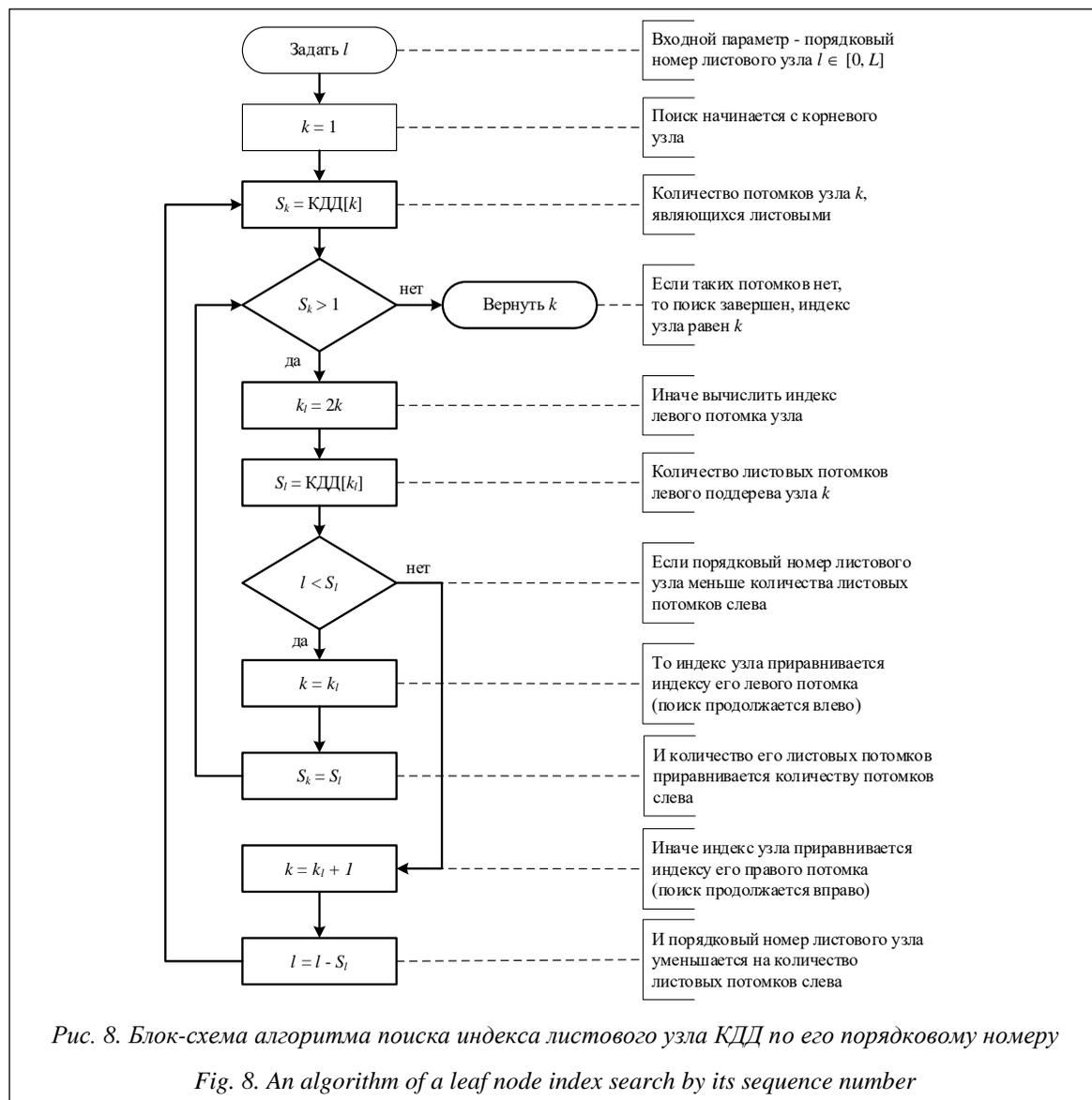


Рис. 8. Блок-схема алгоритма поиска индекса листового узла КДЦ по его порядковому номеру

Fig. 8. An algorithm of a leaf node index search by its sequence number

$$S_{12} = \sum_{d=0}^{n-1} 2^d.$$

Уменьшаемое  $S_{11}$  представляет собой геометрическую прогрессию со знаменателем  $q = 2$  и первым членом  $a_1 = 2^0 = 1$ . Частная сумма ряда  $S_{11}$  будет равна сумме  $D + 1$  членов этой геометрической прогрессии:

$$S_{11} = \frac{a_1(q^{D+1} - 1)}{q - 1} = \frac{1(2^{D+1} - 1)}{1 - 2} = 2^{D+1} - 1. \quad (15)$$

По аналогии с (15)

$$S_{12} = \frac{a_1(q^{n+1} - 1)}{q - 1} = \frac{1(2^n - 1)}{1 - 2} = 2^n - 1.$$

Тогда частная сумма ряда  $S_1$  равна

$$S_1 = S_{11} - S_{12}, \\ S_1 = 2^{D+1} - 2^n.$$

Частная сумма ряда  $S_2$  от  $n$  до  $D$  также равна разности между частной суммой этого ряда от нуля до  $D$  и частной суммой от нуля до  $n - 1$ :

$$S_2 = \sum_{d=0}^D d2^d - \sum_{d=0}^{n-1} d2^d. \quad (16)$$

Обозначим уменьшаемое в (16) как

$$S_{21} = \sum_{d=0}^D d2^d, \quad (17)$$

а вычитаемое как

$$S_{22} = \sum_{d=0}^{n-1} d2^d.$$

Найдем  $S_{21}$ . Для этого домножим обе части равенства (17) на 2, а множитель  $d$  запишем как  $d + 1 - 1$ :

$$2S_{21} = \sum_{d=0}^D (d + 1 - 1)2^{d+1}. \quad (18)$$

Представим (18) в виде разности сумм:

$$2S_{21} = \sum_{d=0}^D (d + 1)2^{d+1} - \sum_{d=0}^D 2^{d+1} \quad (19)$$

или

$$2S_{21} = S_{211} - S_{212}, \quad (20)$$

где

$$S_{211} = \sum_{d=0}^D (d + 1)2^{d+1},$$

$$S_{212} = \sum_{d=0}^D 2^{d+1}.$$

Запишем первую сумму ряда в (19):

$$S_{211} = 1(2^1) + 2(2^2) + 3(2^3) + \dots + D(2^D) + \\ + (D + 1)(2^{D+1}).$$

Все слагаемые в этом ряду, за исключением последнего, можно записать как

$$1(2^1) + 2(2^2) + 3(2^3) + \dots + D(2^D) = \sum_{d=0}^D d2^d.$$

Тогда с учетом (17)

$$S_{211} = S_{21} + (D + 1)2^{D+1}. \quad (21)$$

Вторая сумма ряда в (19)  $S_{212}$  равна сумме  $D + 1$  членов геометрической прогрессии со знаменателем  $q = 2$  и первым членом  $a_1 = 2^1 = 2$ :

$$S_{212} = \frac{a_1(q^{D+1} - 1)}{q - 1} = \frac{2(2^{D+1} - 1)}{1 - 2} = \\ = 2(2^{D+1}) - 2 = 2^{D+2} - 2. \quad (22)$$

Подставляя в (20) выражения (21) и (22), получим

$$2S_{21} = S_{21} + (D + 1)2^{D+1} - 2^{D+2} - 2.$$

Вычтем из обеих частей этого равенства  $S_{21}$ :

$$S_{21} = (D + 1)2^{D+1} - 2^{D+2} - 2.$$

В результате упрощения

$$S_{21} = D(2^{D+1}) - 2^{D+1} - 2. \quad (23)$$

По аналогии с (23)

$$S_{22} = n(2^n) - 2^{n+1} + 2.$$

Таким образом, частная сумма ряда

$$S_2 = S_{21} - S_{22},$$

$$S_2 = D(2^{D+1}) - 2^{D+1} - n(2^n) + 2^{n+1}.$$

Подставляя  $S_1$  и  $S_2$  в (13), получаем

$$N_b = D(2^{D+1} - 2^n) - \\ - (D2^{D+1} - 2^{D+1} - n2^n + 2^{n+1}) + \\ + (2^{D+1} - 2^n).$$

В результате упрощения

$$N_b = 2^{D+2} - D(2^n) + n(2^n) - 3(2^n). \quad (24)$$

Для случая  $n = 0$  выражение (24) принимает вид  $N_b = 2^{D+2} - D - 3$ .

На практике к этому количеству бит необходимо добавить  $D + 1$  бит для записи глубины дерева в элементе с индексом 0 в виде  $2^D$ . Еще два дополнительных бита необходимы для выравнивая их общего количества до границы восьмибитного байта. Таким образом, количество бит, необходимых для размещения КДД глубиной  $D$ , равно

$$N_b = 2^{D+2}.$$

На рисунке 9 представлен пример размещения КДД глубиной  $D = 4$  в непрерывном участке памяти.

Несмотря на то, что рост количества необходимой для размещения КДД памяти растет экспоненциально с ростом глубины дерева, ее объем остается относительно небольшим для средних глубин. Так, КДД глубиной  $D = 20$ , имеющее 1 048 576 листовых узлов на последнем уровне, для хранения требует всего 512 килобайт памяти.

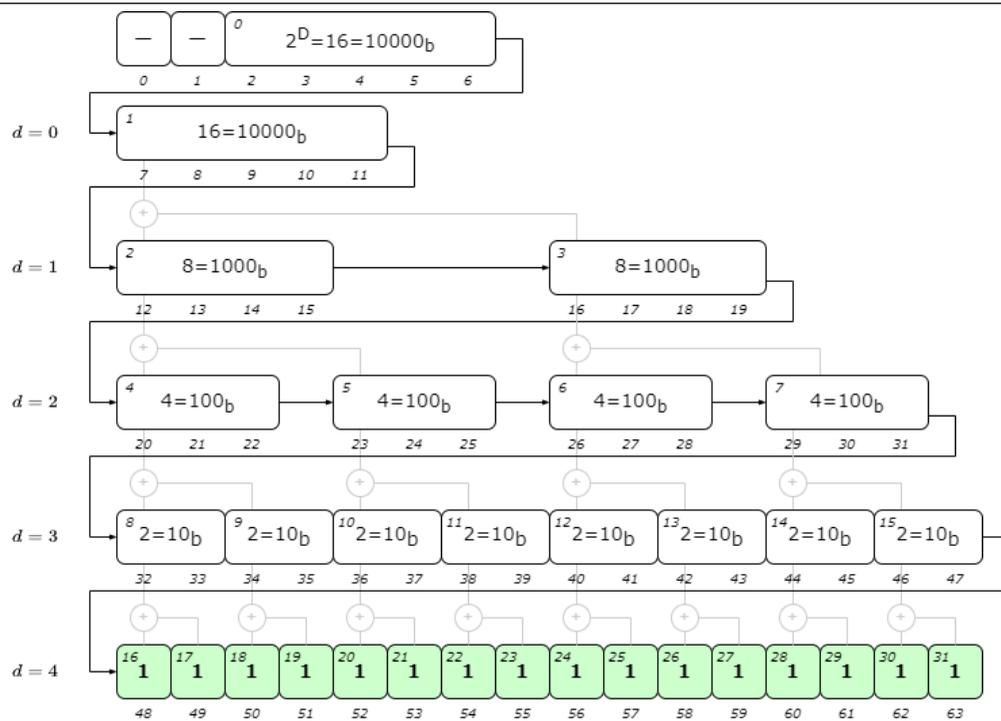


Рис. 9. Расположение структуры данных КДД глубины  $D = 4$  в непрерывном участке памяти (ширина каждого элемента пропорциональна количеству занимаемых бит)

Fig. 9. A concurrent binary tree with a depth  $D = 4$  memory layout (the width of each block is proportional to its bit-size)

Доступ к элементам структуры КДД. Для доступа к  $k$ -му элементу КДД необходимо определить диапазон бит, в которых он хранится в непрерывном представлении КДД в памяти.

Первые два бита в структуре КДД не используются, размер последнего элемента равен одному биту, а полный размер структуры составляет  $2^D$  бит. Следовательно, смещение  $k$ -го элемента от начала структуры КДД  $x_k \in [2, 2^D - 1]$ .

Из предыдущего текста известно, что  $k$ -й элемент занимает  $N_b = D - d_k - 1$  бит, где  $d_k = \lfloor \log_2(k) \rfloor$ .

Выведем выражение для определения индекса первого бита  $x_k$  для  $k$ -го элемента КДД.

Заметим, что размер всех элементов КДД в пределах любого уровня  $d$  одинаков и определяется по формуле (11). Тогда, зная смещение первого элемента уровня  $d$  относительно начала КДД и порядковый номер узла  $k$  на этом уровне, можно однозначно найти искомым  $x_k$ .

Определим индекс бита  $x_{d_k}$ , с которого начинается запись узлов КДД на уровне  $d_k$ . Для этого найдем разницу между полным размером КДД и количеством бит, необходимых для записи узлов уровней дерева, начиная с уровня  $d_k$ :

$$x_{d_k} = N_b - N_{b_{tail}}$$

По аналогии с (12) количество бит  $N_{b_{tail}}$  для кодирования узлов уровней от  $d_k$  до  $D$  равно

$$N_{b_{tail}} = \sum_{d=d_k}^D 2^d (D - d + 1).$$

Ранее эта сумма была найдена для общего случая. Решая (24) для случая  $n = d_k$ , получим

$$N_{b_{tail}} = 2^{D+2} - D2^{d_k} + d_k 2^{d_k} - 3(2^{d_k}).$$

Тогда смещение первого элемента уровня  $d_k$  будет

$$x_{d_k} = 2^{d_k} (D - d_k + 3). \tag{25}$$

Из (2) индекс первого узла на уровне  $d_k$  равен

$$k_d = 2^{d_k}. \tag{26}$$

Следовательно, выразим общее смещение первого бита узла  $k$ :

$$x_k = x_{d_k} + (k - k_d) N_{b_k}.$$

При этом каждый узел на уровне  $d_k$  занимает  $N_{b_k}$  бит. После подстановки (25), (26)

и (11) получаем выражение

$$x_k = 2^{d_k} (D - d_k + 3) + (k - 2^{d_k}) (D - d_k + 1).$$

В результате упрощения

$$x_k = 2^{d_k+1} + k(D - d_k + 1).$$

Таким образом,  $k$ -й элемент КДД занимает в его представлении в памяти биты в интервале  $[x_k, x_k + N_{b_k})$ . Например, первый бит битового поля КДД, которое кодирует его листовые узлы, имеет индекс  $k = 2^D$  и занимает в памяти диапазон бит  $[3(2^D), 3(2^D + 1))$ . Для КДД, изображенного на рисунке 9, это диапазон  $[48, 49)$ .

### Процесс использования КДД

После первичной инициализации КДД его использование представляет собой циклический процесс параллельного обновления топологии путем разбиения или слияния узлов с последующей редукцией суммы (рис. 10).

Для инициализации КДД заданной глубины  $D$  выделяется непрерывный участок памяти размером  $N_b = 2^D + 2$  бит. Выделенная память инициализируется нулями. Затем выполняется первичное разделение дерева до выбранной глубины  $d \in [0, D]$ . Для этого определяется диапазон индексов узлов этого уровня  $k_d \in [2^d, 2^{d+1} - 1]$  и для каждого из них в битовом поле устанавливается в единицу значение соответствующего ему бита с индексом  $x_k = k2^{D-d_k} - 2^D$ .

В результате в битовом поле установленным в единицу оказывается каждый  $2^{D-d}$  бит.

Результат инициализации КДД при  $d = D = 4$  приведен на рисунке 7 (слева).

Редукция суммы КДД выполняется начиная с битового поля на уровне  $d = D$  и продолжается до записи в корневой узел на уровне  $d = 0$  количества всех листовых узлов дерева. Алгоритм требует  $D - 1$  итераций. Начиная с  $d = D$  на каждой итерации выполняется  $2^{d-1}$  сложений. Общее количество сложений будет

$$\sum_{d=1}^D 2^{d-1} = 2^D - 1.$$

Следовательно, сложность алгоритма редукции двоичного дерева составляет  $O(2^D)$ .

Алгоритмы редукции двоичного дерева могут быть очень эффективно реализованы с использованием параллельных вычислений, в том числе с GPU [13, 14]. При вычислении редукции в  $P$  потоков сложность алгоритма составляет

$$O\left(\frac{2^D}{P} + \log_2 P\right).$$

Алгоритм обновления топологии КДД представляет собой простую итерацию по всем его листовым узлам, каждый из которых может быть объединен с соседним узлом (сестринским) или разделен на два дочерних. Условие слияния или разделения узлов зависит от конкретного практического применения КДД.

После завершения обновления топологии выполняется редукция суммы и процесс циклически повторяется.

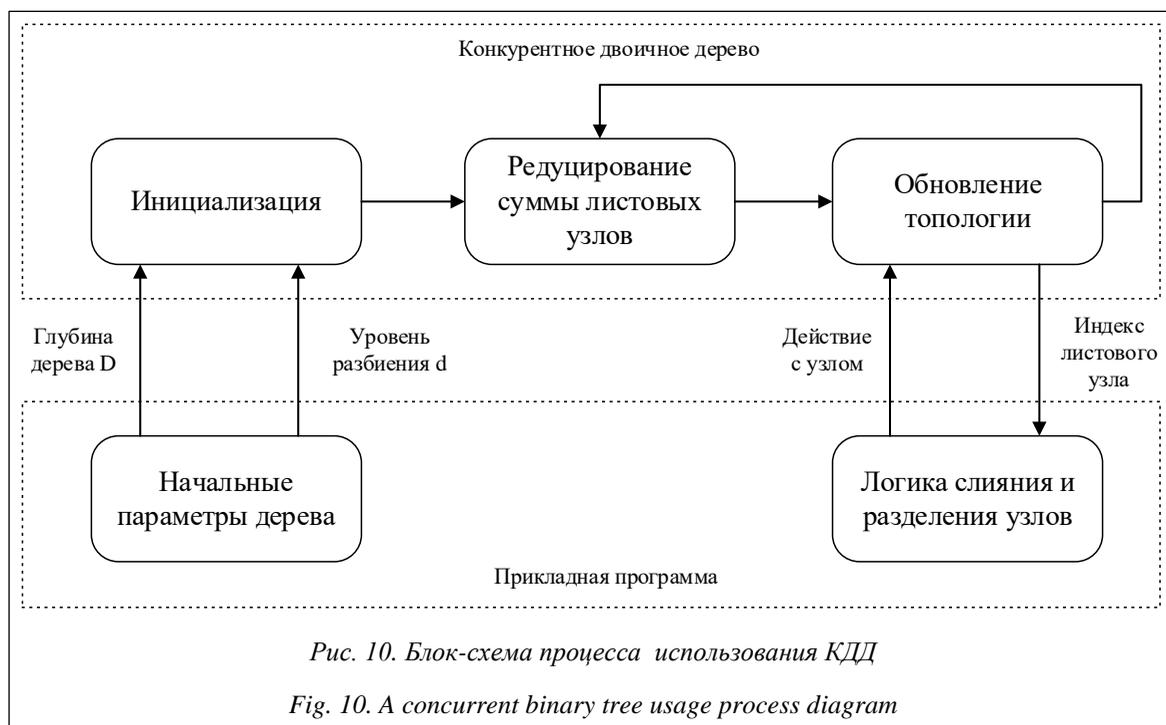


Рис. 10. Блок-схема процесса использования КДД

Fig. 10. A concurrent binary tree usage process diagram

### Тесты производительности КДД

Синтетические тесты производительности процесса использования структуры КДД проводились с использованием его многопоточной программной реализации. Для многопоточной обработки КДД на центральном процессоре использована библиотека OpenMP. Многопоточная обработка на GPU реализована с использованием вычислительных шейдеров OpenGL [11].

Для хранения в памяти двоичной кучи КДД используются 32-битные переменные, причем количество бит для хранения суммы листовых узлов на каждом уровне различно. При многопоточной записи в двоичную кучу возможны ситуации, когда различные потоки будут одновременно обращаться к одной и той же переменной. Это может привести к повреждению данных в двоичной куче и нарушению корректной работы алгоритмов. Для исключения подобной ситуации на CPU применяется специальный тип `std::atomic<uint_32t>`, на аппаратном уровне гарантирующий атомарность операций с ним [15]. На GPU используются специальные атомарные операции над типом

`uint – atomicAnd` и `atomicOr`. Запись нескольких битов в 32-битную переменную – единственное место в программной реализации КДД, требующее применения специальных приемов многопоточного программирования. Такая ситуация возникает при вычислении суммы листовых узлов дерева, а также при выполнении операций слияния и разделения узлов.

Сравнительные результаты замеров производительности редукции суммы листовых узлов КДД представлены по ссылке (<http://www.swsys.ru/uploaded/image/2023-3/2023-3-dop/13.jpg>), их декодирования – на рисунке 11.

Время редуцирования суммы листовых узлов КДД линейно уменьшается с увеличением количества потоков, выполняющих алгоритм редукции. Использование восьми потоков процессора уменьшает время выполнения алгоритма в пять–шесть раз. Редуцирование суммы с использованием нескольких тысяч шейдеров GPU снижает это время на два-три порядка.

Декодирование листовых узлов также эффективно выполняется несколькими потоками. Прирост скорости при выполнении итерации по листовым узлам на GPU составляет более двух тысяч раз.

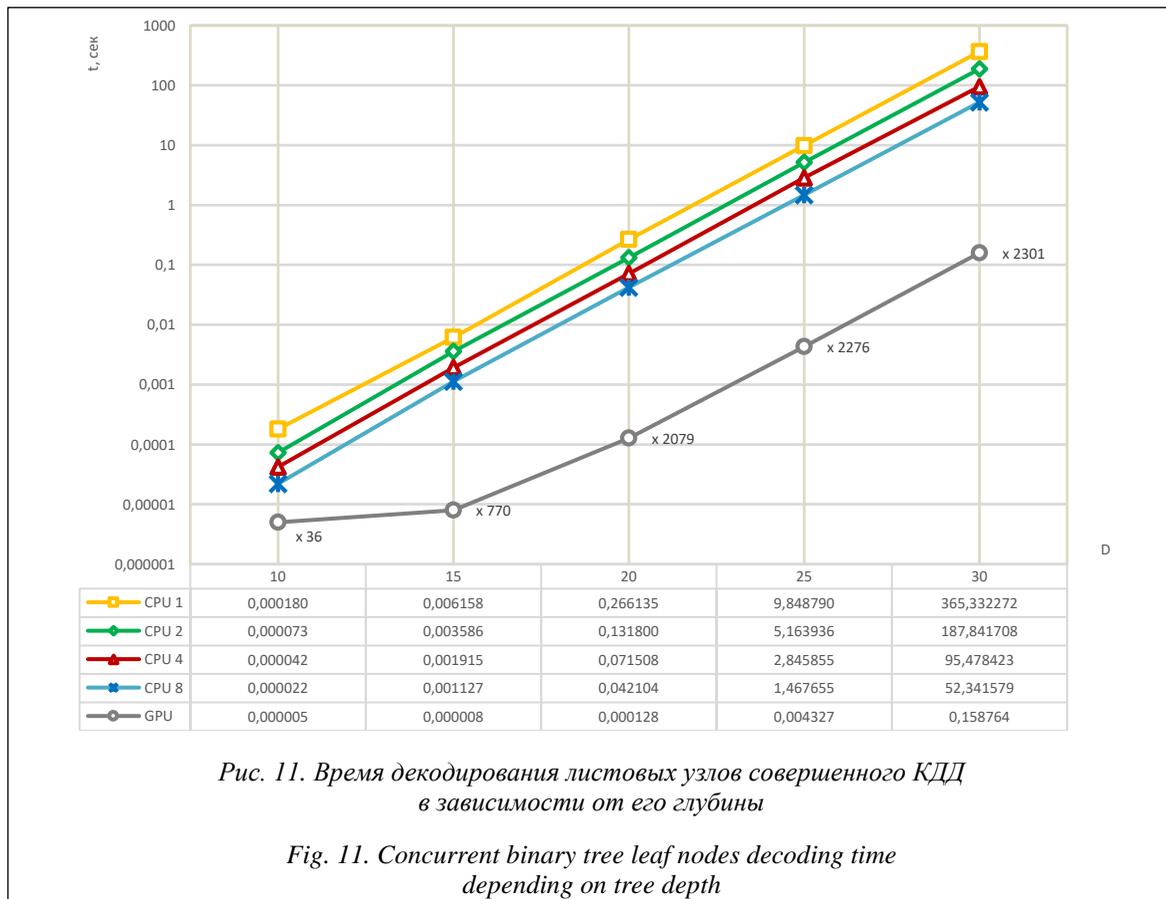


Рис. 11. Время декодирования листовых узлов совершенного КДД в зависимости от его глубины

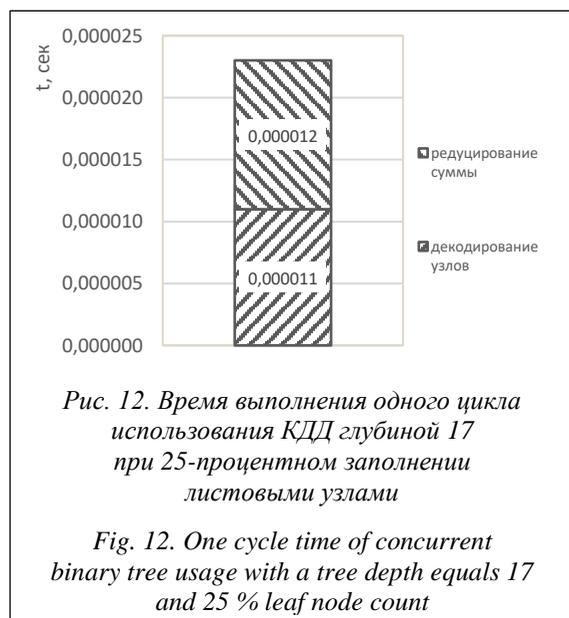
Fig. 11. Concurrent binary tree leaf nodes decoding time depending on tree depth

Необходимо обратить внимание, что на рисунке 11 представлены результаты декодирования максимально возможного количества листовых узлов совершенного КДД заданной глубины, то есть наиболее худший с точки зрения производительности случай. При практическом использовании КДД в СММО для реализации алгоритма рекурсивного деления количество его листовых узлов, как правило, не превышает 20 % от максимально возможного. Время полного перебора листовых узлов дерева пропорционально уменьшается в зависимости от степени его заполнения. Пример интегральной производительности КДД (вычисление редукции суммы плюс декодирование листовых узлов) с 25-процентным заполнением листовыми узлами приведен по ссылке (<http://www.swsys.ru/uploaded/image/2023-3/2023-3-dop/14.jpg>).

На практике нужно дополнительно учесть время, затрачиваемое прикладной программой на принятие решения о слиянии или разделении узлов дерева. Однако оно не повлияет на линейный характер масштабируемости производительности КДД.

Интегральная производительность одного цикла использования КДД на реальном примере использования в СММО отражена на рисунке 12.

В СММО глубина КДД составляет 17 при 25-процентном заполнении листовыми узлами. Общее время выполнения декодирования листовых узлов с последующим расчетом их редукции составляет 23 микросекунды. Если декодирование листовых узлов не



приводит к последующему изменению топологии дерева, то есть не происходит слияние или разделение узлов, это время уменьшается до 11 микросекунд за счет исключения из цикла шага редукции суммы. Такая ситуация в СММО встречается, когда априорно известно, что алгоритм рекурсивного деления не внесет изменений в топологию двоичного дерева, так как не изменились входные данные этого алгоритма.

### Заключение

Задача автоматического построения локально адаптивной трехмерной модели глобального рельефа может быть эффективно решена на GPU с использованием алгоритма рекурсивного деления, допускающего массивную параллельную обработку данных. Для обеспечения корректной параллельной работы такого алгоритма обрабатываемые данные необходимо представлять в виде специализированной структуры – КДД. Рассмотренные в статье теоретические свойства КДД и результаты тестов его производительности позволяют сделать следующие выводы.

КДД сочетает в себе возможность представления произвольной изменяемой топологии дерева и эффективной параллельной обработки его узлов.

Представление КДД в непрерывном участке памяти позволяет использовать его на GPU.

Несмотря на экспоненциальный характер роста потребного размера памяти КДД с увеличением его максимальной глубины ее объем остается в пределах, допускающих применение КДД в условиях ограниченного бюджета видеопамяти.

Параллельная итерация по листовым узлам КДД может быть совмещена с их слиянием и разделением без потери работоспособности структуры и без необходимости промежуточного хранения индексов листовых узлов с дополнительным выделением памяти.

Производительность алгоритмов обработки КДД на GPU позволяет эффективно использовать его в приложениях реального времени, таких как системы визуализации.

Применение КДД ограничено задачами, использующими алгоритмы рекурсивного деления, в которых возможно либо разделение узла на два потомка, либо слияние двух узлов в родительский узел и исключена ситуация возникновения у узла дерева единственного потомка.

## Список литературы

1. Барков В.А., Грачев В.Г., Насыров Р.Р. и др. Система моделирования морских операций (СММО): Свид. о регистр. ПрЭВМ № 201866191. Рос. Федерация, 2018.
2. Барков В.А., Грачев В.Г., Насыров Р.Р. и др. СММО – Моделирующий комплекс: Свид. о регистр. ПрЭВМ № 2018661128. Рос. Федерация, 2018.
3. Lauterbach C., Garland M., Sengupta S., Luebke D., Manocha D. Fast BVH construction on GPUs. *Comput. Graphics Forum*, 2009, vol. 28, no. 2, pp. 375–384. doi: 10.1111/j.1467-8659.2009.01377.x.
4. Garanzha K., Pantaleoni J., McAllister D. Simpler and faster HLBVH with work queues. *Proc. ACM SIGGRAPH Symposium on HPG*, 2011, pp. 59–64. doi: 10.1145/2018323.2018333.
5. Karras T. Maximizing parallelism in the construction of BVHs, octrees, and k-d trees. *Proc. ACM SIGGRAPH Symposium on HPG*, 2012, pp. 33–37.
6. Воронцов Г.В., Преображенский А.П., Чопоров О.Н. Быстрое построение BVY дерева на GPGPU // Моделирование, оптимизация и информационные технологии. 2018. Т. 6. № 2. С. 25–34.
7. Dupuy J. Concurrent binary trees (with application to longest edge bisection). *PACMCGIT*, 2020, vol. 3, no. 2, pp. 1–20. doi: 10.1145/3406186.
8. Moir M., Shavit N. *Concurrent Data Structures*. Boca Raton, CRC Press, 2018, 31 p.
9. Munro J.I., Suwanda H. Implicit data structures for fast search and update. *JCSS*, 1980, vol. 21, no. 2, pp. 236–250.
10. Кормен Т.Х., Лейзерсон Ч.И., Ривест Р.Л., Штайн К. Алгоритмы: построение и анализ; [пер. с англ.]. М.: Вильямс, 2013. 1328 с.
11. Kessenich J., Baldwin D., Rost R. The OpenGL Shading Language, Version 4.60.7. 234 p. URL: <https://registry.khronos.org/OpenGL/specs/gl/GLSLangSpec.4.60.pdf> (дата обращения: 03.05.2023).
12. Drepper U. What Every Programmer Should Know About Memory. 2007, 114 p. URL: <https://www.akkadia.org/drepper/cpumemory.pdf> (дата обращения: 03.05.2023).
13. Harris M. Optimizing Parallel Reduction in CUDA. *NVIDIA Developer Technology*. 38 p. URL: <https://developer.download.nvidia.com/assets/cuda/files/reduction.pdf> (дата обращения: 03.05.2023).
14. Mahardito A., Suhendra A., Hasta D.T. Optimizing Parallel Reduction in CUDA to Reach GPU Peak Performance. URL: <https://core.ac.uk/reader/143963699> (дата обращения: 03.05.2023).
15. Уильямс Э. Параллельное программирование на C++ в действии. Практика разработки многопоточных программ; [пер. с англ.]. М.: ДМК Пресс, 2012. 672 с.

Software &amp; Systems

doi: 10.15827/0236-235X.142.398-413

2023, vol. 36, no. 3, pp. 398–413

## A research on concurrent binary tree properties

Vsevolod G. Grachev  
Aleksandr S. Ushakov

## For citation

Grachev, V.G., Ushakov, A.S. (2023) ‘A research on concurrent binary tree properties’, *Software & Systems*, 36(3), pp. 398–413 (in Russ.). doi: 10.15827/0236-235X.142.398-413

## Article info

Received: 10.05.2023

After revision: 05.06.2023

Accepted: 30.06.2023

**Abstract.** The navigation simulator software “Marine Operation Simulation System” (MOSS) implements the automatic construction technology for a global terrain 3D-model based on the recursive subdivision algorithm with a locally adaptive level of detail control. The algorithm is entirely implemented on the graphics adapter (GPU) and uses massive parallel data processing in compute shaders. The recursive subdivision algorithm is based on using a binary tree; however, classical binary trees explicitly described using pointers are not applicable to GPU implementations due to architectural features. To ensure the parallel execution of the algorithm, the authors use a specialized parallel data structure that is a concurrent binary tree (CBT). The article describes the prerequisites for creating a CBT structure and considers its construction stages from using an implicit binary tree up to representing a binary tree as a binary field augmented with a reduced sum of leaf nodes. The required RAM amount to accommodate a CBT of a given depth is theoretically substantiated. There is an analysis of the algorithmic complexity of constructing a CBT and iterating over its leaf nodes. The article presents and analyzes the results of comparative synthetic performance tests of a concurrent binary tree performed on a central processing unit (CPU) and GPU, as well as the results of the practical application of CBT in MOSS.

**Keywords:** concurrent binary tree, parallel data structure, recursive subdivision algorithm, memory optimization, massive GPU parallelism, marine operations simulation system

## Reference List

1. Barkov, V.A., Grachev, V.G., Nasyrov, R.R. et al. (2018) *Marine Operations Simulation System (MOSS)*, Pat. RF, № 2018661914.
2. Barkov, V.A., Grachev, V.G., Nasyrov, R.R. et al. (2018) *MOSS – Modeling Complex*, Pat. RF, № 2018661128.
3. Lauterbach, C., Garland, M., Sengupta, S., Luebke, D., Manocha, D. (2009) 'Fast BVH construction on GPUs', *Comput. Graphics Forum*, 28(2), pp. 375–384. doi: 10.1111/j.1467-8659.2009.01377.x.
4. Garanzha, K., Pantaleoni, J., McAllister, D. (2011) 'Simpler and faster HLBVH with work queues', *Proc. ACM SIGGRAPH Symposium on HPG*, pp. 59–64. doi: 10.1145/2018323.2018333.
5. Karras, T. (2012) 'Maximizing parallelism in the construction of BVHs, octrees, and k-d trees', *Proc. ACM SIGGRAPH Symposium on HPG*, pp. 33–37.
6. Vorontsov, G.V., Preobrazhenskiy, A.P., Choporov, O.N. (2018) 'The algorithms of parallel radix sorting on GPGPU', *Modeling, Optimization and Inform. Tech.*, 6(2), pp. 25–34 (in Russ.).
7. Dupuy, J. (2020) 'Concurrent binary trees (with application to longest edge bisection)', *PACMCGIT*, 3(2), pp. 1–20. doi: 10.1145/3406186.
8. Moir, M., Shavit, N. (2018) *Concurrent Data Structures*. Boca Raton: CRC Press, 31 p.
9. Munro, J.I., Suwanda, H. (1980) 'Implicit data structures for fast search and update', *JCSS*, 21(2), pp. 236–250.
10. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C. (2013) *Introduction to Algorithms*. Cambridge: The MIT Press, 1292 p. (Russ. ed.: (2013) Moscow, 1328 p.).
11. Kessenich, J., Baldwin, D., Rost, R. *The OpenGL Shading Language, Version 4.60.7*. 234 p., available at: <https://registry.khronos.org/OpenGL/specs/gl/GLSLangSpec.4.60.pdf> (accessed May 03, 2023).
12. Drepper, U. (2007) *What Every Programmer Should Know About Memory*, 114 p., available at: <https://www.akkadia.org/drepper/cpumemory.pdf> (accessed May 03, 2023).
13. Harris, M. *Optimizing Parallel Reduction in CUDA. NVIDIA Developer Technology*, 38 p., available at: <https://developer.download.nvidia.com/assets/cuda/files/reduction.pdf> (accessed May 03, 2023).
14. Mahardito, A. *Optimizing Parallel Reduction in CUDA to Reach GPU Peak Performance*, available at: <https://core.ac.uk/reader/143963699> (accessed May 03, 2023).
15. Williams, A. (2012) *C++ Concurrency in Action: Practical Multithreading*, Shelter Island, NY: Manning Publ., 506 p. (Russ. ed.: (2012) Moscow, 672 p.).

## Авторы

**Грачев Всеволод Геннадиевич**<sup>1</sup>, к.т.н.,  
ведущий научный сотрудник отдела комплексных  
технических средств обучения, [grachev@cps.tver.ru](mailto:grachev@cps.tver.ru)  
**Ушаков Александр Сергеевич**<sup>2</sup>, к.ф.-м.н.,  
начальник отдела программно-аппаратных средств  
и телекоммуникационных систем,  
[al.s.usakov@yandex.ru](mailto:al.s.usakov@yandex.ru)

<sup>1</sup> НИИ «Центрпрограммсистем»,  
г. Тверь, 170024, Россия

<sup>2</sup> ОАО «НИИЭС»,  
г. Москва, 125124,  
Россия

## Authors

**Vsevolod G. Grachev**<sup>1</sup>, Ph.D. (Engineering),  
Leading Researcher,  
[grachev@cps.tver.ru](mailto:grachev@cps.tver.ru)  
**Aleksandr S. Ushakov**<sup>2</sup>,  
Ph.D. (Physics and Mathematics),  
Head of Department,  
[al.s.usakov@yandex.ru](mailto:al.s.usakov@yandex.ru)

<sup>1</sup> R&D Institute Centerprogramsistem,  
Tver, 170024, Russian Federation

<sup>2</sup> Open Joint Stock Company  
"Scientific Testing Institute of Ergatic Systems",  
Moscow, 125124, Russian Federation

## Метод деформации функционально заданных объектов с применением графических процессоров

С.И. Вяткин  
Б.С. Долговесов

### Ссылка для цитирования

Вяткин С.И., Долговесов Б.С. Метод деформации функционально заданных объектов с применением графических процессоров // Программные продукты и системы. 2023. Т. 36. № 3. С. 414–422. doi: 10.15827/0236-235X.142.414-422

### Информация о статье

Поступила в редакцию: 02.03.2023

После доработки: 10.04.2023

Принята к публикации: 12.04.2023

**Аннотация.** В статье рассмотрен метод моделирования деформации эластичных функционально заданных объектов. Деформация и анимация трехмерных объектов являются важными функциями, но требуют большого числа вычислений. Моделирование эластичных динамических объектов применяется в приложениях компьютерной графики, биомеханики и робототехники. Целью является разработка метода моделирования деформации эластичных объектов на графическом процессоре. Предметом исследования стали методы Чебышева и Якоби и их использование в проективной динамике. Теоретическая значимость разработки заключается в комбинировании данных методов для решения задач деформации и анимации поверхностей, где проблемой является большое число вычислений. Проективная динамика может быть эффективно ускорена с применением метода Чебышева. Практическая значимость состоит в реализации метода на графическом процессоре. Комбинация метода Чебышева и метода Якоби для проективной динамики оптимально реализуется на графическом процессоре, поскольку вычисления хорошо распараллеливаются. В результате получаем простую реализацию, которая не нуждается в дополнительных библиотеках для решения этой проблемы и имеет небольшую память. Метод не требует повторной факторизации матрицы при каждом изменении системы, используемой при прямом вычислении, но необходимы большие вычислительные затраты. При тестировании предлагаемого метода использованы патчи свободных форм на основе сеток базовых треугольников. Тестирование показывает, что такой подход ускоряет проективную динамику примерно на один порядок, когда на глобальном этапе используется решение Якоби.

**Ключевые слова:** анимация, деформация, патч свободной формы, метод Якоби, метод Чебышева, проективная динамика, параллельные вычисления, графический процессор

Численные методы для дифференциальных уравнений являются введением в фундаментальную область численного анализа и научных вычислений. Моделирование деформации объекта составляет фундаментальную задачу анимации. Для этого используются как явные, так и неявные модели пошагового времени. Применяются также методы оптимизации для эффективного динамического моделирования.

Анимация, основанная на физике, является важным инструментом в компьютерной графике, но она требует больших вычислений. Моделирование в реальном времени необходимо в таких приложениях, как компьютерные игры и обучающие тренажеры (например, хирургические симуляторы).

Деформация мягких тканей в реальном времени имеет большое значение для интерактивного хирургического моделирования. С этой целью разрабатываются алгоритмы моделирования деформации мягких тканей в реальном времени при силовом взаимодействии, которые имитируют механические деформации тканей. Подходы адаптированы как для изотропных, так и для анизотропных и гетерогенных материалов.

Эти алгоритмы позволяют упрощать свойства тканевого материала, вязкоупругость тканей и их анатомию. Такие методы описывают принципы и компоненты хирургического тренажера, показывают процесс создания симулятора хирургии для конкретного пациента из набора медицинских изображений.

В биомеханике решаются задачи оптимизации траектории для сочлененных тел. Используются дифференцируемые динамические модели, аппроксимации внутренних и внешних сил в виде плавной функции кинематического состояния сочлененного тела. Анизотропные материалы применяются в биомеханике, к ним относятся биологические ткани, такие как кожа и мышцы.

Часто используемыми методами моделирования деформируемых объектов являются позиционная и проективная динамика [1, 2].

В статье [3] предлагается метод, основанный на физике, в режиме реального времени с динамическими деталями. Описано многослойное представление поверхностей, в результате имитируются правдоподобная динамика поверхности и взаимодействие поверхностей.

В основе метода, описанного в [4], лежит разложение деформации на высокочастотную

и низкочастотную составляющие. Используются ядра с узкой полосой пропускания, чтобы гарантировать объединение только прогнозов с аналогичными высокочастотными шаблонами. Веса смеси вычисляются с использованием ядра с узкой полосой пропускания, чтобы гарантировать объединение только прогнозов с аналогичными высокочастотными шаблонами. Различные вариации получаются путем вычисления подпространства деформации, которое удовлетворяет физическим ограничениям. Метод полностью дифференцируем, поэтому может использоваться в области компьютерного зрения и алгоритмов обучения.

В работе [5] предложен метод, основанный на деформируемых графических блоках и управлении их положением и поведением с помощью подхода, основанного на ограничениях. В методе используются неявные деформации. При восстановлении мягких объектов в различных деформационных состояниях результаты имеют неполную геометрию и текстуру из-за самозатенения. Для решения этой проблемы обычно используется априорный шаблон, но он требует сложного сканирования и сложной настройки.

В статье [6] речь идет о структуре для реконструкции деформируемого мягкого объекта с полной геометрией и согласованной текстурой путем введения самогенерируемого шаблона с постепенным завершением. Также предлагается согласованное марковское случайное поле для ограничения моделей сетки в разных состояниях для создания согласованной текстуры и направления нежесткой деформации.

Подход, изложенный в [7], автоматически точно передает детали от источника к цели, используя двухэтапный процесс. На этапе установления семантического соответствия строится плотное соответствие между источником и целью. Используется их семантическая информация в двумерных шаблонах. Сначала вычисляются начальные точки соответствия на парных двумерных шаблонах, используя их семантическую информацию. Эти маркерные точки действуют как ограничения и сопоставляются с соответствующими трехмерными моделями. Затем с помощью нежесткого итеративного алгоритма ближайшей точки устанавливается их соответствие для каждого треугольника. На этапе переноса деформации от источника к цели решается задача оптимизации.

В работе [8] предлагается многосеточная схема для моделирования деформируемых

объектов с высоким разрешением с интерактивной частотой кадров. Описывается специальный критерий проекции, основанный на сканировании пространственных координат с кусочно-постоянными весами, чтобы многосеточный метод Галеркина масштабировался для сеток с высоким разрешением. Использование пространственных координат сканирования позволяет снижать разрешение сеток, а кусочно-постоянные веса гарантируют достаточно разреженные линейные решения. С помощью проекционных матриц можно эффективно управлять многоуровневыми линейными системами. Метод может быть применен к различным схемам оптимизации, таким как метод Ньютона и проективная динамика. В отличие от позиционной динамики проективная точно решает неявное интегрирование по времени динамической системы, сформулированное в вариационной форме.

Метод интерактивного моделирования нелинейных деформируемых объектов предложен в статье [9]. Его особенностью является интеграция моделирования деформаций и отбраковки столкновений. Применяется объемное приближение сложных трехмерных форм. Используется проективная динамика, которая хорошо отражает высокочастотные локальные деформации моделей с высоким разрешением. Нелинейные деформируемые эффекты обеспечиваются проекцией локальных ограничений.

Работа [10] посвящена методу нелинейного деформируемого моделирования с быстрым непрерывным обнаружением столкновений и с надежным разрешением столкновений. В методе применяется преобразование медиальной оси для построения кинематического подпространства. Вместо проективной динамики используется классическая гиперупругость для реальных нелинейных материалов.

В [11] представлен способ использования глубоких нейронных сетей для улучшения моделирования на основе физики. Интегрируется классическая механика Ла-Гранжа с автокодировщиком для ускорения упругого моделирования деформируемых тел. Используется пошаговая конечная разность в сочетании с обратным автоматическим различием. Эта стратегия позволяет пользоваться удобством и точностью конечной разности сложных шагов и экономить избыточные сетевые проходы.

Обработка инъективной деформации описана в работе [12]. Метод решает сложные задачи обработки геометрии и анимации.

В настоящей статье предлагается использовать методы Чебышева и Якоби в проективной динамике.

Метод моделирования, основанный на ограничениях и известный как проективная динамика, становится менее эффективным, когда требуется больше итераций для уменьшения ошибок и артефактов быстро деформирующегося объекта.

В отличие от метода сопряженного градиента метод Чебышева имеет простую рекуррентную форму и не использует внутреннее произведение. Однако существует проблема: при использовании метода Чебышева необходимо знать диапазон собственных значений, что на практике получить трудно.

Если этот диапазон занижен или завышен, метод может медленно сходиться или даже расходиться. Эта проблема становится еще более сложной в силу того, что проективная динамика нелинейная и не существует матрицы для анализа собственных значений.

В данной работе показано применение подхода Чебышева для ускорения проективной динамики, поскольку этот метод прост в реализации и совместим с ускорением графического процессора. При этом комбинация метода Якоби и подхода Чебышева позволяет получить дополнительные преимущества.

Предлагается аппроксимация объектов трехмерных сцен патчами свободных форм на основе аналитических функций возмущения относительно базовых треугольников. Патчи свободных форм обладают достоинством сплайнового представления поверхностей – высокой степенью гладкости, но их главной отличительной чертой является сложность формы, что позволяет достичь компактного описания объектов высокого качества изображения. В отличие от известных патчей – бикубических, Безье, NURBS и т.д., которые ограничены порядком, предлагаемые патчи могут быть свободной формы, удовлетворять требованиям связности, гладкости и компактности.

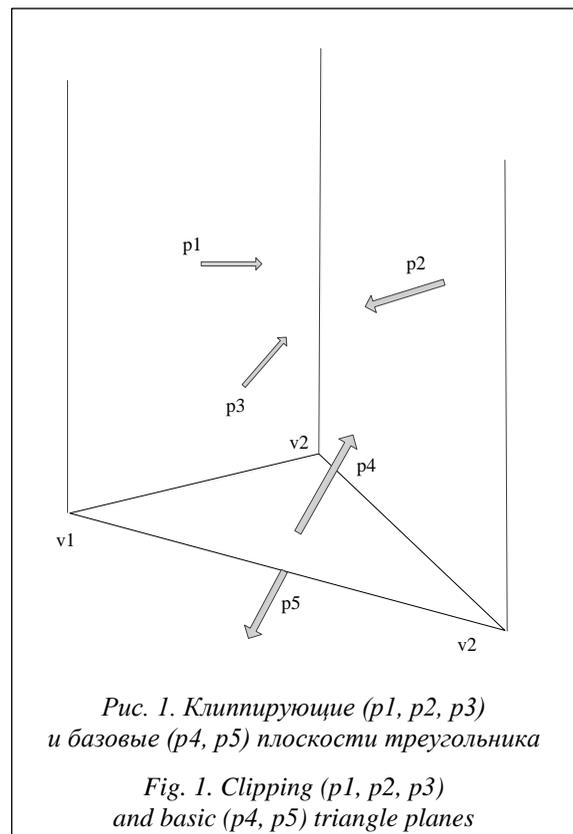
В данной работе представлен метод деформации трехмерных объектов на основе патчей свободных форм на графическом процессоре. Моделируются сетки, состоящие из патчей свободных форм. Определяется упругая энергия патча свободной формы. Для моделирования сеток на основе патчей свободных форм используется полярное разложение. Это необходимо для извлечения вращательной составляющей из градиента деформации с тензором растяжения. Вычисляются основные инварианты

для получения симметричной матрицы. Время вычислений на графическом процессоре зависит от количества ограничений и вершин, участвующих в каждом ограничении, а не от общего количества вершин. Для каждого ограничения патча свободной формы требуются четыре вершины.

Таким образом, представлен эффективный метод ускорения проективной динамики, основанный на методе Чебышева и решении Якоби. Метод требует небольших затрат памяти, может обрабатывать большие временные шаги и деформации и хорошо совместим с ускорением графического процессора.

### Описание метода

**Патч свободной формы** описывается с помощью задания функции отклонения от базового треугольника (рис. 1, 2) [13].



Треугольник ( $v_1, v_2, v_3$ ) в функциональном пространстве образуется пересечением пяти плоскостей: трех клипующих ( $p_1, p_2, p_3$ ), которые перпендикулярны базовым плоскостям треугольника с нормальными, ориентированными внутрь треугольника, и двух базовых ( $p_4, p_5$ ) (рис. 1) с противоположно направленными нормальными.

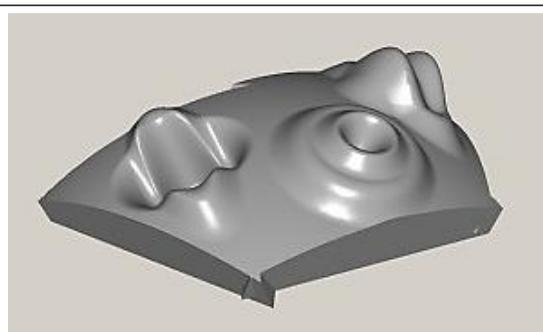


Рис. 2. Три патча свободной формы  
Fig. 2. Three free form patches

Патч свободной формы – это композиция базового треугольника и возмущений

$$F'(x, y, z) = F(x, y, z) + \sum_{i=1}^N R_i(x, y, z),$$

где функция возмущения  $R(x, y, z)$  находится следующим образом:

$$R_i(x, y, z) = \begin{cases} Q_i^3(x, y, z), & \text{if } Q_i(x, y, z) \geq 0, \\ 0, & \text{if } Q_i(x, y, z) < 0, \end{cases}$$

где  $Q(x, y, z)$  – возмущающая квадратика.

**Метод Якоби** – один из наиболее простых методов приведения системы матрицы к виду, удобному для итерации [14]. Из первого уравнения матрицы выражается неизвестное  $x_1$ , из второго –  $x_2$  и т.д. Результатом служит матрица, в которой на главной диагонали находятся нулевые элементы, а все остальные вычисляются по формуле. Для решения систем линейных алгебраических уравнений большой размерности, а также систем, имеющих разреженные матрицы, применение точных методов (например, метода Гаусса) нецелесообразно, так как сказывается ограниченность разрядной сетки и накапливается погрешность округления. Есть разные примеры, в которых необходимо решать системы линейных алгебраических уравнений с разреженной матрицей.

Для построения итеративной процедуры метода Якоби необходимо предварительно преобразовать систему уравнений  $A\bar{x} = \bar{b}$  к итерационному виду:  $\bar{x} = B\bar{x} + \bar{g}$ .

Процедуру нахождения решения представим как  $\bar{x}^{n+1} = B\bar{x}^n + \bar{g}$ , в виде поэлементной формулы

$$\bar{x}_k^{n+1} = \frac{1}{a_{kk}} \left( b_k - \sum_{l \neq k} a_{kl} x_l^n \right), \quad (1)$$

где  $n$  – счетчик итерации.

На каждой итерации хранятся старый и новый векторы приближений.

Для остановки и проверки точности, а также завершения вычислений на основе вычисления расстояния между соседними элементами в последовательности итераций можно воспользоваться евклидовой метрикой.

В этом случае условие завершения вычислений будет следующим:

$$\frac{\sqrt{\sum_{i=1}^n (x_k^{n+1} - x_k^n)^2}}{\sqrt{\sum_{i=1}^n (x_k^{n+1})^2}} < \varepsilon.$$

По сути это относительная погрешность, учитывающая все решения, а именно среднеквадратичные отклонения, чтобы исключить влияние знака: при сложении абсолютных разниц создавались бы иллюзии достижения необходимой точности и маленького разброса. Если бы одна разница была большой со знаком плюс, а другая примерно такой же большой, но со знаком минус, они просто аннигилировали бы друг друга и дали ложный результат. Именно поэтому используются квадраты отклонений, а потом уже накладывается корень. Чтобы считать евклидову метрику на каждой итерации, необходимо сохранять вектор решения на предыдущей итерации. То есть, считая новый набор корней на новой итерации, необходимо сохранять старый набор корней на предыдущей итерации, чтобы потом было с чем сравнивать.

На практике хорошо ограничивать выполнение цикла каким-либо предельным количеством итераций, например, от 100 до 500. Для хорошей сходимости этого достаточно, поскольку при схождении метода система достигает нужной точности примерно за 20–50 итераций (чаще всего при точности  $\varepsilon = 0.0001$ ). Если решение расходится, то вводится ограничение по числу итераций, чтобы не войти в бесконечный цикл.

Следует отметить, что скорость сходимости итерационного процесса выше для матриц, у которых элементы главной диагонали велики по сравнению с внедиагональными элементами. В связи с этим перед началом численного решения задачи желательно преобразовать систему уравнений так, чтобы преобладали диагональные элементы.

Порядок решения систем линейных алгебраических уравнений методом Якоби следующий:

- приведение системы уравнений к виду, в котором на каждой строчке выражено какое-либо неизвестное значение системы;

- произвольный выбор нулевого решения, в качестве которого можно взять вектор-столбец свободных членов;

- подстановка произвольного нулевого решения в систему уравнений, полученную вначале;

- выполнение дополнительных итераций, для каждой из которых используется решение, полученное на предыдущем этапе.

**Метод Чебышева.** Оптимальную последовательность можно вычислить из векторной последовательности, полученной в процессе разделения посредством общих линейных комбинаций [15]:

$$y_k = \sum_{j=0}^k a_{kj} x_j, \tag{2}$$

$$\sum_{j=0}^k a_{kj} = 1x_k - \hat{x} = M^k(x_0 - \hat{x}),$$

$$M = I - B^{-1}A,$$

$$y_k - \hat{x} = \sum_{j=0}^k a_{kj}(x_j - \hat{x}) = \sum_{j=0}^k a_{kj} M^j(x_0 - \hat{x}) =$$

$$= p_k(M)(x_0 - \hat{x}).$$

Начальная ошибка умножается на матричный полином с целью ее уменьшения. Если матрица имеет только действительные собственные значения в интервале, многочлен с наименьшей оценкой спектрального радиуса является сдвинутым многочленом Чебышева.

Для снижения вычислений и памяти вместо вычисления  $y_k$  (2) используется уравнение

$$p_k(x) = \frac{C_k(x/r)}{C_k(1/r)},$$

где  $C_k(x)$  – многочлен Чебышева;  $r$  – спектральный радиус.

Функцию обновления запишем в следующем виде:

$$y_{k+1} = \omega_{k+1}(B^{-1}(Cy_k + \vec{b}) - y_{k-1}) + y_{k-1}, \tag{3}$$

где  $\omega_{k+1} = \frac{4}{4 - r^2 \omega_k}$ ,  $\vec{b}$  – вектор, заданный в линейной системе.

Тогда часть функции обновления Чебышева является одним итерационным шагом решения Якоби, описанным в уравнении (1). Метод похож на взвешенный метод Якоби, но он сходится гораздо быстрее, потому что метод Чебышева изменяет коэффициент на каждой итерации.

Преимуществом метода Чебышева является его простота. Метод имеет короткую рекуррентную форму и не использует внутренние произведения, поэтому он подходит для параллельных вычислений на графическом процессоре. Однако метод Чебышева требует одного

спектрального радиуса, который оценивается численно. Если известен узкий диапазон собственных значений, можно настроить метод для более быстрой сходимости.

В проективной динамике важным является решение глобального шага. Если каждый временной шаг содержит небольшое количество итераций, можно решать задачу проективной динамики в реальном времени. Однако нескольких итераций недостаточно для локального шага, когда объект подвергается большой деформации.

Рассмотрим трехмерную динамическую систему с  $N$  вершинами. Движение от  $t$ -го момента времени до момента времени  $t + 1$  путем неявного интегрирования по времени будет:

$$\vec{p}_{t+1} = \vec{p}_t + s\vec{v}_{t+1},$$

$$\vec{v}_{t+1} = \vec{v}_t + sM^{-1}\vec{f}_{t+1},$$

где  $\vec{p}$  – вектор положения;  $\vec{v}$  – вектор скорости;  $M$  – матрица массы;  $s$  – временной шаг;  $\vec{f}$  – общая сила [16].

Если  $\vec{f}_{t+1} = \vec{f}_i(\vec{p}_{t+1}) + \vec{f}_e$ , внутренняя сила упругости  $\vec{f}_i$  является функцией  $\vec{p}$  и внешняя сила  $\vec{f}_e$  постоянная, то получим нелинейную систему:

$$M(\vec{p}_{t+1} - \vec{p}_t - s\vec{v}_t) = s^2(f_i(\vec{p}_{t+1}) + \vec{f}_e).$$

Используем одну итерацию Якоби на каждом глобальном шаге проективной динамики. В результате каждая итерация проективной динамики будет содержать локальный шаг и итерацию Якоби. Результат локального шага стабилизируется в течение нескольких итераций, поэтому линейная система на глобальном шаге после этого не меняется. Сходимость проективной динамики аналогична сходимости метода Якоби, решающего линейную систему за один глобальный шаг.

Предлагается использовать подход Чебышева для проективной динамики, который заменяет одну итерацию Якоби с помощью одной итерации проективной динамики. На графическом процессоре комбинация метода Якоби и Чебышева эффективна, поскольку она естественно совместима с параллельными вычислениями.

Рассмотрим  $r$  (спектральный радиус) как константу для каждой задачи моделирования. Вычислим  $r$  с помощью предварительного моделирования в два этапа. Пусть  $K$  – общее число итераций проективной динамики.

Инициализируем спектральный радиус  $r$ :

$$\frac{\|e^k\|_2}{\|e^{k-1}\|_2}.$$

Определяем ошибку:

$$e^k = \nabla\varphi(\bar{p}^k).$$

Также метод мощности оценивает спектральный радиус  $r$  для стандартного итерационного метода. Радиус является постоянным на протяжении всего процесса моделирования и зависит от общего числа итераций  $K$ .

Используем ограничения для моделирования сеток из патчей свободных форм и определим упругую энергию сетки:

$$E = stv\|G - R\|^2,$$

где  $st$  – жесткость;  $v$  – объем до деформации;  $G$  – градиент деформации;  $R$  – вращательная составляющая  $G$ .

Для моделирования сеток из патчей свободных форм используем полярное разложение, чтобы извлечь вращательную составляющую из градиента деформации  $G$ :  $G = GT$ .

Вычисляем три основных инварианта тензора растяжения  $T$  из градиента деформации  $G^T G$ , а затем используем их для получения тензора растяжения  $T$ .

На рисунке 3 показан алгоритм вычислений. Начальные условия функции обновления (2) следующие:

$$A: \omega_{k+1} = 1,$$

$$B: \omega_{k+1} = \frac{2}{2 - r^2},$$

$$C: \omega_{k+1} = \frac{4}{4 - r^2 \omega_k},$$

$$\omega_1 = 1, \omega_2 = \frac{2}{2 - r^2}.$$

Таким образом, осуществляется неявное интегрирование по времени динамической системы в вариационной форме. При этом получаются лучшие физические показатели. Сходящийся результат совпадает с точным решением динамической системы при неявном интегрировании Эйлера, которое не зависит от тесселяции сетки и количества итераций. Линейная система на глобальном шаге имеет постоянную матрицу. Критерий сходимости не ставит под угрозу надежность проективной динамики для сеток из патчей свободных форм даже в случае, когда сетка низкого качества или находится под большим напряжением. В итоге можно использовать ограничение деформации, чтобы избежать расхождения в этих случаях.

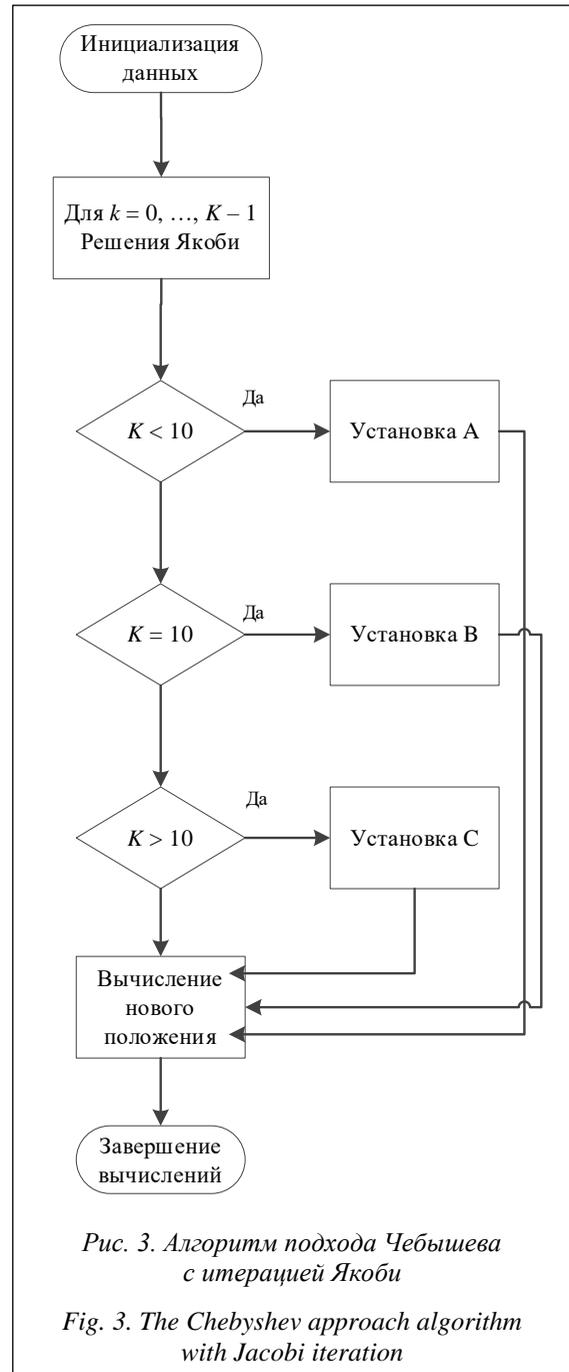


Рис. 3. Алгоритм подхода Чебышева с итерацией Якоби

Fig. 3. The Chebyshev approach algorithm with Jacobi iteration

### Результаты

Производительность метода протестирована на графическом процессоре. Результаты тестов приведены в таблице и проиллюстрированы соответствующими рисунками (см. <http://www.swsys.ru/uploaded/image/2023-3/2023-3-dop/7.jpg>). Тесты графического процессора выполняются на карте NVIDIA GeForce GTX 470. Время вычислений зависит от количества ограничений и вершин, участвующих в каждом ограничении, а не от их общего числа. Для каж-

## Статистические данные и время выполнения тестовых примеров

## Statistics and execution time of test cases

Тест	Количество патчей свободных форм	Количество функций возмущения	Количество итераций	Время вычислений, мс	Количество кадров в сек.
Тест 1	1 200	3 832	64	27	37
Тест 2	1 000	3 484	64	26	38
Тест 3	200	664	64	20	49

дого ограничения патча свободной формы требуются четыре вершины. Локальный шаг, обеспечивающий соблюдение ограничений, потребляет от 10 до 40 % затрат на динамику. В графическом процессоре не используются атомарные операции для передачи результатов обратно в вершины на локальном шаге. Результаты для каждой вершины получаются на глобальном шаге, стоимость которого зависит от доступа к памяти.

Комбинация подхода Чебышева и итераций Якоби была использована по нескольким причинам: она проста в реализации, совместима с ускорением графического процессора и имеет небольшую стоимость памяти. Подход Чебышева не требует повторной факторизации матрицы при каждом изменении системы, для чего необходимы большие вычислительные затраты.

Поведение сходимости метода Якоби без ускорения Чебышева показано на рисунке (см. <http://www.swsys.ru/uploaded/image/2023-3/2023-3-dop/10.jpg>). Метод Якоби решает линейную систему за один глобальный шаг. По оси ординат отражена ошибка, которая представляет собой величину погрешности линейной системы,

по оси абсцисс – количество итераций. Показаны три графика для разной степени жесткости.

Диаграмма точности на рисунке (см. <http://www.swsys.ru/uploaded/image/2023-3/2023-3-dop/11.jpg>) показывает производительность комбинированного метода Чебышева и Якоби по сравнению с другими на примере рисунка (см. <http://www.swsys.ru/uploaded/image/2023-3/2023-3-dop/9.jpg>). Время по шкале абсцисс измеряется в миллисекундах.

## Заключение

В работе представлен метод деформации трехмерных объектов на основе патчей свободных форм. Показано, что сходимость проективной динамики аналогична сходимости итерационного метода. В результате проективная динамика может быть эффективно ускорена с применением метода Чебышева.

Комбинация методов Якоби и Чебышева дополнительно позволяет эффективно обрабатывать позиционные и контактные ограничения на каждой итерации. Этот подход требует небольших затрат памяти и может обрабатывать большие временные шаги и деформации.

## Список литературы

1. Zhao J., Chen C., Feng Y., Wang Z. High effective simulation of solid-fluid coupling for complex scenes. *J. of Comp.-Aided Design and Comp. Graphics*, 2019, vol. 31, no. 8, art. 1358. doi: 10.3724/SP.J.1089.2019.17404.
2. Sommer A., Schwanecke U., Schoemer E. Chebyshev's method on projective fluids. *J. of WSCG*, 2020, vol. 28, no. 1-2, pp. 132–136. doi: 10.24132/JWSCG.2020.28.16.
3. Tao Y., Zheng Z., Zhong Y., Zhao J. SimulCap: Single-view human performance capture with cloth simulation. *Proc. IEEE/CVF Conf. on CVPR*, 2019, pp. 5499–5509. doi: 10.1109/CVPR.2019.00565.
4. Patel C., Liao Z., Pons-Moll G. TailorNet: Predicting clothing in 3D as a function of human pose, shape and garment style. *Proc. IEEE/CVF Conf. on CVPR*, 2020, pp. 7363–7373. doi: 10.1109/CVPR42600.2020.00739.
5. Niyazov A., Mellado N., Barthe L., Serrano M. Dynamic decals: Pervasive freeform interfaces using constrained deformable graphical elements. *Proc. ACM on Human-Comput. Interaction*, 2021, vol. 5, no. ISS, pp. 1–27. doi: 10.1145/3488538.
6. Li J., Liu X., Deng H., Wang T. Reconstruction of colored soft deformable objects based on self-generated template. *Comput.-Aided Design*, 2021, vol. 143, art. 103124. doi: 10.1016/j.cad.2021.103124.
7. Wang L., Li H., Xiao Q., Yao X. Automatic pose and wrinkle transfer for aesthetic garment display. *Comput. Aided Geometric Design*, 2021, vol. 89, art. 102020. doi: 10.1016/j.cagd.2021.102020.
8. Xian Z., Tong X., Liu T. A scalable galerkin multigrid method for real-time simulation of deformable objects. *ACM Transactions on Graphics*, 2019, vol. 38, no. 6, pp. 1–13. doi: 10.1145/3355089.3356486.
9. Lan L., Luo R., Fratarcangeli M., Xu W. Medial elastics: Efficient and collision-ready deformation via medial axis transform. *ACM Transactions on Graphics*, 2020, vol. 39, no. 3, pp. 1–17. doi: 10.1145/3384515.

10. Lan L., Yang Y., Kaufman D., Yao J. Medial IPC: Accelerated incremental potential contact with medial elastics. *ACM Transactions on Graphics*, 2021, vol. 40, no. 4, pp. 1–16. doi: 10.1145/3476576.3476741.
11. Shen S., Yang Y., Shao T., Wang H. High-order differentiable autoencoder for nonlinear model reduction. *ACM Transactions on Graphics*, 2021, vol. 40, no. 4, pp. 1–15. doi: 10.1145/3476576.3476620.
12. Fang Y., Li M., Jiang C., Kaufman D.M. Guaranteed globally injective 3D deformation processing. *ACM Transactions on Graphics*, 2021, vol. 40, no. 4, pp. 1–13. doi: 10.1145/3450626.3459757.
13. Вяткин С.И., Долговесов Б.С. Моделирование деформации эластичных объектов с использованием функций возмущения // Программные продукты и системы. 2022. Т. 35. № 4. С. 748–757. doi: 10.15827/0236-235X.140.748-757.
14. Denton P.B., Parke S.J., Tao T., Zhang X. Eigenvectors from eigenvalues: A survey of a basic identity in linear algebra. *Bull. Amer. Math. Soc.*, 2022, vol. 59, no. 1, pp. 31–58. doi: 10.1090/bull/1722.
15. Liu C., Han X., Li J. The Chebyshev accelerating method for progressive iterative approximation. *Communications in Information and Systems*, 2017, vol. 17, no. 1, pp. 25–43.
16. Ascher U.M., Larionov E., Sheen S.H., Pai D.K. Simulating deformable objects for computer animation: A numerical perspective. *J. of Computational Dynamics*, 2021, vol. 9, no. 2, pp. 47–68. doi: 10.3934/jcd.2021021.

Software &amp; Systems

doi: 10.15827/0236-235X.142.414-422

2023, vol. 36, no. 3, pp. 414–422

### A method of deformation of functionally defined objects using graphics processing units

Sergey I. Vyatkin  
Boris S. Dolgovesov

#### For citation

Vyatkin, S.I., Dolgovesov, B.S. (2023) 'A method of deformation of functionally defined objects using graphics processing units', *Software & Systems*, 36(3), pp. 414–422 (in Russ.). doi: 10.15827/0236-235X.142.414-422

#### Article info

Received: 02.03.2023

After revision: 10.04.2023

Accepted: 12.04.2023

**Abstract.** The article discusses a method of modeling deformation of elastic functionally specified objects. Deformation and animation of three-dimensional objects are important functions, but require a large number of calculations. Modeling elastic dynamic objects is used in computer graphics, biomechanics and robotics applications. The aim of the work is to develop a method for modeling deformation of elastic objects on a graphics processor. The subject of the study is the Chebyshev and Jacobi methods and their use in projective dynamics. The theoretical significance of the work is in the combination of Chebyshev and Jacobi methods for solving problems of deformation and animation of surfaces, where the problem is a large number of calculations. Projective dynamics can be effectively accelerated using the Chebyshev method. The practical significance is in implementing the method on a GPU. The combination of the Chebyshev method and the Jacobi method for projective dynamics is optimally implemented on a GPU, since the calculations are well parallelized. As a result, we get a simple implementation that does not require additional libraries to solve this problem and has a small memory. The method does not need to re-factorize the matrix with each system change, which is used in direct calculation, and requires large computational costs. When testing the proposed method, we used free form patches based on grids of basic triangles. Testing shows that this approach accelerates the projective dynamics by about one order of magnitude when the Jacobi solution is used at the global stage.

**Keywords:** animation, deformation, free-form patch, Jacobi method, Chebyshev method, projective dynamics, parallel computing, GPU

## Reference List

1. Zhao, J., Chen, C., Feng, Y., Wang, Z. (2019) 'High effective simulation of solid-fluid coupling for complex scenes', *J. of Comp.-Aided Design and Comp. Graphics*, 31(8), art. 1358. doi: 10.3724/SP.J.1089.2019.17404.
2. Sommer, A., Schwanecke, U., Schoemer, E. (2020) 'Chebyshev's method on projective fluids', *J. of WSCG*, 28(1-2), pp. 132–136. doi: 10.24132/JWSCG.2020.28.16.
3. Tao, Y., Zheng, Z., Zhong, Y., Zhao, J. (2019) 'SimulCap: Single-view human performance capture with cloth simulation', *Proc. IEEE/CVF Conf. on CVPR*, pp. 5499–5509. doi: 10.1109/CVPR.2019.00565.
4. Patel, C., Liao, Z., Pons-Moll, G. (2020) 'TailorNet: Predicting clothing in 3D as a function of human pose, shape and garment style', *Proc. IEEE/CVF Conf. on CVPR*, pp. 7363–7373. doi: 10.1109/CVPR42600.2020.00739.
5. Niyazov, A., Mellado, N., Barthe, L., Serrano, M. (2021) 'Dynamic decals: Pervasive freeform interfaces using constrained deformable graphical elements', *Proc. ACM on Human-Comput. Interaction*, 5(ISS), pp. 1–27. doi: 10.1145/3488538.
6. Li, J., Liu, X., Deng, H., Wang, T. (2021) 'Reconstruction of colored soft deformable objects based on self-generated template', *Comput.-Aided Design*, 143, art. 103124. doi: 10.1016/j.cad.2021.103124.
7. Wang, L., Li, H., Xiao, Q., Yao, X. (2021) 'Automatic pose and wrinkle transfer for aesthetic garment display', *Comput. Aided Geometric Design*, 89, art. 102020. doi: 10.1016/j.cagd.2021.102020.
8. Xian, Z., Tong, X., Liu, T. (2019) 'A scalable galerkin multigrid method for real-time simulation of deformable objects', *ACM Transactions on Graphics*, 38(6), pp. 1–13. doi: 10.1145/3355089.3356486.
9. Lan, L., Luo, R., Fratarcangeli, M., Xu, W. (2020) 'Medial elastics: Efficient and collision-ready deformation via medial axis transform', *ACM Transactions on Graphics*, 39(3), pp. 1–17. doi: 10.1145/3384515.
10. Lan, L., Yang, Y., Kaufman, D., Yao, J. (2021) 'Medial IPC: Accelerated incremental potential contact with medial elastics', *ACM Transactions on Graphics*, 40(4), pp. 1–16. doi: 10.1145/3476576.3476741.
11. Shen, S., Yang, Y., Shao, T., Wang, H. (2021) 'High-order differentiable autoencoder for nonlinear model reduction', *ACM Transactions on Graphics*, 40(4), pp. 1–15. doi: 10.1145/3476576.3476620.
12. Fang, Y., Li, M., Jiang, C., Kaufman, D.M. (2021) 'Guaranteed globally injective 3D deformation processing', *ACM Transactions on Graphics*, 40(4), pp. 1–13. doi: 10.1145/3450626.3459757.
13. Vyatkin, S.I., Dolgovesov, B.S. (2022) 'Modelling of deformation of elastic objects using perturbation functions', *Software & Systems*, 35(4), pp. 748–757 (in Russ.). doi: 10.15827/0236-235X.140.748-757.
14. Denton, P.B., Parke, S.J., Tao, T., Zhang, X. (2022) 'Eigenvectors from eigenvalues: A survey of a basic identity in linear algebra', *Bull. Amer. Math. Soc.*, 59(1), pp. 31–58. doi: 10.1090/bull/1722.
15. Liu, C., Han, X., Li, J. (2017) 'The Chebyshev accelerating method for progressive iterative approximation', *Communications in Information and Systems*, 17(1), pp. 25–43.
16. Ascher, U.M., Larionov, E., Sheen, S.H., Pai, D.K. (2021) 'Simulating deformable objects for computer animation: A numerical perspective', *J. of Computational Dynamics*, 9(2), pp. 47–68. doi: 10.3934/jcd.2021021.

## Авторы

**Вяткин Сергей Иванович**<sup>1</sup>, к.т.н.,  
старший научный сотрудник, sivser@mail.ru  
**Долгovesов Борис Степанович**<sup>1</sup>, к.т.н.,  
зав. лабораторией, bsd@iae.nsk.su

## Authors

**Sergey I. Vyatkin**<sup>1</sup>, Ph.D. (Engineering),  
Senior Researcher, sivser@mail.ru  
**Boris S. Dolgovesov**<sup>1</sup>, Ph.D. (Engineering),  
Head of Laboratory, bsd@iae.nsk.su

<sup>1</sup> Институт автоматизации и электрометрии СО РАН,  
г. Новосибирск, 630090,  
Россия

<sup>1</sup> Institute of Automation and Electrometry,  
Siberian Branch, Russian Academy of Sciences,  
Novosibirsk, 630090, Russian Federation

## Компенсация разных типов неопределенностей при управлении техническим объектом с помощью интеллектуальных регуляторов

В.В. Игнатьев

### Ссылка для цитирования

Игнатьев В.В. Компенсация разных типов неопределенностей при управлении техническим объектом с помощью интеллектуальных регуляторов // Программные продукты и системы. 2023. Т. 36. № 3. С. 423–431. doi: 10.15827/0236-235X.142.423-431

### Информация о статье

Поступила в редакцию: 27.06.2023

После доработки: 30.06.2023

Принята к публикации: 14.07.2023

**Аннотация.** В статье приводятся результаты исследований по разработке гибридных интеллектуальных регуляторов, позволяющих обеспечить эффективность управления техническим объектом, в том числе функционирующим в условиях неопределенности. Рассмотрены вопросы компенсации разных типов неопределенностей при управлении техническим объектом с применением интеллектуальных регуляторов. Дана обобщенная классификация неопределенностей и предложена новая с определением тех ее видов, компенсация которых достигается с помощью разработанных методов и алгоритмов. Такими видами неопределенностей являются параметрическая неопределенность, неопределенности, вызванные внешними воздействиями, лингвистическая неопределенность. Отдельно выделена лингвистическая неопределенность, приведены способы ее компенсации, основанные на подходе управления, который базируется на сочетании классической теории управления, нечеткой логики, нейросетевых технологий и генетических алгоритмов. В рамках демонстрации подхода в общем виде описан весь процесс получения желаемого управления для технического объекта, в том числе функционирующего в условиях неопределенности. Представлены результаты работы созданных методов управления техническими объектами с применением интеллектуальных регуляторов на основе самоорганизации баз знаний. Основу методов составляет алгоритм самоорганизации робастных баз знаний без привлечения эксперта с автоматизацией процесса генерации управляющих правил. Реализация алгоритма в моделях управления способствует получению желаемого управления для технических объектов, включая функционирующие в условиях неопределенности, представленных линейными или нелинейными математическими моделями первого, второго и третьего порядков, в том числе с запаздыванием. Результаты моделирования подтверждают, что применение подхода с гибридным управлением позволяет получать желаемое управление техническим объектом, в том числе функционирующим в условиях неопределенности.

**Ключевые слова:** управление, технический объект, интеллектуальный регулятор, синтез, обучение, компенсация неопределенности

Современные автоматизированные системы управления сложно представить без использования интеллектуальных технологий. Их внедрение обусловлено прежде всего возможностью гибкой подстройки под сложные задачи управления техническими объектами для обеспечения требуемой точности и надежности.

Особую значимость применение интеллектуальных технологий приобретает при наличии неопределенностей, от которых зависит эффективность не только управления объектом, но и всего производственного процесса. К неопределенностям, негативно влияющим на управление, можно отнести отсутствие информации о его цели, характеристиках и параметрах объекта управления, свойствах процесса управления, внешних условиях. Системы управления с применением интеллектуальных технологий позволяют устранять отдельные виды неопределенностей, но не все и, что особенно важно, не в режиме реального времени. В связи с этим при проектировании таких систем необходимо разрабатывать новые науч-

ные подходы, гарантирующие эффективное управление с требуемой точностью и надежностью.

В данной статье выполнен обзор разных видов неопределенностей, приведена их классификация, продемонстрирована работа созданных автором методов и алгоритмов управления техническими объектами с помощью интеллектуальных регуляторов, позволяющих для заданных классов объектов компенсировать определенные типы неопределенностей.

### Обзор исследований по научной проблеме

Созданию эффективных систем управления, спроектированных на основе интеллектуальных регуляторов, посвящено немало научных исследований. Рассмотрим некоторые из них, наиболее близкие к теме данной работы.

Задача разработки нечеткого пропорционально-интегрального регулятора на основе оптимальной функции принадлежности для управления мощностью активной зоны жидко-солевого реактора-размножителя решается в

исследовании [1]. Ее основная идея – автоматическая настройка параметров классического регулятора с помощью нечеткого. Для этого при проектировании нечеткого регулятора реализуется процедура оптимизации функций принадлежности на основе коэффициентов нелинейного динамического ускорения.

Исследование [2] посвящено разработке пропорционально-интегрально-дифференциального регулятора на нечеткой логике с оптимальным масштабированием функций принадлежности для системы вентиляции легких. Как и в работе [1], нечеткий регулятор используется для настройки классического (в данном случае ПИД-регулятора). Интерес представляет то, что функция принадлежности нечетких входных данных (ошибка и изменение ошибки по давлению в дыхательных путях) и выходных сигналов (параметров ПИД-регулятора) оптимизируется с помощью метода роя частиц.

Авторами [3] разработан адаптивный нечеткий регулятор на основе раннего обнаружения и предотвращения асимметричных неисправностей в энергосистемах. Для краткосрочного прогнозирования нагрузки используются нейронные сети с целью наблюдения за нелинейным поведением вследствие асимметричных сбоев в профиле потребления энергосистемы. Данный нечеткий регулятор отличается возможностью работы в режиме реального времени и предназначен для обнаружения колебаний, которые невозможно отследить с помощью традиционных методов прогнозирования нагрузки.

Робастный адаптивно-нечетко-пропорционально-производный регулятор для реабилитационного экзоскелета нижних конечностей предложен в работе [4]. Объектом управления является двигатель постоянного тока в качестве привода сустава. В регуляторе для достижения требуемой работы объекта управления реализуется новая оптимизация роя частиц, основанная на инициализации эталонной модели. Следует отметить, что спроектированный регулятор работает в режиме реального времени.

Управлению выходным напряжением двухкамерного микробного топливного элемента с использованием оптимизированного адаптивного нейро-нечеткого регулятора на основе интеллектуальных данных посвящена работа [5]. В регуляторе реализованы алгоритмы оптимизации роя частиц и алгоритм серых волков для настройки его параметров. Полученные результаты показали, что данный регулятор об-

ладает характеристиками, позволяющими быстро и эффективно отслеживать напряжение при внешних изменениях нагрузки, распределениях и неопределенности параметров.

В статье [6] предложена новая конструкция нечеткого адаптивного ПИД-регулятора скорости для современной лифтовой тяговой машины. Продемонстрировано, что для заданного объекта управления нечеткий адаптивный ПИД-регулятор имеет более лучшие характеристики управления по сравнению с традиционными методами ПИД-регулирования (в частности, получены лучшее время переходного процесса и меньшая величина статической ошибки).

Нечеткому регулятору, синтезированному на основе классического регулятора (ПИД), посвящена работа [7]. Исследование выполнено на примере нелинейной математической модели квадрокоптера. В системе управления анализируются сигналы ошибок и управления и с их помощью разрабатываются входные и выходные лингвистические переменные. Далее проектируется база нечетких правил. Предложенный метод позволяет исключить знания эксперта при проектировании базы правил. Разработанный нечеткий регулятор взаимодействует с ПИД-регулятором для поддержки его производительности и компенсирует неопределенности коэффициентов модели.

Нечеткий вибрационный регулятор на основе нейронной сети для морской платформы со случайной временной задержкой создан авторами работы [8]. Регулятор предназначен для ослабления вибрации морской платформы, вызванной нерегулярными волнами. Нейро-нечеткая система наблюдения, обученная на выборочных данных из нечетких моделей, реконструирует систему управления. Регулятор на основе нейронной сети эффективно работает в различных случаях задержки, обеспечивая лучшие характеристики ослабления вибрации при неопределенной задержке управления и случайных волнах по сравнению с существующими законами оптимального управления и нечеткими регуляторами без нейронных сетей.

В работе [9] рассмотрено управление для системы слежения телескопа. Применяется адаптивный нечеткий ПИД-регулятор, автоматически настраиваемый с помощью генетического подхода. Суть этого подхода заключается в автоматической настройке функций принадлежности и правил управления нечеткого регулятора за счет использования эволюционного алгоритма. Данный регулятор достаточно

эффективен при высокоточном управлении, однако нет пояснений относительно его поведения при возникновении разных типов неопределенностей, в том числе одновременно.

Из проведенного анализа научных изысканий следует, что для эффективного управления техническими объектами, в том числе в условиях неопределенности, чаще применяются гибридные методы с использованием интеллектуальных технологий. Таким образом, вопросы, рассмотренные в настоящей статье, и представленные решения с подтверждением результатов их реализации говорят об актуальности проводимых исследований.

**Процедура компенсации разных типов неопределенностей при управлении техническим объектом с помощью интеллектуальных регуляторов**

Данное исследование основано на ранее созданной концепции управления техническим объектом в условиях неопределенности и является продолжением изысканий по разработке методов синтеза интеллектуальных регуляторов и их оптимизации [10–12].

Концепция базируется на методах управления техническими объектами с помощью интеллектуальных регуляторов на основе самоорганизации баз знаний и алгоритме самоорганизации робастных баз знаний без привлечения эксперта в системах управления с интеллектуальными регуляторами. Реализация концепции позволяет получать желаемое управление техническим объектом, функционирующим в изменяющихся условиях.

В работах [13–15] рассматриваются подходы к управлению, в том числе и основанные на применении интеллектуальных технологий. Отдельное внимание уделяется видам неопределенностей, которые приводят к ухудшению точности работы системы управления или к полной потере ее работоспособности.

На основании анализа данных работ и ряда других можно обобщенно классифицировать неопределенности (рис. 1).

Разработанные автором [10–12] методы и алгоритмы позволяют решать задачу компенсации параметрической неопределенности, неопределенностей, вызванных внешними воздействиями, и лингвистической неопределенности (рис. 2, выделенный в блоках текст). Для компенсации указанных на рисунке неопределенностей реализуются методы и алгоритмы, работа которых основана на сочетании класси-

ческой теории управления, теории нечетких множеств, нейросетевых технологий и генетических алгоритмов [10].

Это позволяет эффективно управлять линейными и нелинейными математическими моделями первого, второго, третьего порядков, в том числе с запаздыванием (в контурах управления существуют запаздывания по управлению, по выходному сигналу, по измерению). Общий вид реализации такого подхода к управлению графически представлен на рисунке 3.

Разработанные в рамках подхода гибридные методы синтеза интеллектуального регулятора и алгоритмы управления на основе самоорганизации робастных баз знаний позволяют достичь желаемого управления техническим объектом в условиях неопределенности.

Ключевая особенность реализации гибридного управления заключается в ранее разработанном алгоритме самоорганизации робастных баз знаний без привлечения эксперта в системах управления с интеллектуальными регуля-

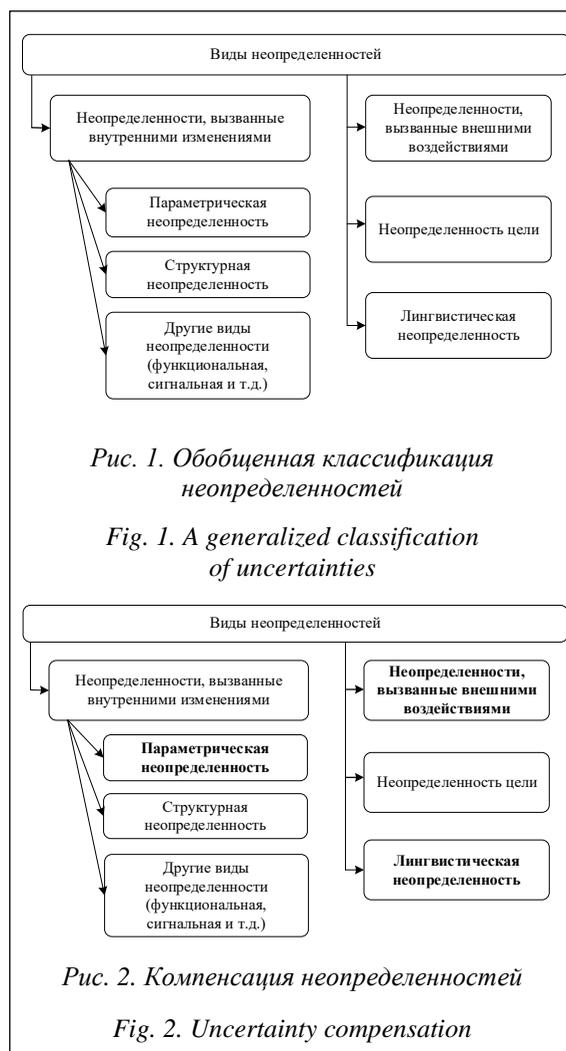
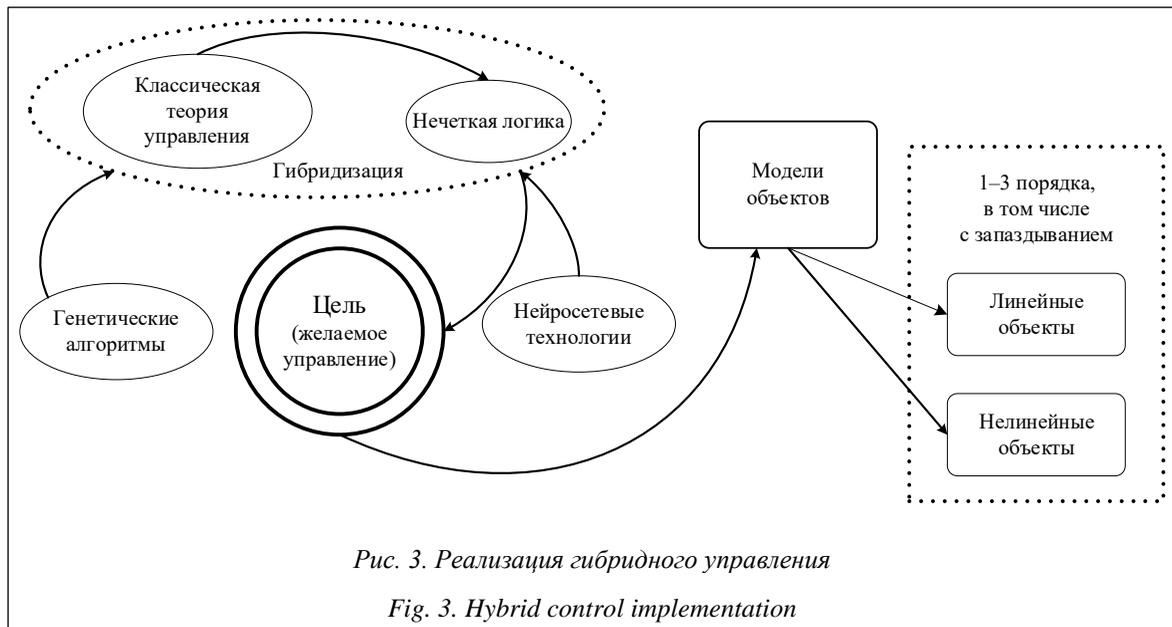


Рис. 1. Обобщенная классификация неопределенностей

Fig. 1. A generalized classification of uncertainties

Рис. 2. Компенсация неопределенностей

Fig. 2. Uncertainty compensation



торами (рис. 4). Блоки алгоритма и его работа подробно описаны в [12]. Алгоритм, представленный на рисунке 4, отличается от алгоритма из работы [12] наличием блока лингвистической неопределенности в изменяющихся условиях функционирования объекта управления.

Компенсация лингвистической неопределенности – достаточно сложная задача, поэтому рассмотрим ее более подробно. Когда получение адекватной математической модели является сложным процессом или вообще невозможно, при разработке системы управления используются знания (об объекте управления, его свойствах и/или характере поведения при функционировании) и опыт экспертов, разрабатываются регуляторы с применением технологий искусственного интеллекта.

С одной стороны, такие регуляторы позволяют исключить негативно влияющие на эффективность управления факторы, с другой – работа регулятора становится зависимой от знаний эксперта. В случае появления неопределенности знаний и опыта эксперта в предметной области может быть недостаточно, чтобы охватить все возможные состояния объекта. В связи с этим формирование и оптимальность базы правил интеллектуального регулятора напрямую зависят от квалификации эксперта, что может существенно снизить эффективность регулятора.

Лингвистическая неопределенность обусловлена расплывчатостью и неоднозначностью естественного языка, что приводит к трудностям формирования базы правил интел-

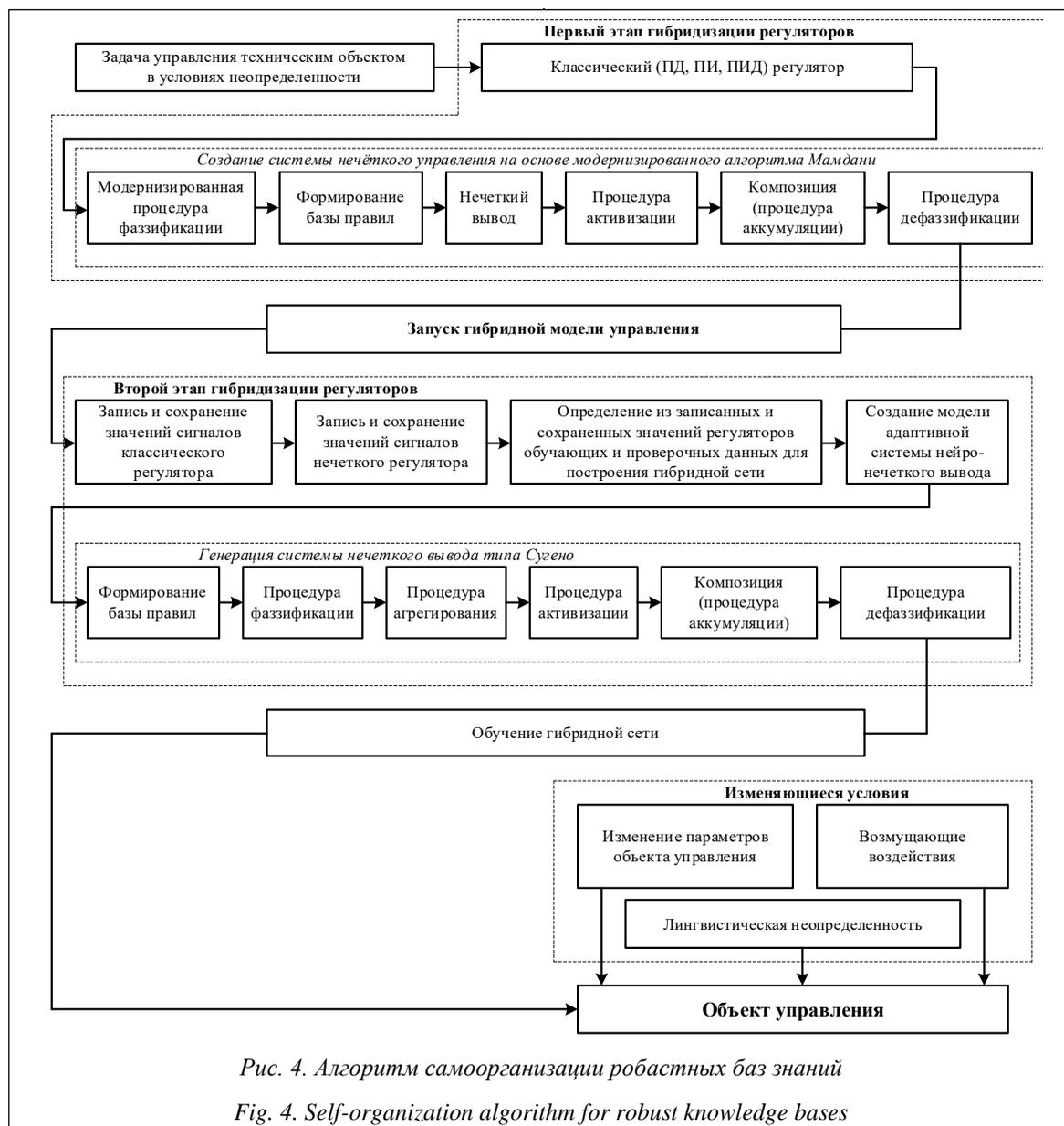
лектуального регулятора из-за необходимости определения вида, количества, параметров функций принадлежности лингвистических переменных.

Для решения вопросов избыточности и противоречивости данных в базе правил и получения желаемого управления, в том числе в непредвиденных ситуациях, следует исключать ошибки, обусловленные человеческим фактором. Представленный на рисунке 4 алгоритм самоорганизации робастных баз знаний позволяет решить обозначенные проблемы.

Кроме реализации процедуры самоорганизации робастных баз знаний без привлечения эксперта, в алгоритме автоматизирован процесс генерации управляющих правил систем с нечеткой логикой на основе данных классических регуляторов. В общем виде весь процесс получения желаемого управления для технического объекта, в том числе функционирующего в условиях неопределенности, заключается в прохождении следующей последовательности этапов.

На первом этапе для управления техническим объектом разрабатывается классический регулятор (ПИ, ПД, ПИД). После запуска модели управления с классического регулятора в одинаковые моменты времени снимаются значения сигналов отклонения системы, интеграла или дифференциала отклонения и управляющего воздействия. Значения каждого из сигналов сохраняются в файл в виде текстового массива.

На втором этапе синтезируется нечеткий регулятор, база правил которого формируется на



основе модифицированного алгоритма Мамдани в автоматическом режиме. После запуска модели проверяется качество полученного управления и уже с нечеткого регулятора в одинаковые моменты времени снимаются значения сигналов отклонения системы, интеграла или дифференциала отклонения и управляющего воздействия. Эти значения также сохраняются в файл в виде текстового массива.

На третьем этапе строится адаптивная система нейро-нечеткого вывода (ANFIS), в которой второй текстовый массив является обучающим, а первый необходим для проверки. Синтезируется система нечеткого вывода, запускается модель с новой структурой, проверяется качество полученного управления.

На четвертом этапе в определенной последовательности реализуется оптимизация:

- центров функций принадлежности входных лингвистических переменных;
- центров функций принадлежности выходной лингвистической переменной;
- операций агрегирования, активизации, аккумуляции, дефазификации алгоритма Мамдани;
- весовых коэффициентов базы нечетких правил.

В результате для разных объектов (линейных, в том числе с запаздыванием) получены переходные процессы с требуемыми показателями качества, примеры которых показаны на рисунке 5.

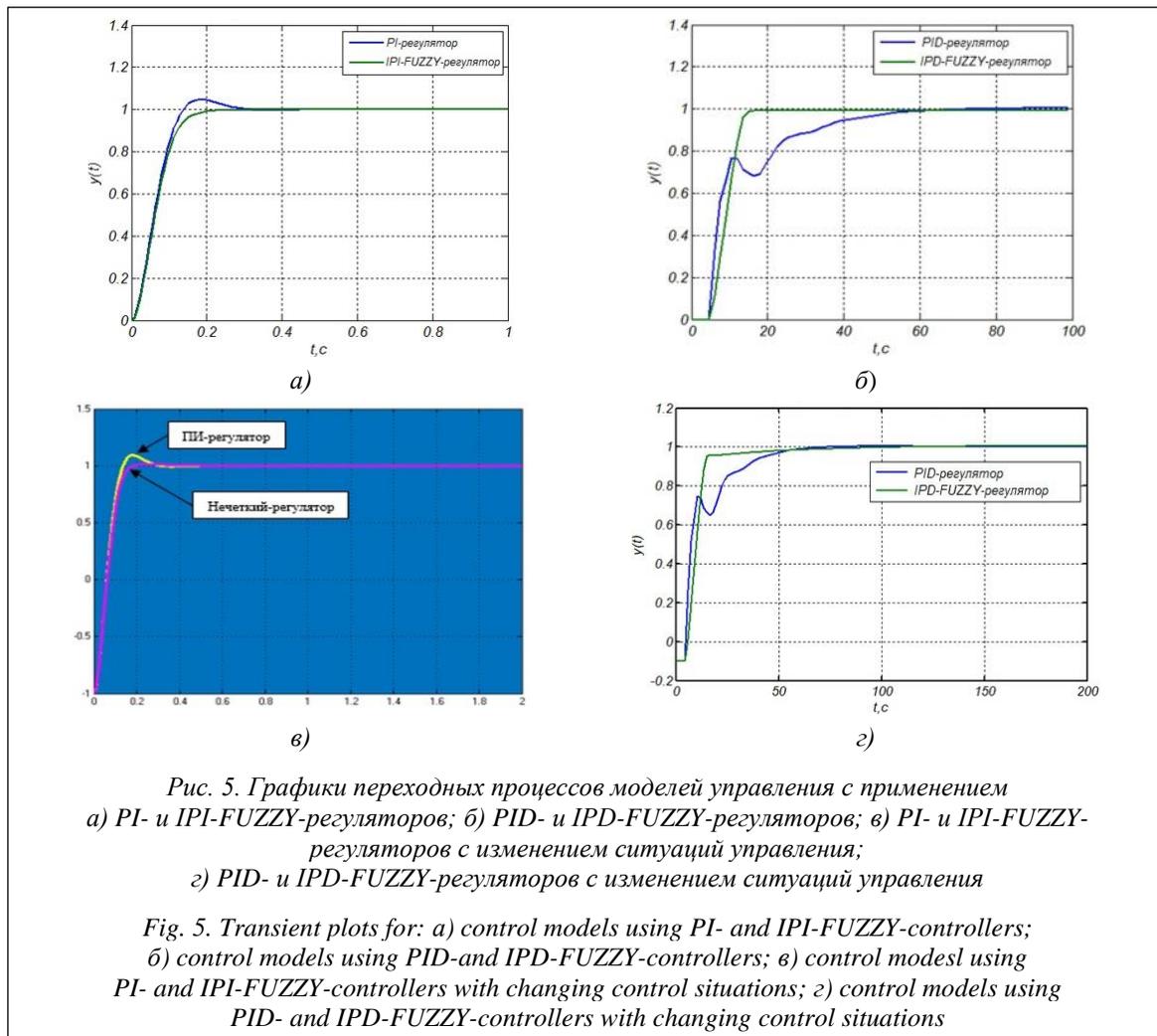


Рис. 5. Графики переходных процессов моделей управления с применением  
 а) PI- и IPI-FUZZY-регуляторов; б) PID- и IPD-FUZZY-регуляторов;  
 в) PI- и IPI-FUZZY-регуляторов с изменением ситуаций управления;  
 з) PID- и IPD-FUZZY-регуляторов с изменением ситуаций управления

Fig. 5. Transient plots for: а) control models using PI- and IPI-FUZZY-controllers;  
 б) control models using PID- and IPD-FUZZY-controllers; в) control models using  
 PI- and IPI-FUZZY-controllers with changing control situations; з) control models using  
 PID- and IPD-FUZZY-controllers with changing control situations

Следует отметить, что желаемое управление можно получить уже на втором этапе, однако оно может не обладать робастными свойствами и, как показали различные экспериментальные исследования, в отдельных случаях при изменении ситуаций управления требуемыми показателями качества. Поэтому разработаны и реализуются третий и четвертые этапы.

Примеры переходного процесса с требуемыми показателями качества для неустойчивого нелинейного технического объекта третьего порядка и реализации процедуры поиска оптимальных параметров нечеткого регулятора представлены на рисунках по адресу <http://www.swsys.ru/uploaded/image/2023-3/2023-3-dop/32.jpg>.

### Анализ результатов

Полученные графики переходных процессов для разных технических объектов свидетельствуют о достоинствах разработанных методов управления, основанных на гибридном

подходе, включающем комбинирование классической теории управления, нечеткой логики, нейросетей и генетических алгоритмов. Для достижения желаемого управления подход может как реализоваться при выполнении всей последовательности этапов, так и закончиться на втором или третьем этапе в зависимости от требований к управлению.

Однозначно можно сказать, что показатели управления при реализации такого подхода значительно выше, чем при применении традиционных методов управления. Кроме того, система управления техническим объектом, построенная с применением интеллектуальных регуляторов, отличается повышенной робастностью и позволяет компенсировать разные типы неопределенностей.

### Обсуждение

Применя представленные в статье научные решения в прикладных задачах управления, связанных с автоматизацией техноло-

гических процессов, необходимо понимать, каких целей требуется достичь. Например, представленный на рисунке 4 алгоритм самоорганизации робастных баз знаний отличается сложным взаимодействием различных технологий управления (классические методы, нечеткие, нейросети, генетические алгоритмы) и предназначен для получения желаемого управления сразу в условиях разных типов неопределенностей. В отдельных случаях достаточно на основе классического регулятора с помощью разработанных методов и алгоритмов построить эффективный нечеткий регулятор с заданными параметрами (имеется возможность их изменения).

То же самое можно сказать и о разработанной процедуре поиска оптимальных параметров нечеткого регулятора: исходя из полученного качества управления делается заключение о необходимости ее применения. Вне зависимости от указанных задач разработанные методы и алгоритмы позволяют успешно решить их. Это дает возможность в значительной сте-

пени снизить трудозатраты на разработку эффективного регулятора.

### Заключение

В статье представлен новый алгоритм, в котором, кроме параметрической неопределенности и неопределенностей, вызванных внешними воздействиями, выделена лингвистическая неопределенность.

В подходе управления нечеткий регулятор строится на основе данных с классического регулятора, а не является источником его настройки. Это позволяет более гибко реагировать на возникающие в процессе управления изменения.

Интеллектуальный регулятор изначально строится на данных, полученных с классического регулятора. Это дает возможность спроектировать базу правил интеллектуального регулятора, обладающую робастными свойствами (на основе всех разработанных методов и алгоритмов).

### Список литературы

1. Acharya D., Das D.K. Optimal membership function based fuzzy proportional-integral controller for power control of molten salt breeder reactor core. *Progress in Nuclear Energy*, 2023, vol. 161, art. 104753. doi: 10.1016/j.pnucene.2023.104753.
2. Acharya D., Das D.K. Design of a fuzzy-based proportional integral derivative controller with optimal membership function scaling for respiratory ventilation system. *Biomedical Signal Processing and Control*, 2022, vol. 78, art. 103938. doi: 10.1016/j.bspc.2022.103938.
3. Ali M., Kaur K., Adnan M., Nisar Sh. Adaptive fuzzy controller based early detection and prevention of asymmetrical faults in power systems. *Control Engineering Practice*, 2023, vol. 130, art. 105380. doi: 10.1016/j.conengprac.2022.105380.
4. Aliman N., Ramli R., Haris S.M., Amiri M.S., Van M. A robust adaptive-fuzzy-proportional-derivative controller for a rehabilitation lower limb exoskeleton. *JESTECH*, 2022, vol. 35, art. 101097. doi: 10.1016/j.jestech.2022.101097.
5. Demir M.H., Eren B. Output voltage control of double chambers microbial fuel cell using intelligence-based optimized adaptive neuro fuzzy inference controller. *Int. J. of Hydrogen Energy*, 2022, vol. 47, no. 45, pp. 19837–19849. doi: 10.1016/j.ijhydene.2022.03.113.
6. Duan X., Zhi P., Zhu W., Wei H. Fuzzy adaptive PID speed controller design for modern elevator traction machine. *Energy Reports*, 2023, vol. 9, pp. 175–183. doi: 10.1016/j.egy.2023.04.262.
7. Alsafadi L.A., Chulin N.A., Mironova I.V. Synthesis of fuzzy controller based on simple PID controller. *Procedia Comput. Sci.*, 2019, vol. 150, pp. 28–38. doi: 10.1016/j.procs.2019.02.008.
8. Zhang Y., Ma H., Xu J. Neural network-based fuzzy vibration controller for offshore platform with random time delay. *Ocean Engineering*, 2021, vol. 225, art. 108733. doi: 10.1016/j.oceaneng.2021.108733.
9. Demidova G.L., Lukichev D.V., Kuzin A.Yu. A Genetic approach for auto-tuning of adaptive fuzzy PID control of a telescope's tracking system. *Procedia Comput. Sci.*, 2019, vol. 150, pp. 495–502. doi: 10.1016/j.procs.2019.02.084.
10. Игнатъев В.В., Шацкий В.Б. Концепция управления техническим объектом в условиях неопределенности // *Актуальные вопросы науки*. 2020. № 70. С. 28–31.
11. Игнатъев В.В. Методы управления техническими объектами с помощью интеллектуальных регуляторов на основе самоорганизации баз знаний. Ростов-на-Дону; Таганрог: изд-во ЮФУ, 2020. 142 с. doi: 10.18522/801273622.
12. Игнатъев В.В. Алгоритм самоорганизации робастных баз знаний без привлечения эксперта в системах управления с интеллектуальными регуляторами // *ИС&ИТ-2020: тр. междунар. науч.-технич. конгресса*. 2020. Т. 1. С. 362–377.
13. Никифоров В.О., Слита О.В., Ушаков А.В. Интеллектуальное управление в условиях неопределенности. СПб: изд-во СПбГУ ИТМО, 2009. 232 с.
14. Збрищак С.Г., Звягин Л.С. Неопределённость в задачах моделирования и управления сложных, слабо формализуемых многокомпонентных систем // *Междунар. конф. по мягким вычислениям и измерениям*. 2020. Т. 1. С. 15–19.

15. Никифоров В.О., Ушаков А.А. Управление в условиях неопределенности: чувствительность, адаптация и робастность. СПб: изд-во СПбГУ ИТМО, 2003. 232 с.

Software &amp; Systems

doi: 10.15827/0236-235X.142.423-431

2023, vol. 36, no. 3, pp. 423–431

### Compensation of various types of uncertainties when controlling a technical object using intelligent controllers

Vladimir V. Ignatyev

**For citation**

Ignatyev, V.V. (2023) 'Compensation of various types of uncertainties when controlling a technical object using intelligent controllers', *Software & Systems*, 36(3), pp. 423–431 (in Russ.). doi: 10.15827/0236-235X.142.423-431

**Article info**

Received: 27.06.2023

After revision: 30.06.2023

Accepted: 14.07.2023

**Abstract.** The article presents the research results on the development of hybrid intelligent controllers, which make it possible to ensure the effectiveness of the technical object control, including one functioning under uncertainty. The article is devoted to the issues of compensation of various types of uncertainties in controlling a technical object using intelligent controllers. It also gives a generalized classification of uncertainties and proposes a new classification with the definition of its types that are compensated using the developed methods and algorithms. The uncertainty types are parametric uncertainty, uncertainties caused by external influences, linguistic uncertainty. The author highlights linguistic uncertainty and gives methods for its compensation based on a control approach based on a combination of a classical control theory, fuzzy logic, neural network technologies and genetic algorithms. As a part of the demonstration of the approach, the author presents a general description of the entire process of obtaining the desired control for a technical object, including the one operating under conditions of uncertainty. The paper demonstrates the results of the work of the developed methods for controlling technical objects using intelligent controllers based on self-organization of knowledge bases. The basis of the methods is a self-organization algorithm of robust knowledge bases without involving an expert and automating the process of generating control rules. The implementation of these algorithms in control models makes it possible to obtain the desired control for technical objects including those functioning under uncertainty represented by linear or nonlinear mathematical models of the first, second, third order, including the ones with a delay. The presented simulation results confirm that the implementation of the hybrid control approach allows obtaining the desired control of a technical object including the one functioning under uncertainty.

**Keywords:** control, technical object, intelligent controller, synthesis, learning, uncertainty compensation

**Reference List**

1. Acharya, D., Das, D.K. (2023) 'Optimal membership function based fuzzy proportional-integral controller for power control of molten salt breeder reactor core', *Progress in Nuclear Energy*, 161, art. 104753. doi: 10.1016/j.pnucene.2023.104753.
2. Acharya, D., Das, D.K. (2022) 'Design of a fuzzy-based proportional integral derivative controller with optimal membership function scaling for respiratory ventilation system', *Biomedical Signal Processing and Control*, 78, art. 103938. doi: 10.1016/j.bspc.2022.103938.
3. Ali, M., Kaur, K., Adnan, M., Nisar, Sh. (2023) 'Adaptive fuzzy controller based early detection and prevention of asymmetrical faults in power systems', *Control Engineering Practice*, 130, art. 105380. doi: 10.1016/j.conengprac.2022.105380.
4. Aliman, N., Ramli, R., Haris, S.M., Amiri, M.S., Van, M. (2022) 'A robust adaptive-fuzzy-proportional-derivative controller for a rehabilitation lower limb exoskeleton', *JESTECH*, 35, art. 101097. doi: 10.1016/j.jestech.2022.101097.
5. Demir, M.H., Eren, B. (2022) 'Output voltage control of double chambers microbial fuel cell using intelligence-based optimized adaptive neuro fuzzy inference controller', *Int. J. of Hydrogen Energy*, 47(45), pp. 19837–19849. doi: 10.1016/j.ijhydene.2022.03.113.

6. Duan, X., Zhi, P., Zhu, W., Wei, H. (2023) 'Fuzzy adaptive PID speed controller design for modern elevator traction machine', *Energy Reports*, 9, pp. 175–183. doi: 10.1016/j.egy.2023.04.262.
7. Alsafadi, L.A., Chulin, N.A., Mironova, I.V. (2019) 'Synthesis of fuzzy controller based on simple PID controller', *Procedia Comput. Sci.*, 150, pp. 28–38. doi: 10.1016/j.procs.2019.02.008.
8. Zhang, Y., Ma, H., Xu, J. (2021) 'Neural network-based fuzzy vibration controller for offshore platform with random time delay', *Ocean Engineering*, 225, art. 108733. doi: 10.1016/j.oceaneng.2021.108733.
9. Demidova, G.L., Lukichev, D.V., Kuzin, A.Yu. (2019) 'A genetic approach for auto-tuning of adaptive fuzzy PID control of a telescope's tracking system', *Procedia Comput. Sci.*, 150, pp. 495–502. doi: 10.1016/j.procs.2019.02.084.
10. Ignatyev, V.V., Shatskiy, V.B. (2020) 'The concept of managing a technical object under uncertainty', *Actual Issues of Sci.*, (70), pp. 28–31 (in Russ.).
11. Ignatyev, V.V. (2020) *Methods for Controlling Technical Objects Using Intelligent Controllers Based on Self-organization of Knowledge Bases*. Rostov-on-Don; Taganrog, 142 p. (in Russ.). doi: 10.18522/801273622.
12. Ignatyev, V.V. (2020) 'Algorithm for self-organization of robust knowledge bases without involving an expert in control systems with intelligent controllers', *Proc. IS&IT-2020*, 1, pp. 362–377 (in Russ.).
13. Nikiforov, V.O., Slita, O.V., Ushakov, A.V. (2009) *Intelligent Control Under Uncertainty*. St. Petersburg, 232 p. (in Russ.).
14. Zbrischak, S.G., Zvyagin, L.S. (2020) 'Uncertainty in modeling and control problems of complex, poorly formalized multicomponent systems', *Proc. SCM*, 1, pp. 15–19 (in Russ.).
15. Nikiforov, V.O., Ushakov, A.A. (2003) *Management Under Uncertainty: Sensitivity, Adaptation and Robustness*. St. Petersburg, 232 p. (in Russ.).

**Авторы**

**Игнатьев Владимир Владимирович**<sup>1</sup>, к.т.н.,  
ведущий научный сотрудник, vvignatev@sfedu.ru

<sup>1</sup> Южный федеральный университет,  
г. Таганрог, 347900, Россия

**Authors**

**Vladimir V. Ignatyev**<sup>1</sup>, Ph.D. (Engineering),  
Leading Researcher, vvignatev@sfedu.ru

<sup>1</sup> Southern Federal University,  
Taganrog, 347900, Russian Federation

## Контроль подключений USB-устройств в локальной вычислительной сети компьютеров под управлением Astra Linux SE

А.В. Баранов  
П.М. Корепанов  
И.А. Лепешев

### Ссылка для цитирования

Баранов А.В., Корепанов П.М., Лепешев И.А. Контроль подключений USB-устройств в локальной вычислительной сети компьютеров под управлением Astra Linux SE // Программные продукты и системы. 2023. Т. 36. № 3. С. 432–441. doi: 10.15827/0236-235X.142.432-441

### Информация о статье

Поступила в редакцию: 19.06.2023

После доработки: 28.06.2023

Принята к публикации: 03.07.2023

**Аннотация.** Рост требований к информационной безопасности, а также тенденция к импортозамещению в области системного ПО обусловили широкое распространение инфраструктурных решений, построенных на базе отечественной операционной системы Astra Linux Special Edition (Astra Linux SE). Применение Astra Linux SE позволяет строить защищенные программно-аппаратные системы для обработки информации ограниченного доступа, в том числе в научных суперкомпьютерных центрах. При этом одним из важнейших аспектов обеспечения информационной безопасности является контроль подключения USB-устройств к компьютерам в локальной вычислительной сети. Анализ доступных современных источников показывает, что готовых комплексных решений, работающих в среде Astra Linux SE, в настоящее время не существует. В статье рассмотрен возможный технологический стек подобного решения, включающий, помимо Astra Linux SE, систему организации очередей сообщений RabbitMQ, микрофреймворк для разработки web-приложений Flask, СУБД PostgreSQL, а также средство выявления подключений USB-устройств USBRip. Рассмотрена предложенная на базе технологического стека модульная структура программной системы аудита подключений USB-устройств, включающая модули сбора информации о USB-подключениях на контролируемых компьютерах, модуль агрегации собранной информации на серверной стороне и модуль проверки легитимности выявленных подключений USB-устройств к контролируемым компьютерам под управлением Astra Linux SE. Предложенные структура и технологический стек реализованы в виде макета программной системы, получившей название ALUMNUS. Макет был развернут и прошел опытную эксплуатацию в защищенном сегменте суперкомпьютера МВС-10П ОП, установленном в Межведомственном суперкомпьютерном центре РАН.

**Ключевые слова:** суперкомпьютерный центр, информационная безопасность, Astra Linux SE, контроль USB-подключений

**Благодарности.** Работа выполнена в МЦЦ РАН в рамках государственного задания по теме FNEF-2022-0016

Информационная безопасность научных суперкомпьютерных центров долгое время во многом обеспечивалась за счет механизмов защиты, встроенных в *операционные системы* (ОС) вычислительных узлов и сетевые устройства суперкомпьютеров, а также организационно-технических мер, базирующихся на этих механизмах. В связи с постоянным ростом компьютерных угроз и повышением требований к информационной безопасности вычислительно-информационная инфраструктура все чаще строится на базе системного ПО, сертифицированного по требованиям безопасности информации. Широкое распространение в последние годы получили инфраструктурные решения на базе отечественной ОС Astra Linux [1], которая имеет множество модификаций, в том числе *Astra Linux Special Edition* (Astra Linux SE).

Astra Linux SE представляет собой ОС для обработки информации ограниченного доступа, в том числе содержащей сведения, составляющие государственную тайну. Система обладает

широкими возможностями [2, 3] по защите информации, позволяющими создавать на своей основе автоматизированные и информационные системы в защищенном исполнении. Важную роль Astra Linux играет в процессе импортозамещения системного ПО. Как отмечается в работе [4], важнейшими отличительными особенностями разработки и обеспечения доверия к ОС Astra Linux SE являются реализация в ней собственной подсистемы безопасности PARSEC и отказ от применения аналогичных заимствованных иностранных механизмов, в том числе SELinux [5] и AppArmor [6]. Подчеркивается, что Astra Linux SE – единственное отечественное решение в классе ОС общего назначения, получившее сертификаты соответствия по самым высоким классам защиты в системах сертификации ФСТЭК, ФСБ и Минобороны России [7]. При этом Astra Linux SE – это единая платформа для всех типов устройств [1] на базе процессоров x86-64, ARM, «Эльбрус», «Байкал», «Комдив», совместимая с большин-

ством аппаратного оборудования, в том числе отечественного производства. ОС широко применяется для обеспечения безопасности информации, а также для защиты объектов критической информационной инфраструктуры.

Важными аспектами обеспечения информационной безопасности являются контроль и учет подключаемых внешних USB-устройств в защищенном сегменте *локальной вычислительной сети* (ЛВС). Базовые средства Astra Linux SE не позволяют вести централизованный учет подключаемых USB-устройств к компьютерам в ЛВС. Впервые эта проблема и методы ее решения были рассмотрены в работе [8], в которой справедливо отмечается, что угроза использования неучтенных USB-устройств может привести к снижению уровня информационной безопасности и образованию канала утечки информации в защищенном сегменте ЛВС. В работах [8, 9] рассматриваются следующие методы учета подключений USB-устройств в среде Astra Linux:

- ручная настройка учета подключений USB-устройств, а именно генерация определенных правил для каждого устройства, согласно которым ОС разрешает или не разрешает использование того или иного устройства;
- использование встроенной в Astra Linux SE графической утилиты fly-admin-smc;
- использование службы Astra Linux Directory (ALD) [10].

Первые два метода требуют ручной работы администратора защищенного сегмента, что при наличии большого числа контролируемых компьютеров обуславливает значительную трудоемкость и высокую вероятность ошибок [8]. В основу домена ALD [3] положен принцип объединения логически связанных сетевых ресурсов и учетных записей пользователей в единую систему авторизации с централизованным управлением и мандатным разграничением доступа пользователей к информации. Использование службы ALD требует включения в его домен всех компьютеров, на которых предполагается вести аудит подключения USB-устройств, что далеко не всегда можно осуществить на практике.

В публикации [11] в качестве базового метода обнаружения подключения устройств USB предлагается метод анализа системных журналов контролируемого компьютера. Он рассмотрен в самых общих чертах, применим для контроля отдельного компьютера или автоматизированного рабочего места. Способы

и средства применения метода анализа системных журналов на множестве контролируемых компьютеров в ЛВС не рассматриваются.

Среди существующих средств учета подключений USB-устройств в среде Astra Linux следует отметить программы для ЭВМ [12, 13]. Программа, рассмотренная в [12], является результатом исследований [8] и предназначена для автоматизации процесса учета и контроля использования машинных носителей информации на средствах вычислительной техники. В ее состав входят подсистемы учета и контроля использования машинных носителей информации на средствах вычислительной техники в организации. Подсистема контроля функционально разделяется на агента, непрерывно функционирующего на контролируемых компьютерах, и менеджера, принимающего данные от агентов и сигнализирующего администратору об использовании неучтенных носителей. В программе, предложенной в [13], реализован аудит подключения USB-устройств к отдельному персональному компьютеру под управлением Astra Linux, не включенному в состав ЛВС. Обе программы [12, 13] отсутствуют в свободном доступе, информация о них, полученная из рефератов свидетельств о государственной регистрации программ для ЭВМ, не содержит сведений о способах и методиках построения, структуре и стеке технологических решений.

Задача контроля подключений устройств USB к компьютерам в ЛВС может быть решена за счет применения систем управления событиями информационной безопасности (систем SIEM – Security Information and Event Management [14]). Подобные инструменты собирают информацию из системных журналов контролируемых компьютеров и анализируют ее на предмет выявления событий безопасности, связанных с реализацией тех или иных угроз. Настроив соответствующим образом SIEM-систему, можно организовать в том числе контроль USB-подключений. Как и в случае с ALD, установка, применение и, что немаловажно, сопровождение SIEM-системы требуют существенных материальных и трудовых затрат, которые та или иная организация может понести далеко не всегда.

Таким образом, анализ публикаций в открытом доступе показывает, что существующие решения задачи контроля подключения USB-устройств к компьютерам под управлением Astra Linux SE либо носят характер общих рекомендаций, либо не предназначены для приме-

нения в ЛВС, либо требуют обязательного включения контролируемых компьютеров в домен ALD, либо связаны с установкой и настройкой SIEM-системы. Настоящая работа направлена на преодоление указанных недостатков, ее целями являются исследование методов, способов и средств построения программной системы аудита подключений USB-устройств к компьютерам под управлением Astra Linux SE в ЛВС, а также разработка подобной системы.

### **Методы, способы и средства аудита действий пользователя компьютера под управлением ОС Astra Linux SE**

Под аудитом действий пользователя понимают анализ информации о связанных с этими действиями событиях, которые происходят или происходили в ОС. Наиболее распространенным методом аудита действий пользователя в ОС Astra Linux SE является журналирование, предполагающее сбор информации о происходящих системных событиях и сохранение этой информации в специальных структурах данных, как правило, в файлах. Собранная информация анализируется, в том числе средствами SIEM, на предмет выявления событий безопасности, связанных с реализацией тех или иных угроз. Рассмотрим способы журналирования, применяемые в ЛВС контролируемых компьютеров.

Централизованное журналирование – это сбор информации о критичных с точки зрения безопасности системных событиях на одном из компьютеров (серверов) сети. Такой способ позволяет одновременно контролировать большое количество компьютеров и оперативно реагировать на события, представляющие угрозу информационной безопасности. Недостатком централизованного подхода являются накладные расходы на выделенный сервер и организацию служебного сетевого трафика для сбора информации. При децентрализованном журналировании сбор информации о событиях безопасности осуществляется на каждом из узлов ЛВС по отдельности. В этом случае накладные расходы на организацию работы центрального узла отсутствуют, но возникает необходимость анализа журналов событий отдельно на каждом компьютере. На практике это ограничивает число контролируемых машин и увеличивает время реакции системы на события безопасности. При смешанном способе журналирования контролируемые компьютеры подразделяются на подключенные к ЛВС и автономные.

Внутри ЛВС организуется централизованное журналирование, автономные компьютеры накапливают информацию о событиях безопасности отдельно и независимо друг от друга.

В ОС семейства Linux, базирующихся на дистрибутиве Debian, существует стандартная система аудита, реализуемая службой rsyslog. Служба позволяет как вести журналирование на локальной машине, так и отправлять журналы событий на специальный сервер [15]. После обновления x.7 служба rsyslog не соответствует поддерживаемым сценариям эксплуатации Astra Linux SE (см. справочный центр Astra Linux, <https://wiki.astralinux.ru/pages/viewpage.action?pageId=9011231>), и разработчики Astra Linux рекомендуют использовать в качестве средства журналирования службу syslog-ng [16]. Эта служба предоставляет достаточно широкий функционал по сравнению со стандартными средствами журналирования Linux, однако для решения задачи выявления подключений USB-устройств требует дополнительных инструментов.

Как уже отмечалось, в Astra Linux SE существует собственная подсистема аудита, реализуемая подсистемой безопасности PARSEC [2]. Информация о системных событиях записывается в файлы kernel, mlog и user.mlog, по умолчанию размещаемые в каталоге /var/log/parsec. Каждая запись файла соответствует одному зарегистрированному событию. Подсистема PARSEC позволяет более эффективно управлять регистрацией событий, непосредственно связанных с безопасностью ОС. Однако из-за сравнительно малой распространенности Astra Linux SE в мире выбор готовых инструментов для анализа собранной подсистемой PARSEC информации весьма скуден.

Среди известных средств журналирования следует также отметить службу auditd [17], предназначенную для мониторинга событий ОС Linux и их фиксации в соответствующих журналах. Этот инструмент тесно взаимодействует с ядром ОС и, наблюдая за системными вызовами, может отслеживать практически любые события, происходящие в ОС, например, связанные с чтением, записью, выполнением, изменением прав доступа для файлов. Среди недостатков auditd выделяют то, что большинство событий безопасности на уровне системных вызовов трудно отличить от нормальной работы приложения. Кроме этого, auditd может существенно замедлять работу ОС.

Современные средства аудита сохраняют информацию о подключениях USB-устройств

в системных журналах, таких как `/var/log/syslog` и `/var/log/messages`. Чтобы узнать, было ли USB-устройство подключено к компьютеру, достаточно проверить системные журналы на наличие соответствующей записи. Ручная проверка администратором содержимого журналов практически нереализуема при большом числе контролируемых компьютеров и частом подключении USB-устройств. Как уже отмечалось, доступ к разработанным специально для Astra Linux SE программам [12, 13], автоматизирующим сбор информации о подключаемых устройствах, отсутствует, поэтому авторы рассматривают аналогичные программные средства, разработанные для других версий Linux. Таким программным средством является Usbrip (<https://github.com/snovvcrash/usbrip>).

Средство Usbrip анализирует данные системных журналов, выявляет события подключения USB-устройств и направляет собранную информацию в стандартный поток вывода или в файл. В удобном для анализа виде Usbrip предоставляет исчерпывающую информацию о USB-устройствах, подключенных к контролируемому компьютеру. Собранная информация позволяет понять, какого типа устройство было подключено: носитель информации, периферийное устройство ввода-вывода или беспроводное устройство. Однако Usbrip не может быть непосредственно внедрено в среду Astra Linux SE, поскольку она не позволяет устанавливать некоторые необходимые для работы Usbrip компоненты. Для возможности работы в среде Astra Linux средство Usbrip было модифицировано путем исключения необязательных компонентов.

### **Программная система аудита подключений USB-устройств ALUMNUS**

Для достижения поставленных целей была разработана программная система аудита подключений USB-устройств, получившая название ALUMNUS (Astra Linux USB Monitoring Network Unified System). Архитектура ALUMNUS определяется следующими требованиями. Система должна обеспечивать централизованный сбор информации в ЛВС о подключениях USB-устройств к контролируемым компьютерам под управлением ОС Astra Linux SE, а также длительное хранение собранных данных и доступ к ним пользователей системы ALUMNUS. Программная система должна работать с большим количеством контролируемых компьютеров и, соответственно, орга-

низовывать очередь запросов от них к серверу. Пользователями программной системы ALUMNUS являются администраторы вычислительно-информационной инфраструктуры, в которую включена ЛВС компьютеров под управлением Astra Linux SE. Факты подключений USB-устройств к контролируемым компьютерам должны обнаруживаться ALUMNUS. Системный администратор как пользователь ALUMNUS должен иметь возможность просматривать и анализировать события, связанные с обнаруженными подключениями.

Структура системы ALUMNUS представлена на рисунке 1. На каждом контролируемом компьютере под управлением Astra Linux SE функционирует модуль сбора данных о подключениях USB-устройств, который направляет информацию об обнаруженных подключениях на специально выделенный сервер для последующей обработки. Для обеспечения отказоустойчивости системы необходимо, чтобы запросы, отправляемые с контролируемых компьютеров, гарантированно обрабатывались сервером. Для этого организована очередь сообщений-запросов, в которую осуществляют запись модули сбора информации. Запросы из очереди в порядке поступления обрабатываются функционирующим на сервере модулем проверки легитимности подключений USB-устройств. Этот модуль записывает информацию о событии в общее хранилище данных и отправляет запрос модулю обработки запросов к БД легитимных USB-устройств на наличие в этой базе информации о подключенном устройстве. Последний делает запрос к БД легитимных устройств и, используя очередь запросов, отправляет ответ серверу программной системы ALUMNUS. После того как сервер получил ответ о наличии или отсутствии информации о USB-устройстве в БД, информация о событии дополняется информацией о легитимности подключения USB-устройства к контролируемому компьютеру. Администратор вычислительно-информационной инфраструктуры в качестве пользователя системы ALUMNUS может посмотреть и проанализировать информацию о USB-подключениях через соответствующий интерфейс.

Рассмотрим предложенный технологический стек программной системы ALUMNUS, представленный на рисунке 2.

Для удобства разобьем структуру программной системы ALUMNUS на уровни (рис. 3) (синим обозначено взаимодействие с вычислительно-информационной инфраструктурой).

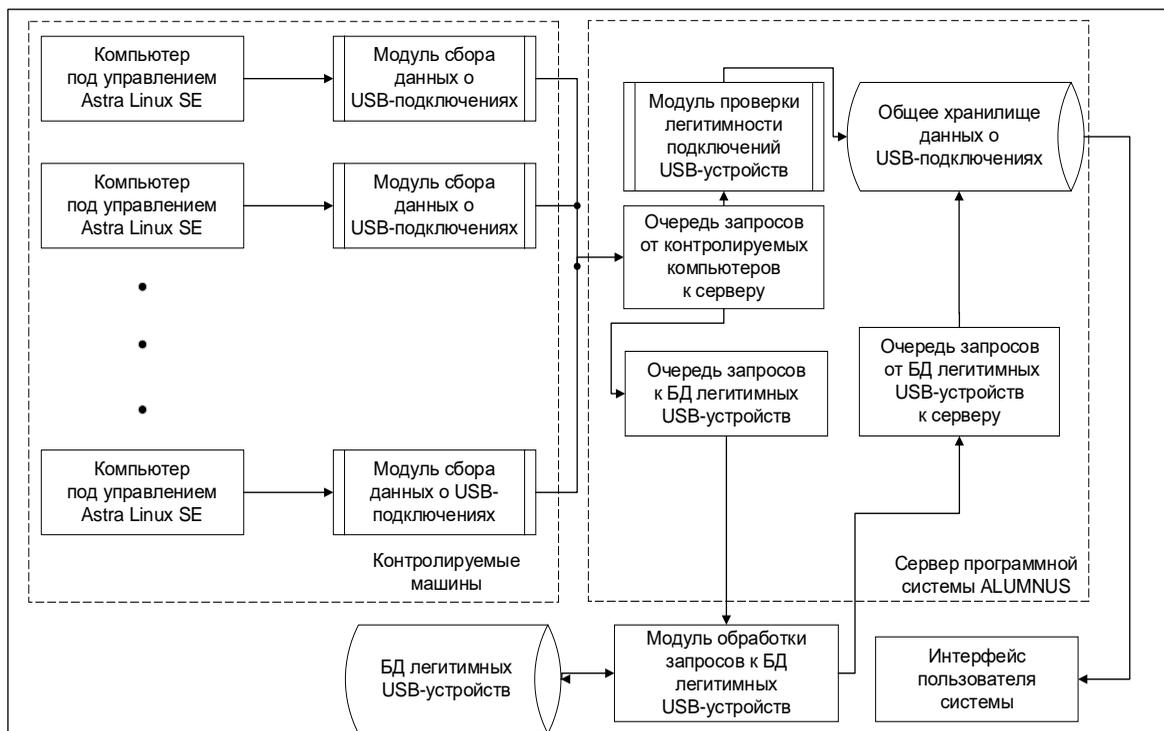


Рис. 1. Структура программной системы ALUMNUS

Fig. 1. The structure of the ALUMNUS software system

- 1-4 – разработанные модули
- 6, 7 – модули, для которых производилась настройка
- 5, 8-12 – примененные технологии

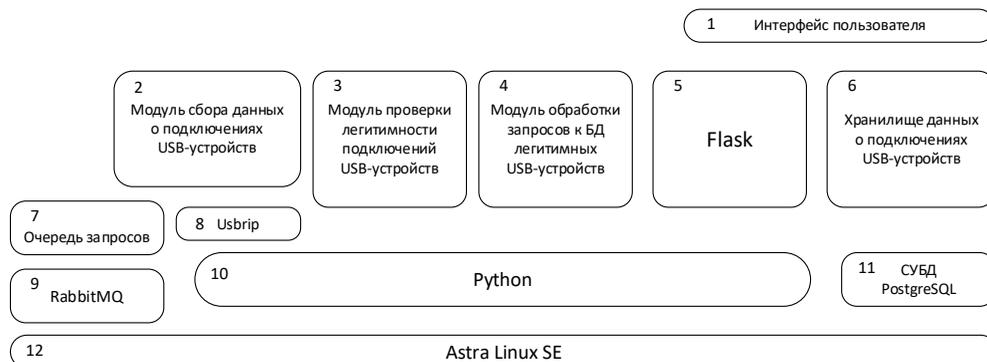


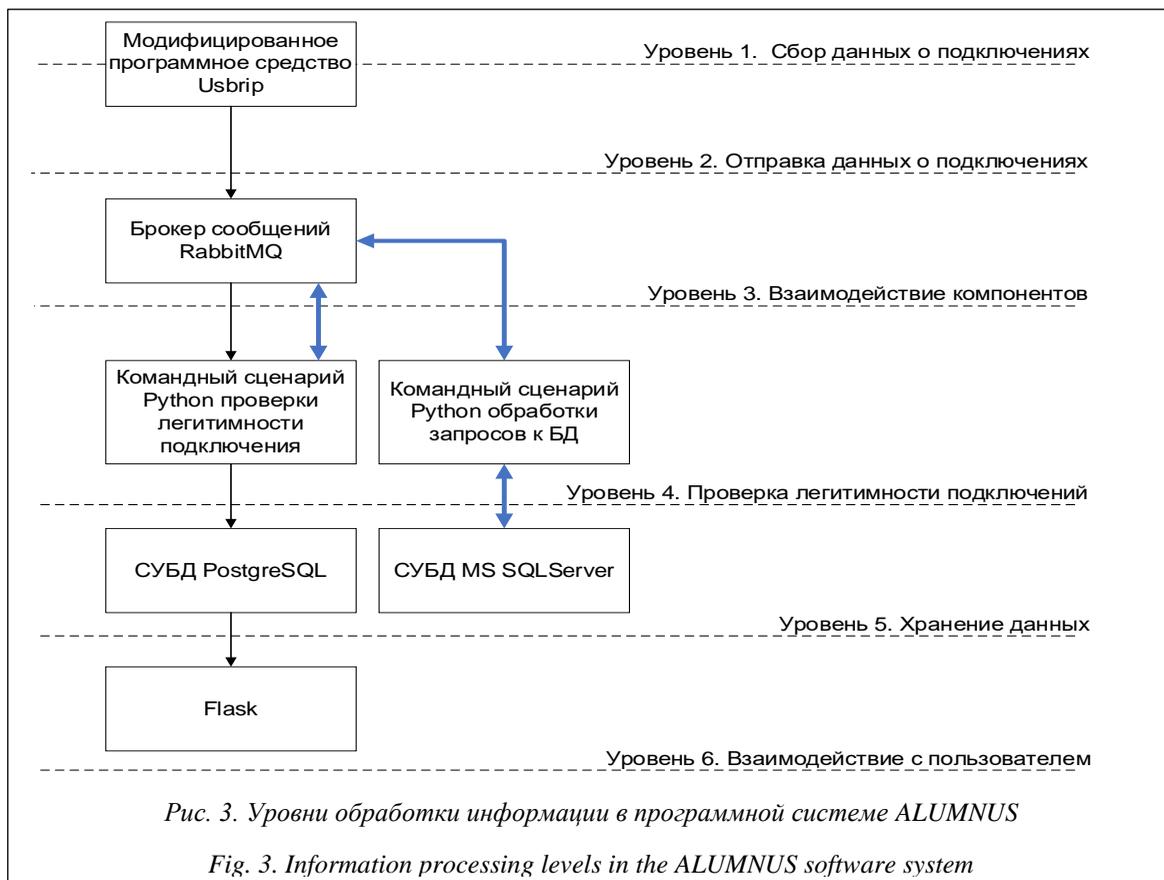
Рис. 2. Технологический стек программной системы ALUMNUS

Fig. 2. The technological stack of the ALUMNUS software system

На уровнях 1 и 2 функционирует модуль сбора данных о USB-подключениях, основанный на модифицированном для работы в Astra Linux SE программном средстве Usbrp.

Уровень 3 реализован с помощью брокера сообщений RabbitMQ [18]. Он позволяет организовать очереди запросов для обеспечения надежного взаимодействия компонентов системы (в частности, между контролируемыми

компьютерами и сервером, между сервером Astra Linux и сервером БД легитимных USB-устройств). RabbitMQ совместим с Astra Linux SE и позволяет в достаточной мере масштабировать процесс приема и обработки запросов. В случае увеличения количества контролируемых компьютеров и недостаточности мощностей сервера для своевременной обработки запросов от контролируемых машин RabbitMQ



позволит настроить несколько обработчиков на одну очередь запросов.

На уровне 4 функционируют модуль проверки легитимности и модуль обработки запросов к БД легитимных USB-устройств. Эти модули реализованы на языке программирования Python в виде командных сценариев (скриптов).

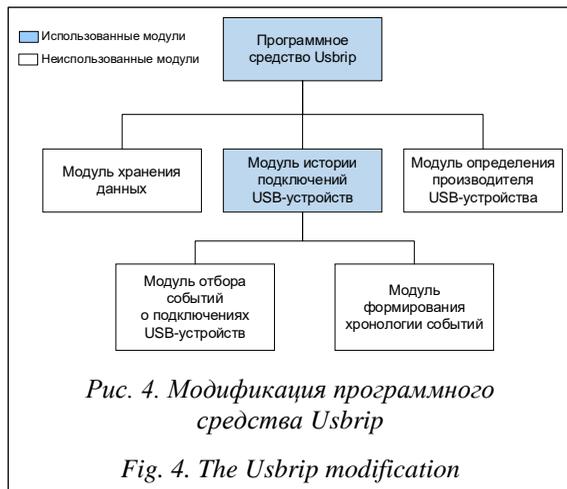
Уровень хранения данных о подключениях USB-устройств к компьютерам под управлением ОС Astra Linux SE реализован на основе СУБД PostgreSQL [19]. PostgreSQL официально поддерживается разработчиками Astra Linux SE и обеспечивает необходимый уровень как производительности, так и безопасности. PostgreSQL поддерживает широкий спектр языков программирования и платформ, что позволяет достаточно просто связать СУБД и уровень, обеспечивающий взаимодействие компонентов системы ALUMNUS.

Данные из хранилища доступны пользователю через интерфейс, реализованный с помощью фреймворка Flask [20]. Этот фреймворк представляет собой написанный на языке Python микрофреймворк для разработки веб-приложений. Flask, не требуя для своей работы специальных инструментов или библиотек, существенно упрощает разработку приложений

или расширений (плагинов). Активное сообщество разработчиков и пользователей поддерживает Flask, что позволяет минимизировать трудозатраты на разработку.

С точки зрения выявления подключений USB-устройств на контролируемых компьютерах центральным модулем системы ALUMNUS является модуль сбора данных, который, как уже упоминалось, представляет собой модифицированное программное средство Usbrip. Его модификация отображена на рисунке 4.

Из программного средства Usbrip были исключены модули, функциональность которых оказалась невостребованной в рамках системы ALUMNUS. Модуль отбора событий о подключениях USB-устройств был доработан функцией отправки событий на сервер ALUMNUS. Кроме того, добавлена возможность работы с временной меткой, необходимой для корректного определения даты и времени события. В итоге модуль сбора данных, используя модифицированный модуль Usbrip отбора событий о подключениях USB-устройств, производит на контролируемом компьютере поиск системных журналов и отбор из них информации о подключениях USB-устройств. После этого временная метка отобранного события сравнивается с



временной меткой последнего события, информация о котором была отправлена в очередь запросов на сервере. Если выявленное подключение является новым, соответствующее сообщение отправляется в очередь запросов.

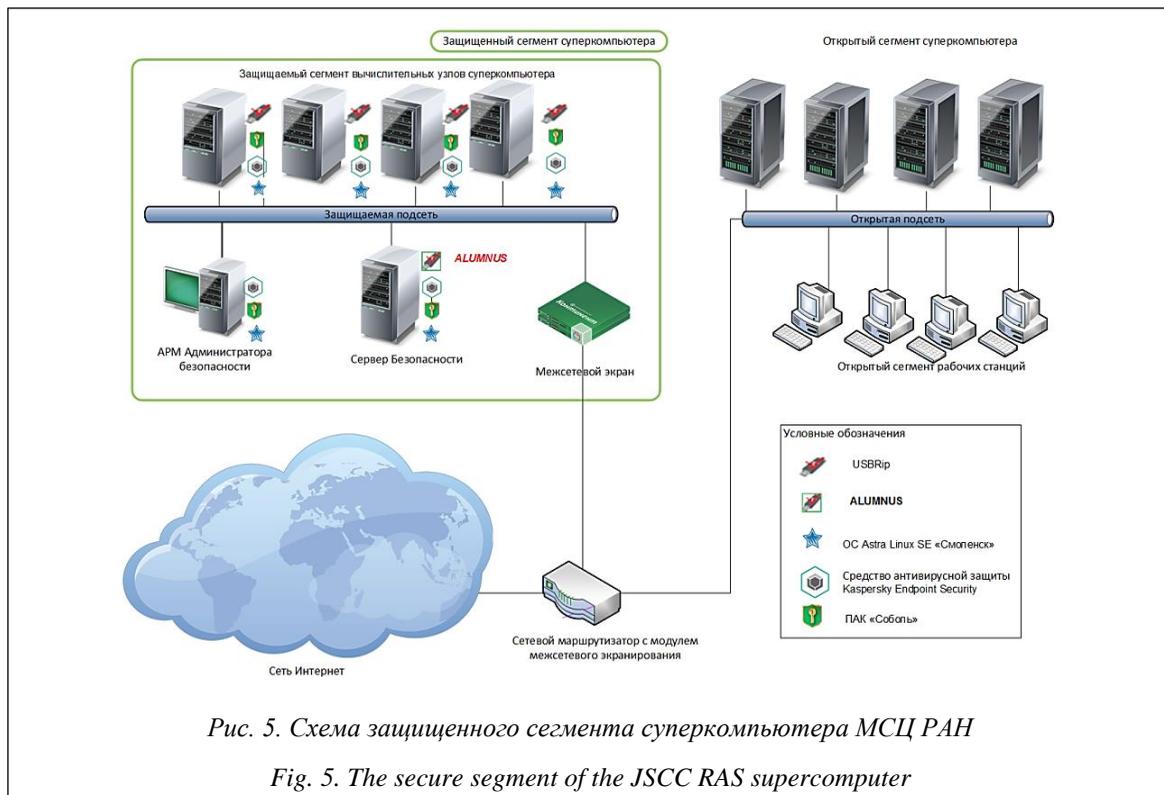
Модуль проверки легитимности подключений USB-устройств получает информацию о событии (подключение USB-устройства) из очереди запросов. Далее происходит запись информации о событии в общее хранилище данных о подключениях USB-устройств и отправка через очередь сообщений запроса модулю обработки запросов к БД легитимных USB-устройств. После получения ответа собы-

тие в общем хранилище данных помечается как представляющее угрозу информационной безопасности (если данных о USB-устройстве нет в БД) или не представляющее угрозу (если данные о USB-устройстве есть в БД).

### Опытная эксплуатация программной системы ALUMNUS

Программная система ALUMNUS была реализована в виде действующего макета, развернутого в защищенном сегменте установленного в Межведомственном суперкомпьютерном центре (МЦЦ) РАН суперкомпьютера МВС-10П ОП. Схема сегмента в составе ЛВС центра представлена на рисунке 5. Для обработки информации в защищенном режиме в составе суперкомпьютера часть вычислительных узлов выделена в отдельную подсеть. Изоляция и защита закрытой подсети обеспечиваются за счет применения межсетевых экранов как на сетевом маршрутизаторе, так и внутри самой подсети.

На каждом из вычислительных узлов установлена ОС Astra Linux SE в версии «Смоленск». Дополнительную защиту узлов составляют программно-аппаратный комплекс доверенной загрузки ОС «Соболь» [21] и средство антивирусной защиты Kaspersky Endpoint Security. В защищаемой подсети выделены АРМ



администратора безопасности и сервер безопасности, на котором функционирует и БД легитимных подключаемых USB-устройств.

Модули системы ALUMNUS сбора данных о подключениях USB-устройств были развернуты на вычислительных узлах. Сервер программной системы ALUMNUS функционирует на сервере безопасности, а интерфейс пользователя доступен через АРМ администратора безопасности. Таким образом, структура системы ALUMNUS органично легла на существующую структуру защищенного сегмента суперкомпьютера.

В ходе опытной эксплуатации модуль обработки запросов к БД легитимных устройств связывался с двумя разными СУБД – PostgreSQL и Microsoft SQL Server, продемонстрировав совместимость с обеими СУБД. Опытная эксплуатация макета программной системы ALUMNUS продемонстрировала применимость разработанного технологического стека для построения подобного рода программных решений. ALUMNUS позволяет в оперативном режиме автоматически выявлять подключения USB-устройств к контролируемым компьютерам в защищаемой подсети, выявлять факты нелегитимных подключений и извещать о таких фактах администратора безопасности. За время опытной эксплуатации произведено несколько десятков подключений USB-устройств, часть из которых были нелегитимными. Программная система ALUMNUS выявила все произведенные подключения и зафиксировала факты нелегитимных подключений в своей БД.

Перспективными направлениями развития ALUMNUS видятся интеграция ее модулей в

функционирующую на сервере безопасности SIEM-систему и расширение спектра контролируемых действий пользователя, фиксируемых модулем сбора информации.

### Заключение

Анализ доступных источников показал, что в настоящее время отсутствует комплексное программное решение, позволяющее проводить контроль подключений USB-устройств к компьютерам под управлением ОС Astra Linux SE в ЛВС. Для построения такого решения авторами предложен технологический стек компонентов, включающий, помимо ОС Astra Linux SE, систему очередей сообщений RabbitMQ, СУБД PostgreSQL, фреймворк Flask и модифицированное средство USBRip обнаружения подключений USB-устройств. На базе предложенного технологического стека разработана структура программной системы контроля USB-подключений, получившей название ALUMNUS. Структура включает модули сбора информации об USB-подключениях, функционирующие на контролируемых компьютерах, и серверную часть, агрегирующую собранную информацию. Реализованный макет программной системы ALUMNUS был развернут в защищенном сегменте суперкомпьютера МВС-10П ОП, установленного в МСЦ РАН. Опытная эксплуатация макета показала применимость предложенных технологического стека и структуры для построения программных систем оперативного контроля подключений USB-устройств к компьютерам под управлением ОС Astra Linux SE.

### Список литературы

1. Буранова М.М., Вахрушева Е.А. Операционная система Astra Linux // Информационные технологии в науке, промышленности и образовании: сб. тр. науч.-технич. конф. 2021. С. 216–222.
2. Девянин П.Н., Тележников В.Ю., Третьяков С.В. Основы безопасности операционной системы Astra Linux Special Edition. Управление доступом. М.: Горячая линия–Телеком, 2022. 148 с.
3. Кочетова И.В. Возможности использования операционной системы особого назначения «Astra Linux Special Edition 1.5» // Информационные технологии XXI века: сб. науч. тр. 2020. С. 386–390.
4. Девянин П.Н., Хорошилов А.В., Тележников В.Ю. Формирование методологии разработки безопасного системного программного обеспечения на примере операционных систем // Тр. ИСП РАН. 2021. Т. 33. № 5. С. 25–40. doi: 10.15514/ISPRAS-2021-33(5)-2.
5. Negus C. Enhancing Linux security with SELinux. In: Linux Bible, 2020, pp. 669–697. doi: 10.1002/9781119209539.ch24.
6. Ecarot T., Dussault S., Souid A., Lavoie L., Ethier J.-F. AppArmor for health data access control: Assessing risks and benefits. Proc. 7th IOTSMS, 2020, pp. 1–7. doi: 10.1109/IOTSMS52051.2020.9340206.
7. Мылицын Р.Н., Девянин П.Н. Практика построения информационных систем в защищенном исполнении на базе операционной системы Astra Linux Special Edition // Состояние и перспективы развития современной науки по направлению «Информационная безопасность»: сб. статей. 2020. С. 448–453.
8. Невров А.А., Попов Г.А. Подходы к контролю использования съемных машинных носителей информации на средствах вычислительной техники под управлением ОС Astra Linux SE // Южно-Сибирский науч. вестн. 2017. № 1. С. 45–48.

9. Nevrov A.A., Andreev I.L. Automation control of use attached media devices on computers with Astra Linux SE operation system. Proc. XXIII Int. Open Sci. Conf. MIP, 2018, pp. 245–249.
10. Лобач А.О. Интеграция СЗИ со службой каталогов Astra Linux Directory: проблемы и подходы // Вопросы защиты информации. 2021. № 3. С. 3–7. doi: 10.52190/2073-2600\_2021\_3\_3.
11. Шарафутдинова Л.В., Щерба М.В. Обнаружение несанкционированных действий в операционной системе Astra Linux // НИР-22: матер. науч.-практич. конф. 2022. С. 106–112.
12. Невров А.А., Андреев И.Л., Машошин В.Ю., Бердникова М.Р. Учет и контроль USB флеш-накопителей: Свид. о регистр. ПрЭВМ № 2018611763. Рос. Федерация, 2018.
13. Толстых А.А., Поликарпов Е.С., Лунев Ю.С., Цимбал В.Н. Программа для аудита подключений USB устройств к персональному компьютеру под управлением ОС Astra Linux: Свид. о регистр. ПрЭВМ № 2021615138. Рос. Федерация, 2021.
14. Vielberth M. Security information and event management (SIEM). In: Encyclopedia of Cryptography, Security and Privacy, 2021, pp. 1–3. doi: 10.1007/978-3-642-27739-9\_1681-1.
15. Gerhards R. Rsyslog: Going up from 40K messages per second to 250K. Proc. Linux Kongress, 2010. URL: [https://www.researchgate.net/profile/Rainer-Gerhards-2/publication/228694459\\_Rsyslog\\_going\\_up\\_from\\_40K\\_messages\\_per\\_second\\_to\\_250K/links/5830958108ae004f74c0f24f/Rsyslog-going-up-from-40K-messages-per-second-to-250K.pdf](https://www.researchgate.net/profile/Rainer-Gerhards-2/publication/228694459_Rsyslog_going_up_from_40K_messages_per_second_to_250K/links/5830958108ae004f74c0f24f/Rsyslog-going-up-from-40K-messages-per-second-to-250K.pdf) (дата обращения: 04.05.2023).
16. Chuvakin A., Schmidt K., Phillips C. Syslog-ng case study. In: Logging and Log Management, 2013, pp. 93–101. doi: 10.1016/B978-1-59-749635-3.00005-1.
17. Zam Zam M. Auditd: Rule writing for better threat detection on \*nix devices. BS Comput. Sci., 2021, pp. 1–21.
18. Christudas B. Install, configure, and run RabbitMQ cluster. In: Practical Microservices Architectural Patterns, 2019, pp. 827–847. doi: 10.1007/978-1-4842-4501-9\_21.
19. Shaik B., Vallarapu A. PostgreSQL architecture. In: Beginning PostgreSQL on the Cloud, 2018, pp. 33–61. doi: 10.1007/978-1-4842-3447-1\_2.
20. Relan K. Beginning with Flask. In: Building REST APIs with Flask, 2019, pp. 1–26. doi: 10.1007/978-1-4842-5022-8\_1.
21. Велюллаев Э.У., Гончаренко Ю.Ю., Девицына С.Н. Аппаратная защита автоматизированных рабочих мест // РТ-2022: матер. Междунар. науч.-технич. конф. 2022. № 5. С. 208.

### USB connections control in the local network of computers running under Astra Linux SE

Anton V. Baranov  
Pavel M. Korepanov  
Ignat A. Lepeshev

#### For citation

Baranov, A.V., Korepanov, P.M., Lepeshev, I.A. (2023) 'USB connections control in the local network of computers running under Astra Linux SE', *Software & Systems*, 36(3), pp. 432–441 (in Russ.). doi: 10.15827/0236-235X.142.432-441

#### Article info

Received: 19.06.2023

After revision: 28.06.2023

Accepted: 03.07.2023

**Abstract.** The constant growth of information security requirements, as well as a major trend towards import replacement in the system software, have led to the widespread use of infrastructure solutions based on the domestic Astra Linux Special Edition (SE) operating system. Astra Linux makes it possible to build secure computing systems including supercomputers for processing confidential information. At the same time, the audit of connecting USB devices to computers is one of the most important problems of ensuring information security. An analysis of the existing open access works shows the lack of turnkey solutions working in the Astra Linux SE environment. The article discusses a possible technological stack of such solution. The technological stack includes besides Astra Linux the RabbitMQ message-broker software, the Flask micro web framework, the PostgreSQL database, and the USBRip forensics tool for keeping track of USB event history on Linux machines. The proposed modular structure of the software USB connections audit system is considered. It includes the modules for collecting USB connections artifacts on controlled computers, the collected information aggregation module, and the module for checking the USB device connection permissions. The proposed structure and technological stack were implemented as a prototype of the software system called ALUMNUS. The prototype was deployed and tested in the secure segment of the MVS-10P OP supercomputer installed at the Joint Supercomputer Center of the Russian Academy of Sciences.

**Keywords:** supercomputer center, information security, Astra Linux SE, USB connection control

**Acknowledgements.** The work has been carried out at the JSCC RAS within the framework of the state assignment FNEF-2022-0016

## Reference List

1. Buranova, M.M., Vakhrusheva, E.A. (2021) 'Astra Linux Operating System', *Proc. Conf. Inform. Tech. in Sci., Industry and Education*, pp. 216–222 (in Russ.).
2. Devyanin, P.N., Telezhnikov, V.I., Tretyakov, S.V. (2022) *Fundamentals of the Operating System Astra Linux Special Edition Security. Access Control*. Moscow, 148 p. (in Russ.).
3. Kochetova, I.V. (2020) 'Possibilities of using the special-purpose operating system Astra Linux Special Edition 1.5', *Proc. Inform. Tech. of the XXI Century*, pp. 386–390 (in Russ.).
4. Devyanin, P.N., Telezhnikov, V.I., Khoroshilov, A.V. (2021) 'Building a methodology for secure system software development on the example of operating systems', *Proc. of the ISP RAS*, 33(5), pp. 25–40 (in Russ.). doi: 10.15514/ISPRAS-2021-33(5)-2.
5. Negus, C. (2020) 'Enhancing Linux security with SELinux', in *Linux Bible*, pp. 669–697. doi: 10.1002/9781119209539.ch24.
6. Ecarot, T., Dussault, S., Souid, A., Lavoie, L., Ethier, J.-F. (2020) 'AppArmor for health data access control: Assessing risks and benefits', *Proc. 7th IOTSMS*, pp. 1–7. doi: 10.1109/IOTSMS52051.2020.9340206.
7. Mylitsyn, R.N., Devyanin, P.N. (2020) 'The practice of building secure information systems based on the Astra Linux special edition operating system', *Proc. Conf. State and Prospects for the Modern Sci. Development in the Inform. Security*, pp. 448–453 (in Russ.).
8. Nevrov, A.A., Popov, G.A. (2017) 'Approaches of controlling of the use of removable computer media of computer equipment under the control of Astra Linux SE operating system', *South-Siberian Sci. Bull.*, (1), pp. 45–48 (in Russ.).
9. Nevrov, A.A., Andreev, I.L. (2018) 'Automation control of use attached media devices on computers with Astra Linux SE operation system', *Proc. XXIII Int. Open Sci. Conf. MIP*, pp. 245–249.
10. Lobach, A.O. (2021) 'Integration of information security system with the Astra Linux directory service: Problems and approaches', *Inform. Security Iss.*, (3), pp. 3–7. doi: 10.52190/2073-2600\_2021\_3\_3 (in Russ.).
11. Sharafutdinova, L.V., Scherba, M.V. (2022) 'Unauthorized actions detection in the Astra Linux operating system', *Proc. Sci-Tech. Conf. NIR-22*, pp. 106–112 (in Russ.).
12. Nevrov, A.A., Andreev, I.L., Mashoshin, V.Yu., Berdnikova, M.R. (2018) *Accounting and Control of USB Flash Drives*, Pat. RF, № 2018611763.
13. Tolstykh, A.A., Polikarpov, E.S., Lunev, Yu.S., Cimbali, V.N. (2021) *A Program for Auditing USB Device Connections to a Personal Computer Running under Astra Linux OS*, Pat. RF, № 2021615138.
14. Vielberth, M. (2021) 'Security information and event management (SIEM)', in *Encyclopedia of Cryptography, Security and Privacy*, pp. 1–3. doi: 10.1007/978-3-642-27739-9\_1681-1.
15. Gerhards, R. (2010) 'Rsyslog: Going up from 40K messages per second to 250K', *Proc. Linux Kongress*, available at: [https://www.researchgate.net/profile/Rainer-Gerhards-2/publication/228694459\\_Rsyslog\\_going\\_up\\_from\\_40K\\_messages\\_per\\_second\\_to\\_250K/links/5830958108ae004f74c0f24f/Rsyslog-going-up-from-40K-messages-per-second-to-250K.pdf](https://www.researchgate.net/profile/Rainer-Gerhards-2/publication/228694459_Rsyslog_going_up_from_40K_messages_per_second_to_250K/links/5830958108ae004f74c0f24f/Rsyslog-going-up-from-40K-messages-per-second-to-250K.pdf) (accessed May 04, 2023).
16. Chuvakin, A., Schmidt, K., Phillips, C. (2013) 'Syslog-ng case study', in *Logging and Log Management*, pp. 93–101. doi: 10.1016/B978-1-59-749635-3.00005-1.
17. Zam Zam, M. (2021) 'Audit: Rule writing for better threat detection on \*nix devices', *BS Comput. Sci.*, pp. 1–21.
18. Christudas, B. (2019) 'Install, configure, and run RabbitMQ cluster', in *Practical Microservices Architectural Patterns*, pp. 827–847. doi: 10.1007/978-1-4842-4501-9\_21.
19. Shaik, B., Vallarapu, A. (2018) 'PostgreSQL architecture', in *Beginning PostgreSQL on the Cloud*, pp. 33–61. doi: 10.1007/978-1-4842-3447-1\_2.
20. Relan, K. (2019) 'Beginning with Flask', in *Building REST APIs with Flask*, pp. 1–26. doi: 10.1007/978-1-4842-5022-8\_1.
21. Velullaev, E.U., Goncharenko, J.J., Devitsyna, S.N. (2022) 'Hardware protection of automated jobs', *Proc. Int. Sci.-Tech. Conf. RT-2022*, (5), pp. 208 (in Russ.).

## Авторы

**Баранов Антон Викторович**<sup>1</sup>, к.т.н., доцент,  
ведущий научный сотрудник,  
antbar@mail.ru, abaranov@jscs.ru  
**Корепанов Павел Михайлович**<sup>1</sup>, начальник сектора  
информационной безопасности, kpm@jscs.ru  
**Лепешев Игнат Анатольевич**<sup>1</sup>,  
стажер-исследователь, rin.l@yandex.ru

## Authors

**Anton V. Baranov**<sup>1</sup>, Ph.D. (Engineering),  
Associate Professor, Leading Researcher,  
antbar@mail.ru, abaranov@jscs.ru  
**Pavel M. Korepanov**<sup>1</sup>, Head of information  
security Sector, kpm@jscs.ru  
**Ignat A. Lepeshev**<sup>1</sup>, intern Researcher,  
rin.l@yandex.ru

<sup>1</sup> Межведомственный суперкомпьютерный  
центр РАН, г. Москва, 119991, Россия

<sup>1</sup> Joint Supercomputer Center of RAS,  
Moscow, 119991, Russian Federation

## Организация адаптивной маршрутизации данных в электроэнергетических комплексах с использованием онтологических нечетких классификаторов

А.С. Федулов  
А.И. Лазарев

### Ссылка для цитирования

Федулов А.С., Лазарев А.И. Организация адаптивной маршрутизации данных в электроэнергетических комплексах с использованием онтологических нечетких классификаторов // Программные продукты и системы. 2023. Т. 36. № 3. С. 442–450. doi: 10.15827/0236-235X.142.442-450

### Информация о статье

Поступила в редакцию: 29.05.2023

После доработки: 04.07.2023

Принята к публикации: 05.07.2023

**Аннотация.** В работе рассматриваются теоретические аспекты применения методов машинного обучения, в частности, адаптация глубоких моделей к управлению сетевыми топологиями TCP/IP электроэнергетических комплексов. Предметом исследования является подход к организации централизованного управления сегментами сети в рассматриваемой сфере. Изучение процессов взаимодействия субъектов электроэнергетических подразделений на основе разработанных онтологических моделей позволило выявить основные свойства полиформатных данных, которые могут быть уязвимыми при эксплуатации. Практическая значимость исследования заключается в создании многомодульной структуры отслеживания, классификации и прогнозирования изменений в потребляемом трафике, за счет которой возможно повышение эффективности функционирования сложных сетевых корпоративных структур. Проведено тестирование существующих алгоритмов получения хеш-функций. Его результаты позволили сделать вывод о целесообразности применения базового алгоритма шифрования BLAKE3 в качестве основного механизма верификации подлинности клиентов в сравнении с алгоритмами SHA-384, SHA-512, SHA-224, MD5. Показана реализация алгоритма нечеткого посимвольного сравнения в качестве модуля принятия решений, что подтверждает актуальность предлагаемого подхода при работе с нечеткими структурами данных. В качестве основного решения указанных проблем предложен подход к гибкому управлению сегментом электроэнергетических установок, представляемых комплексом генерирующих, электросетевых, энергосбытовых и других компаний. Основным результатом предлагаемого решения является централизованный анализ возможных изменений с учетом адаптации к сетевым нагрузкам на основе выделенных онтологических переменных. При реализации данного подхода возможна совместимость с существующими аппаратными сетевыми устройствами за счет уникальной архитектуры построенной топологии.

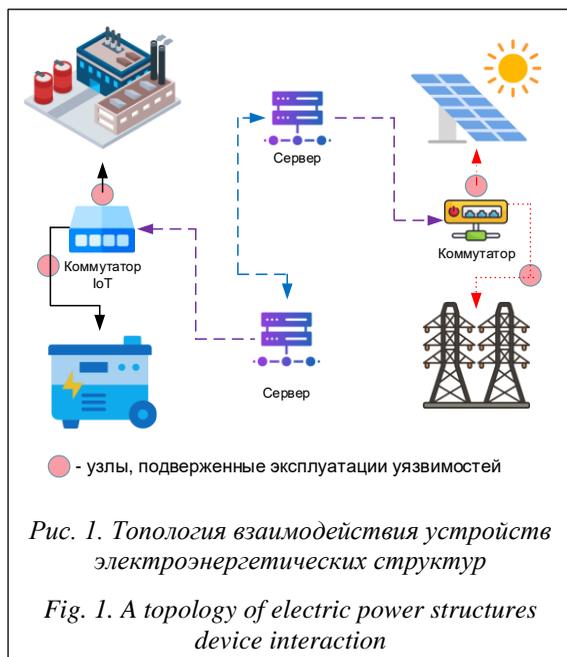
**Ключевые слова:** электроэнергетические комплексы, онтологические модели, классификация данных, принятие решений, нечеткая логика, обеспечение безопасности, глубокое обучение

**Благодарности.** Работа выполнена в рамках государственного задания, проект № FSWF-2023-0012

Совокупность множества управляемых электросетевых, энергосбытовых компаний, включая энергосистемы и подразделения поддержки коммерческой инфраструктуры российского рынка, представляет собой электроэнергетический комплекс, осуществляющий бесперебойное снабжение потребителей необходимыми средствами [1, 2]. Развитие данной области сопровождается активным внедрением IT-технологий в процессы автоматизации деятельности, отслеживания неполадок и проведения критически важных испытаний. Идентичные структуры и процессы присущи также электромеханическим и теплотехнологическим системам.

Развитие подразделений в указанных инфраструктурах поддерживается внедрением сетевых структур на базе протоколов TCP/IP, включая функциональные возможности резервирования важных данных, поддержки отказоустойчивости, оперативного изменения состава сегментов предприятия. Существенная

часть как аппаратных (cisco, microtik), так и программных (pfSense, NethServer, ClearOS) решений поддерживают интеллектуальное управление трафиком клиентов с учетом корректной настройки маршрутизации до конечного клиента [3, 4]. Вместе с тем существует ряд проблем, основной из которых является правильность настройки сетевого оборудования, включая возможное взаимодействие IoT-оборудования (Internet of Things) с аппаратными техническими комплексами (рис. 1). Исследования в области оптимизации адресации TCP/IP проводятся многими учеными. Так, в статье [5] авторы предлагают решить проблему недостаточности выделяемого пула в маршрутизаторах малого офиса за счет изменения операционной системы OpenWRT. Данный подход позволяет частично решить указанную проблему оптимизированной доставки трафика клиентам, однако остается актуальной проблема адаптации к другим операционным системам при использовании альтернативных



поставщиков сетевого оборудования. В публикации [6] описан процесс разработки протокола туннелирования с использованием IPv6-адресации и перезаписи полей заголовков для идентификации клиента, представлены перспективные направления в развитии методов безопасной передачи данных. Предлагаемое решение обладает новизной в области прогнозирования изменений в сетевом трафике. В то же время возможности использования обученных моделей не предполагают дополнительных затрат на усовершенствование аппаратной составляющей электроэнергетических комплексов.

Представленная на рисунке 1 топология взаимодействия затрагивает два магистральных узла, а также ряд устройств, напрямую взаимодействующих с управляющими электроэнергетическими установками. Нарушение корректной конфигурации в одном из узлов топологии (выделены круговыми указателями) может повлечь за собой ряд проблем. Среди них можно выделить перегрузку трафика на каком-либо узле сети, а также нарушение конфиденциальности из-за использования устаревших версий протоколов и ПО (включая полнофункциональный доступ к узлу сети), что подтверждает актуальность исследования.

Для решения указанных проблем предлагается система управления сетевыми топологиями ТСР/Р в электроэнергетических комплексах с использованием данных онтологической модели для автоматизации процессов принятия решений по оптимизации работы сегментов сети.

### Исследование основных параметров эксплуатации уязвимостей на базе онтологических моделей

Рассматривая большую часть уязвимостей и полезных нагрузок на ресурсах OffSec (<https://www.exploit-db.com>), предполагающих прямое или косвенное воздействие на программную платформу информационных структур, следует обратить внимание на таргетированные атаки на протоколы доставки данных. Различия в данной базе определяются конкретным портом приложения: на текущий момент насчитывается более 300 портов приложений и служб, для которых существуют уязвимости различных уровней.

Для решения потенциальных проблем с обеспечением безопасности компаний, являющихся поставщиками сервисов, на постоянной основе предлагают клиентам возможность своевременного обновления интегрируемого ПО. Вместе с тем часть специфических программных средств могут напрямую зависеть от аппаратной составляющей, изменение которой в большинстве случаев нерентабельно. Альтернативной проблемой является некорректность конфигурации одного или нескольких узлов сетевых топологий, приводящая к нарушению конфиденциальности – зачастую организации на территории РФ используют технологии на стадии альфа-релиза.

В качестве основного подхода к реализации оптимального управления электроэнергетическими структурами предлагается онтологическая модель, изображенная на рисунке 2. Она представляет собой совокупный набор большинства полиформатных данных, обрабатываемых на электроэнергетических комплексах. Как можно увидеть из этой модели, на текущий момент основными сервисами для обработки данных являются приложения, работающие на портах 20, 21, 25, 80, 443, 3306, 3389, 110 [7, 8]. Указанные порты в большинстве случаев являются стандартными для доступа к интерфейсу управления, что также нарушает политику информационной безопасности.

Условное подразделение, представленное на рисунке 2, указывает на множественные точки уязвимостей, например, внедрение вредоносного исполняемого кода в передаваемый файл, изменение и перехват аудио- и видеопотока, использование слабых ключей шифрования к удаленному терминалу, а также обеспечение удаленного управления через устаревшие версии протоколов RDP/VNC.

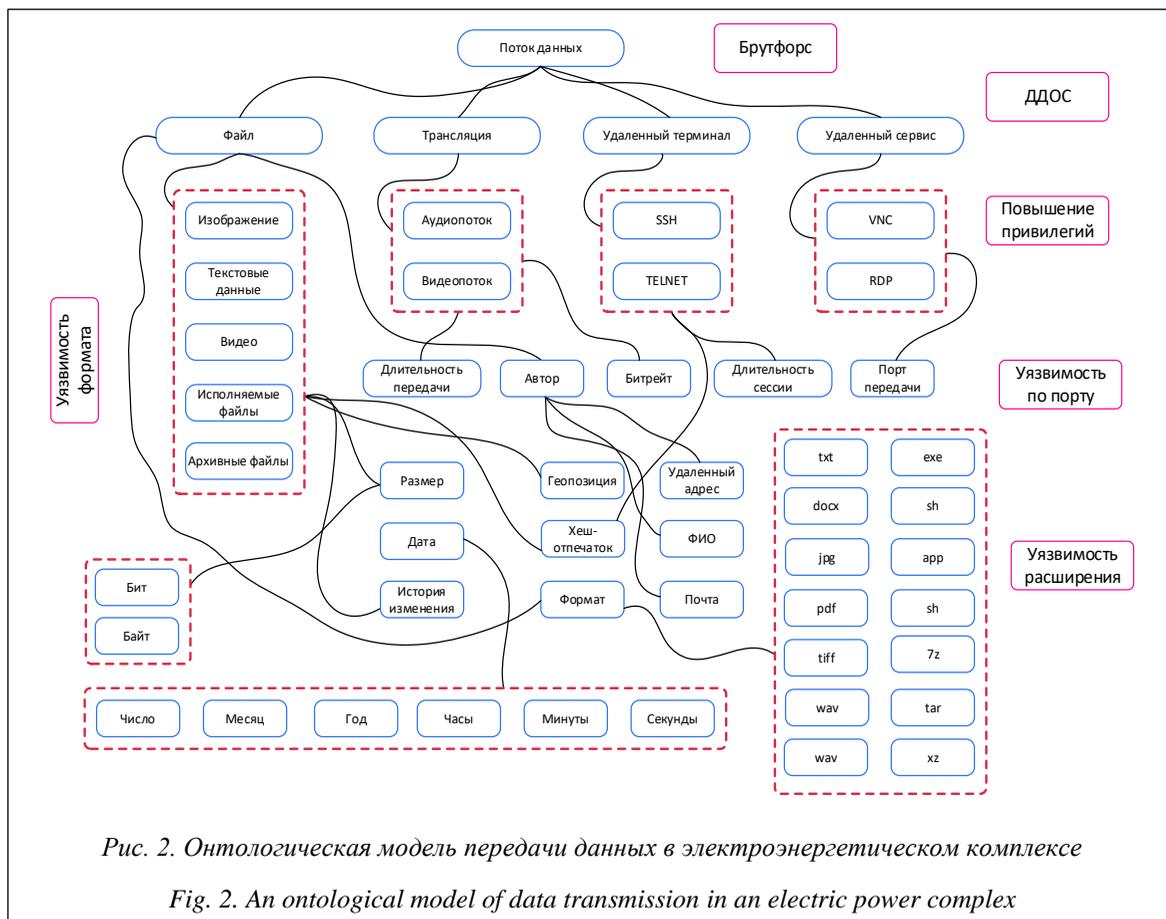


Рис. 2. Онтологическая модель передачи данных в электроэнергетическом комплексе

Fig. 2. An ontological model of data transmission in an electric power complex

Использование данной онтологической модели в процессе разработки программного алгоритма также позволяет выявить основные изменяемые третьими лицами показатели для маскировки изменений в оригинальной сигнатуре, такие как хеш файла, информация об авторе, геолокация. В качестве основных примеров реализации уязвимостей можно привести CVE-2019-0708, CVE-2019-1935, CVE-2020-0688, некоторые из них не требуют дополнительного вмешательства со стороны атакуемого лица.

Рассматривая структуру обмена данными между пользователями, можно выделить основные отличия передаваемых данных (рис. 3). Выделяемые классификации трафика позволяют в последующем оптимизировать работу отдельных подсетей для обеспечения минимальной задержки взаимодействия с внешним сервером.

Результат рассмотрения представленных моделей, в частности, упрощенной модели классификации трафика, позволяет в последующей реализации подхода оптимизации сети электроэнергетических образований выделить основные факторы (параметры), используе-

мые в дальнейшем как главные прогнозируемые показатели в системе принятия решений по управлению сегментированными участками.

### Разработка многомодульной системы управления электроэнергетическими структурами

Немаловажной составляющей при разработке интеллектуального подхода к управлению сетевыми структурами электроэнергетических комплексов являются алгоритмы определения ключевых устройств в сегменте сети, а также выявления возможных мест эксплуатации уязвимостей в соответствии с представленной онтологической моделью классификации данных. В предлагаемом алгоритме интеллектуального управления (рис. 4) основным является первичный анализ доступных устройств в сегменте сети с последующим выявлением доступных портов (глубокое сканирование) для классификации трафика и обучения глубокой модели.

Как видно из представленного алгоритма, процесс оптимизации нацелен на внесение

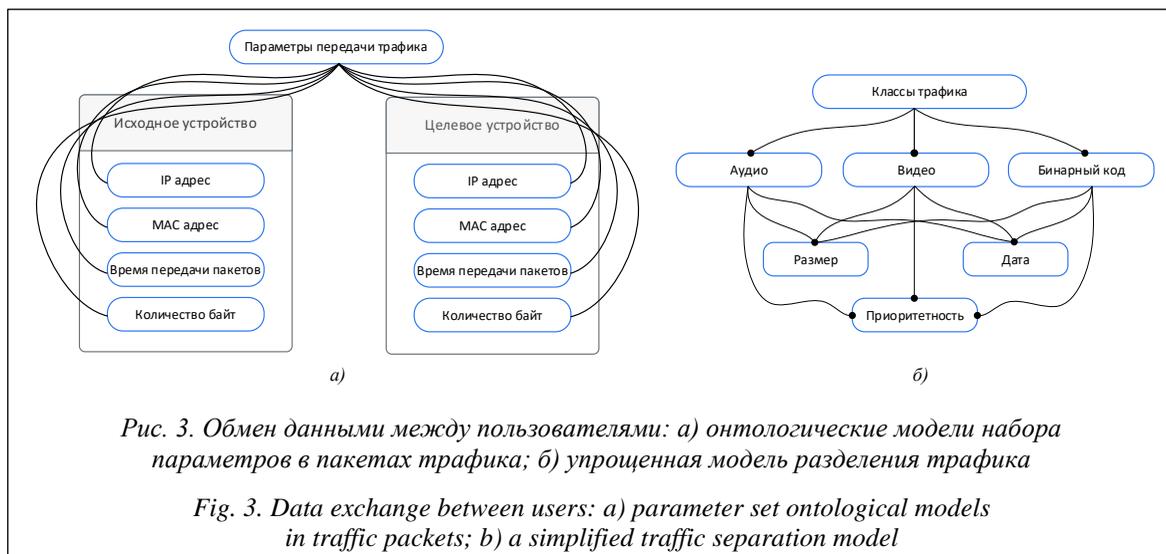


Рис. 3. Обмен данными между пользователями: а) онтологические модели набора параметров в пакетах трафика; б) упрощенная модель разделения трафика

Fig. 3. Data exchange between users: a) parameter set ontological models in traffic packets; b) a simplified traffic separation model

изменений в реальном времени с учетом выявленных отклонений в соответствии с первоначальной сетевой топологией TCP/IP. Этапы сопоставления предиктивного трафика с реальными данными, изменения маршрутизации на основе расходуемого трафика, принятия решений по изменению топологии в данном алгоритме обособлены из-за изменчивости структуры электроэнергетических систем, то есть добавления или удаления различных сегментов.

С учетом выделенной топологии взаимодействия устройств и алгоритма интеллектуального управления электроэнергетическими структурами была разработана углубленная топология взаимодействия устройств (<http://www.swsys.ru/uploaded/image/2023-3/2023-3-dop/15.jpg>).

В качестве основной программной составляющей предполагается наличие базового DHCP сервера разделения сегментации (dhcpd), виртуальных компьютерных сетей (VLAN), а также нескольких управляемых точек виртуальных внешних сетей (VPN), инициализируемых сервером WireGuard/OVPN [9]. Выделяя возможные подразделения сегментов сети, следует отметить, что соединение выделяемых подразделений осуществляется через виртуальный сегмент с использованием частных виртуальных сетей.

Организация процесса классификации трафика в данном решении основана на применении дополнительных средств анализа выходных данных фреймворка NFStream [10]. Используемый пакет задействует глубокий анализ пакетов за счет выделения множества параметров начальной и конечной точек IP-адресации, задержки отправки и получения данных,

количества выходной и входной информации (в байтах), а также размера пакетов и нагрузки сети. В качестве основного средства для прогнозирования изменений в трафике используется библиотека numpy с последующей записью данных в датафрейм (библиотека pandas).

Процесс прогнозирования и сравнения изменений в трафике выглядит следующим образом:

```
# import lib
import numpy
import datetime
import pandas as pd

class ModelPrediction(NFPlugin):
    # Flow initialization method
    def on_initial(self, packet, flow):
        flow.udps.model_prediction = 0
    # Method called when updating the stream
    def on_update(self, flow):
        to_predict = numpy.array([flow.bidirectional_packets,
                                   flow.bidirectional_bytes]).reshape((1,-1))
        flow.udps.model_prediction = self.my_model.predict(to_predict)

ml_streamer = NFStreamer(source="en0",
                          udps=ModelPrediction(my_model=model))
# Writing data to pandas
data = ml_streamer.to_csv(path=None,
                           columns_to_anonymize=[],
                           flows_per_file=0, rotate_files=0)
```

Очевидно, что в модели прогнозирования используется процесс захвата потока из адаптера en0 с последующим сохранением в csv-файл – в качестве основных параметров для прогнозирования используются bidirectional packets, bidirectional\_bytes, составляющие ак-

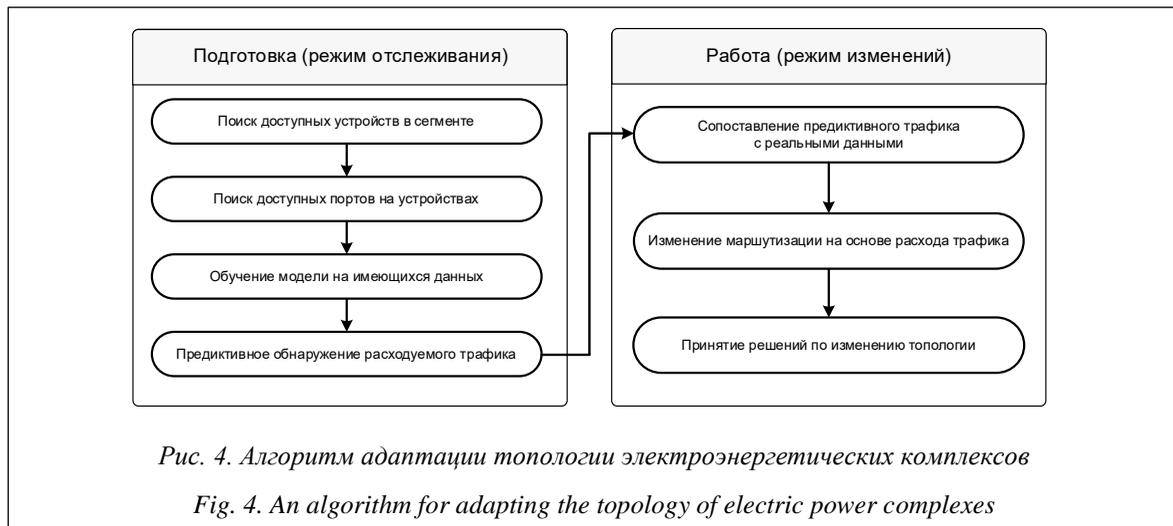


Рис. 4. Алгоритм адаптации топологии электроэнергетических комплексов  
 Fig. 4. An algorithm for adapting the topology of electric power complexes

кумулятор потоков и количество байтов для двунаправленных пакетов.

В качестве дополнительных средств для прогнозирования изменений в возможных превышениях трафика предлагается использование глубокой модели на основе выходных данных pcap-файла как входных данных для обучения сети [11]. Основу предлагаемой нейронной сети составляет модель Long-Short Term Memory (LSTM), предоставляющая возможность долговременного хранения зависимостей [12].

Продемонстрируем работу слоя фильтра забывания:

$$g_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f),$$

входной слой:

$$l_t = g_t c_{t-1} + \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \times \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c),$$

а также выходной:

$$h_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_o \tanh(c_t)),$$

где  $W, b$  – обучаемые параметры;  $\sigma, \tanh$  – функции активации [13].

Программная реализация модели LSTM основана на библиотеке Keras. Модель включает слои *Dense*, *LSTM*, *SimpleRNN*, основное количество нейронов для которых – 1, 4, 3 соответственно [14]. Главной функцией оптимизатора является adaptive moment estimation (adam) с целевым параметром потерь – расчет среднеквадратичной ошибки:

$$R = \sqrt{\frac{\sum_{i=1}^N \|y(i) - \hat{y}(i)\|^2}{N}},$$

где  $y(i)$  –  $i$ -я итерация изменений;  $N$  – количество данных;  $\hat{y}(i)$  – прогноз.

Таким образом, при использовании комбинации фреймворка NStream и сторонней

LSTM-модели возможно получение как данных по классификации трафика, так и прогнозируемых значений по расходуемому трафику в различных периодах.

### Разработка модуля обеспечения безопасного функционирования электроэнергетической ТСП/IP-топологии

Немаловажной составляющей в процессе ТСП/IP-взаимодействия является шифрование данных с использованием как протоколов защиты трафика, например, SSL, VPN, так и средств шифрования данных на стороне клиента. Существующие средства туннелирования трафика в большинстве случаев основаны на первичной установке клиент-серверных корневых сертификатов, обмене двухсторонними ключами (подключение как пира), а также применении пары логин–пароль. Альтернативная поддержка обеспечения безопасности также возможна путем сертификации с использованием SSL-шифрования в случаях взаимодействия с веб-приложениями, однако актуальным остается вопрос обеспечения безопасности при проведении таргетированных атак на отдельные службы и сервисы. Как можно увидеть из представленной на рисунке 1 онтологии, непреднамеренное использование хотя бы одного из уязвимых сервисов может привести к утечке данных, следующей за несанкционированным доступом.

Для решения проблемы предлагается использование средств хеширования на основе алгоритма BLAKE3 с последующим комбинированием TOTP-алгоритма обновления ключей шифрования, реализующих совокупный

алгоритм непрерывной аутентификации субъектов. Алгоритм шифрования BLAKE3 является хорошей альтернативой существующим хеш-функциям, предоставляя возможность быстрого получения зашифрованного сообщения наряду с надежностью, идентичной SHA-3. В качестве основного тестирования производительности была проведена проверка вычисления хеш-суммы двоичного файла размером 1 Гб на процессоре MAC M1 и построен график (<http://www.swsys.ru/uploaded/image/2023-3/2023-3-dop/16.jpg>).

Согласно результатам последовательного тестирования указанных функций можно заключить, что для расчета хеш-функции алгоритму SHA-224 требуется 3,708 сек., в то время как алгоритму BLAKE3 – 0,18 сек.

С учетом того, что при TOTP-шифровании важным является процесс обновления функции за определенный фиксированный интервал, наиболее целесообразно применение алгоритма шифрования BLAKE3. Его алгоритмическая составляющая образована за счет комбинирования средств предыдущего хеш-шифра (BLAKE2) и утилиты Bao, что в совокупности снимает ограниченное ветвление (рис. 5). На основании рисунка можно заключить, что архитектура ограничивается блоками по 1 024 байта, при которых возможно переполнение с последующим образованием двух родительских узлов, причем переполнение возможно начиная с предоставления 1 байта для следующего фрагмента.

Для организации процесса динамического обновления ключа предлагается использовать TOTP-генерацию зависимого ключа, образуемого по формуле

$$\text{BLAKE3}(\text{VALUE}) = \text{CLIENT}_{id},$$

$$\text{VALUE}(\text{ID}) = \left[ \frac{T_1 - T_0}{T} \right] \cdot \text{DELAY},$$

где VALUE – цифровой временный отпечаток; BLAKE3 – общий идентификатор для пары устройств, образуемый от VALUE и хеш-функции;  $T_1$  – текущее системное время;  $T_0$  – статичный параметр времени; DELAY – время действия отпечатка;  $\text{CLIENT}_{id}$  – идентификатор пары.

Программная реализация указанного метода осуществляется с учетом передаваемого потока информации: предлагаемое решение за счет своей гибкости позволяет вычислять цифровые отпечатки для определенных данных, действительных в течение  $n$ -го времени.

### Разработка модуля принятия решений по изменению структуры распределения трафика ТСП/ІР

Немаловажным компонентом в реализации указанного подхода является система принятия решений по изменению структуры в отдельных частях топологии сети ТСП/ІР. В большинстве случаев принятие автоматизированных решений нацелено на работу с достаточными данными либо с алгоритмами машинного обучения на нечеткой основе. С учетом ранее предложенных методов классификации трафика, прогнозирования скорости загрузки и отдачи через WAN-канал предлагается использовать автоматизированные функции вычисления возможных совпадений при линейном сравнении с оригинальными выходными данными.

Как сказано ранее, фреймворк NFStream обладает возможностью прогнозирования классов трафика, в то время как реализованный подход к прогнозированию потребления трафика на основе LSTM-моделей позволяет получать возможные значения по потреблению. Использование выходных данных указанных модулей позволяет автоматизировать процессы принятия решений по снижению потребления. С учетом того, что большая часть прогнозных данных представлена числами с плавающей точкой, предлагается использовать нечеткое сопоставление данных, нацеленное на идентификацию похожих, но не идентичных данных.

Для сопоставления данных можно использовать множество алгоритмов, включая расстояние Хэмминга, Дамерау–Левенштейна, Левенштейна. Реализация последнего из приведенных алгоритмов сводится к измерению расстояния между двумя последовательностями, цель которого – вычисление количества односимвольных правок, по формуле

$$\text{lev}(a, b) = \begin{cases} |a|, & \text{if } |b| = 0, \\ |b|, & \text{if } |a| = 0, \\ \text{lev}(\text{tail}(a), \text{tail}(b)), & \text{if } a[0] = b[0], \\ 1 + \min \begin{cases} \text{lev}(\text{tail}(a), b), \\ \text{lev}(a, \text{tail}(b)) \\ \text{lev}(\text{tail}(a), \text{tail}(b)), \end{cases} & \text{otherwise} \end{cases}$$

где  $\text{tail}(x)$  – символьная строка, исключаяющая 1-й символ как  $x$ ;  $x[n]$  – символ  $n$  строки  $x$ , начинающийся с нулевой позиции.

С учетом указанной формулы возможно вычисление процентного совпадения по прогно-

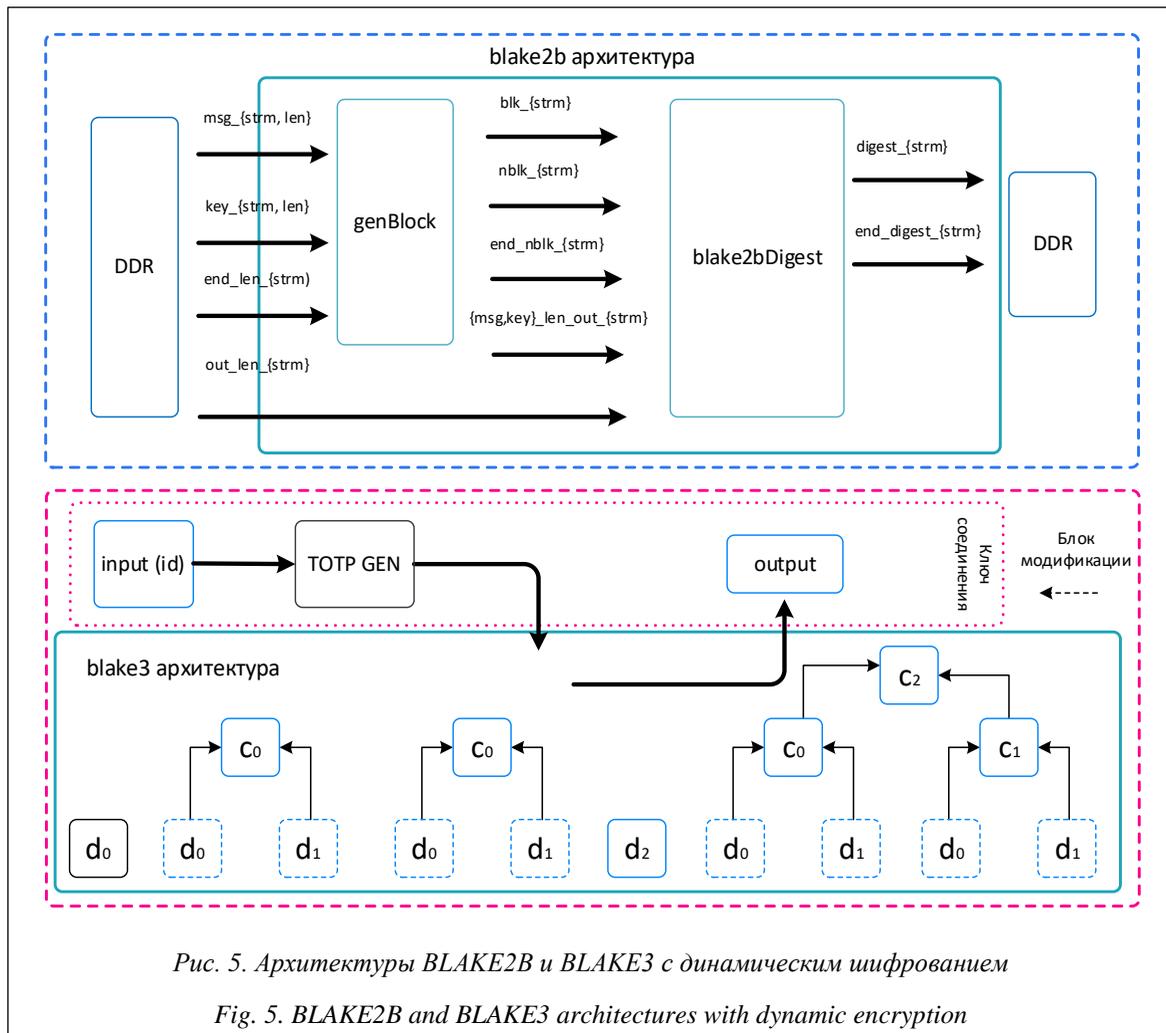


Рис. 5. Архитектуры BLAKE2B и BLAKE3 с динамическим шифрованием

Fig. 5. BLAKE2B and BLAKE3 architectures with dynamic encryption

зируемым изменениям – например, процент совпадения по 1.023 и 1.029 Mb/s (расход трафика) позволяет перераспределить приоритеты классификации для снижения нагрузки на сеть. Программная реализация указанного метода произведена с использованием библиотеки Fuzzy на языке Python, где реализованы функции как частичного, так и посимвольного сравнения [15].

### Заключение

Рассмотрение инфраструктуры электросетевых компаний как единой экосистемы взаимодействия устройств в рамках TCP/IP-адресации позволило выявить существующие проблемы, такие как недостаточная эффективность при наличии множественных удаленных управляющих узлов, а также использование устаревших средств прикладного ПО и протоколов передачи данных. Для решения данных проблем в статье предложены онтологические модели классифи-

кации трафика, которые позволили разработать функциональные модули классификации и прогнозирования трафика между как локальными, так и центральными узлами сети. Для случаев несанкционированного доступа и предотвращения возможных утечек данных предлагается комбинирование средств шифрования на основе алгоритма BLAKE3 и TOTP-авторизации. Разработанный алгоритм динамического шифрования не уступает существующим алгоритмам шифрования по скорости вычисления хеш-функции и обладает поддержкой уникальности при обработке полиформатного трафика.

Существенным программным модулем в предлагаемом решении является алгоритм автоматизированного принятия решений по управлению изменениями в структуре электроэнергетических систем на основе расстояния Левенштейна. Посимвольное сравнение позволило вычислять отклонения между исходными и прогнозируемыми значениями для изменения сетевой структуры.

## Список литературы

1. Васильев Д.А. Перекрестное субсидирование в электроэнергетике: текущее состояние и векторы решения проблем // Современная конкуренция. 2021. Т. 15. № 3. С. 17–30. doi: 10.37791/2687-0649-2021-15-3-17-30.
2. Склюев А.М., Хабаров В.И., Мусатова И.В., Попова О.В. Организационно-управленческие инновации в электронной промышленности России: современные тренды // Современная конкуренция. 2022. Т. 16. № 6. С. 103–116. doi: 10.37791/2687-0657-2022-16-6-103-116.
3. Zientara D. Mastering pfSense: Manage, secure, and monitor your on-premise and cloud network with pfSense 2.4. Birmingham, Packt Publ., 2018, 450 p.
4. Muthukumar M., Senthilkumar P., Jawahar M. Firewall scheduling and routing using pfSense. In: AISC, 2019, vol. 1172, pp. 749–757. doi: 10.1007/978-981-15-5566-4\_67.
5. Syafei W.A., Soetrisno Y.A.A., Prasetijo A.B. Simple smart algorithm for flexibility of dynamic allocation in DHCP server for SOHO wireless router. Proc. Int. Conf. CENIM, 2020, pp. 321–325. doi: 10.1109/CENIM51130.2020.9297852.
6. Yi B., Congxiao B., Xing L. FlowLAN: A non-tunneling distributed virtual network based on IPv6. Proc. IEEE ITNEC, 2016, pp. 229–234. doi: 10.1109/ITNEC.2016.7560355.
7. Singh G.D. The Ultimate Kali Linux Book: Perform Advanced Penetration Testing Using Nmap, Metasploit, Aircrackng, and Empire. Birmingham, Packt Publ., 2022, 742 p.
8. Бобряков А.В., Борисов В.В., Мисник А.Е., Прокопенко С.А. Моделирование и проектирование информационно-аналитических производственных процессов на основе нейронечетких темпоральных сетей Петри // Прикладная информатика. 2022. Т. 17. № 2. С. 65–78. doi: 10.37791/2687-0649-2022-17-2-65-78.
9. Yi T., Chen X., Zhu Y., Ge W., Han G. Review on the application of deep learning in network attack detection. J. of Network and Comput. Applicat., 2023, vol. 212, art. 103580. doi: 10.1016/j.jnca.2022.103580.
10. Aouini Z., Pekar A. NFStream: A flexible network data analysis framework. Comput. Networks, 2021, vol. 204, art. 108719. doi: 10.1016/j.comnet.2021.108719.
11. Пучков А.Ю., Дли М.И., Прохимнов Н.Н., Шутова Д.Ю. Многоуровневые алгоритмы оценки и принятия решений по оптимальному управлению комплексной системой переработки мелкодисперсного рудного сырья // Прикладная информатика. 2022. Т. 17. № 6. С. 102–121. doi: 10.37791/2687-0649-2022-17-6-102-121.
12. Мешалкин В.П., Дли М.И., Пучков А.Ю., Лобанева Е.И. Предварительная оценка прагматической ценности информации в задаче классификации на основе глубоких нейронных сетей // Прикладная информатика. 2021. Т. 16. № 3. С. 9–20. doi: 10.37791/2687-0649-2021-16-3-9-20.
13. Дли М.И., Синяевский Ю.В., Рысина Е.И., Василькова М.А. Метод классификации перемешивающих устройств с использованием глубоких нейронных сетей с расширенным рецептивным полем // Прикладная информатика. 2022. Т. 17. № 5. С. 51–61. doi: 10.37791/2687-0649-2022-17-5-51-61.
14. Дли М.И., Бульгина О.В., Соколов А.М. Рубрицирование текстовой информации на основе голосования интеллектуальных классификаторов // Прикладная информатика. 2020. Т. 15. № 5. С. 29–36. doi: 10.37791/2687-0649-2020-15-5-29-36.
15. Дли М.И., Власова Е.А., Соколов А.М., Моргунова Э.В. Создание цифрового двойника химико-технологической системы с использованием языка Python // Прикладная информатика. 2021. Т. 16. № 1. С. 22–31. doi: 10.37791/2687-0649-2021-16-1-22-31.

### Organization of adaptive data routing in electric power complexes using ontological fuzzy classifiers

Alexander S. Fedulov  
Alexey I. Lazarev

#### For citation

Fedulov, A.S., Lazarev, A.I., (2023) 'Organization of adaptive data routing in electric power complexes using ontological fuzzy classifiers', *Software & Systems*, 36(3), pp. 442–450 (in Russ.). doi: 10.15827/0236-235X.142.442-450

#### Article info

Received: 29.05.2023

After revision: 04.07.2023

Accepted: 05.07.2023

**Abstract.** The paper discusses the theoretical aspects of the machine learning application methods, in particular, the adaptation of deep models to the TCP/IP network topologies management in electric power complexes. The subject of the research in the paper is the author's approach to the organization of centralized network segments management in the field under consideration. The study of the interaction subjects in electric power units processes on the basis of the developed ontological models allowed to identify the main properties of multiformat data that may represent vulnerabilities for exploiting vulnerabilities. The practical significance of the research is represented by the development of a multi-module structure for tracking, classifying and predicting changes in consumed traffic, due to which it is possible to increase the efficiency of complex corporate network structures. Practical testing of existing algorithms for obtaining hash functions was carried out - the results allowed to conclude that it is advisable to use the basic BLAKE3 encryption algorithm as the main mechanism for verifying the authenticity of clients in comparison with the SHA-384, SHA-512, SHA-224, MD5

algorithms. The analytical implementation of the fuzzy character-by-character comparison algorithm as a decision-making module is given - this also allowed to confirm the relevance of the proposed approach when working with fuzzy data structures. As the main solution to these problems, an implemented approach to flexible management of the electric power plants segment represented by a complex of generating, electric grid, power supply and other companies is proposed. The main result of the proposed solution is the possible changes centralized analysis approach, taking into account adaptation to network loads based on selected ontological variables. Additional features in the implementation of this approach are compatibility with existing hardware network devices due to the unique architecture of the topology built.

**Keywords:** electric power systems, ontological models, data classification, decision-making, fuzzy logic, security, deep learning

**Acknowledgements.** This study was performed within the framework of the state assignment, project № FSWF-2023-0012

### Reference List

1. Vasilyev, D. (2021) 'Cross-subsidization in the electric power industry: Current state and problem-solving vectors', *J. of Modern Competition*, 15(3), pp. 17–30 (in Russ.). doi: 10.37791/2687-0649-2021-15-3-17-30.
2. Sklyuev, A., Khabarov, V., Musatova, I., Popova, O. (2022) 'Organizational and managerial innovations in the Russian electronics industry: Current trends', *J. of Modern Competition*, 16(6), pp. 103–116 (in Russ.). doi: 10.37791/2687-0657-2022-16-6-103-116.
3. Zientara, D. (2018) *Mastering pfSense: Manage, Secure, and Monitor your On-premise and Cloud Network with pfSense 2.4*. Birmingham: Packt Publ., 450 p.
4. Muthukumar, M., Senthikumar, P., Jawahar, M. (2019) 'Firewall scheduling and routing using pfSense', in *AISC*, 1172, pp. 749–757. doi: 10.1007/978-981-15-5566-4\_67.
5. Syaifei, W.A., Soetrisno, Y.A.A., Prasetijo, A.B. (2020) 'Simple smart algorithm for flexibility of dynamic allocation in DHCP server for SOHO wireless router', *Proc. Int. Conf. CENIM*, pp. 321–325. doi: 10.1109/CENIM51130.2020.9297852.
6. Yi, B., Congxiao, B., Xing, L. (2016) 'FlowLAN: A non-tunneling distributed virtual network based on IPv6', *Proc. IEEE ITNEC*, pp. 229–234. doi: 10.1109/ITNEC.2016.7560355.
7. Singh, G.D. (2022) *The Ultimate Kali Linux Book: Perform Advanced Penetration Testing Using Nmap, Metasploit, Aircrackng, and Empire*. Birmingham: Packt Publ., 742 p.
8. Bobryakov, A., Borisov, V., Misnik, A., Prakapenka, S. (2022) 'Modeling and design of information-analytical production processes based on neuro-fuzzy temporal Petri nets', *J. of Applied Inform.*, 17(2), pp. 65–78 (in Russ.). doi: 10.37791/2687-0649-2022-17-2-65-78.
9. Yi, T., Chen, X., Zhu, Y., Ge, W., Han, G. (2023) 'Review on the application of deep learning in network attack detection', *J. of Network and Comput. Applicat.*, 212, art. 103580. doi: 10.1016/j.jnca.2022.103580.
10. Aouini, Z., Pekar, A. (2021) 'NFSstream: A flexible network data analysis framework', *Comput. Networks*, 204, art. 108719. doi: 10.1016/j.comnet.2021.108719.
11. Puchkov, A., Dli, M., Prokinnov, N., Shutova, D. (2022) 'Multilevel algorithms for evaluating and making decisions on the optimal control of an integrated system for processing fine ore raw materials', *J. of Applied Inform.*, 17(6), pp. 102–121 (in Russ.). doi: 10.37791/2687-0649-2022-17-6-102-121.
12. Meshalkin, V., Dli, M., Puchkov, A., Lobaneva, E. (2021) 'Preliminary assessment of the pragmatic value of information in the classification problem based on deep neural networks', *J. of Applied Inform.*, 16(3), pp. 9–20 (in Russ.). doi: 10.37791/2687-0649-2021-16-3-9-20.
13. Dli, M., Sinyavsky, Yu., Rysina, E., Vasilkova, M. (2022) 'A method for classifying mixing devices using deep neural networks with an expanded receptive field', *J. of Applied Inform.*, 17(5), pp. 51–61 (in Russ.). doi: 10.37791/2687-0649-2022-17-5-51-61.
14. Dli, M., Bulygina, O., Sokolov, A. (2020) 'Rubrication of text information based on the voting of intellectual classifiers', *J. of Applied Inform.*, 15(5), pp. 29–36 (in Russ.). doi: 10.37791/2687-0649-2020-15-5-29-36.
15. Dli, M., Vlasova, E., Sokolov, A., Morgunova, E. (2021) 'Creation of a chemical-technological system digital twin using the Python language', *J. of Applied Inform.*, 16(1), pp. 22–31 (in Russ.). doi: 10.37791/2687-0649-2021-16-1-22-31.

### Авторы

**Федулов Александр Сергеевич**<sup>1</sup>, д.т.н.,  
профессор, директор,  
зав. кафедрой вычислительной техники,  
director@sbmpei.ru  
**Лазарев Алексей Игоревич**<sup>1</sup>, старший лаборант,  
anonymous.prodject@gmail.com

### Authors

**Alexander S. Fedulov**<sup>1</sup>, Dr.Sc. (Engineering),  
Professor, Director,  
Head of the Department Computer Engineering,  
director@sbmpei.ru  
**Alexey I. Lazarev**<sup>1</sup>, Senior Laboratory Assistant,  
anonymous.prodject@gmail.com

<sup>1</sup> Филиал Национального исследовательского университета «МЭИ» в г. Смоленске, г. Смоленск, 214013, Россия

<sup>1</sup> Branch of the National Research University "MPEI" in Smolensk, Smolensk, 214013, Russian Federation

## Архитектура системы поддержки принятия решений в процессе мониторинга технического состояния критического оборудования

В.А. Емельянов  
С.Г. Черный  
Н.Ю. Емельянова

### Ссылка для цитирования

Емельянов В.А., Черный С.Г., Емельянова Н.Ю. Архитектура системы поддержки принятия решений в процессе мониторинга технического состояния критического оборудования // Программные продукты и системы. 2023. Т. 36. № 3. С. 451–458. doi: 10.15827/0236-235X.142.451-458

### Информация о статье

Поступила в редакцию: 30.01.2023

После доработки: 24.03.2023

Принята к публикации: 28.03.2023

**Аннотация.** В работе решается прикладная задача совершенствования существующей системы диагностики критического футерованного оборудования без его вывода из эксплуатации. Актуальность исследования обусловлена необходимостью повышения уровня автоматизации и объективности принятия решений при эксплуатации этого оборудования для предотвращения аварийных ситуаций на производстве. Целью являются проектирование архитектуры и разработка системы поддержки принятия решений в процессе мониторинга технического состояния критического футерованного оборудования. Для высокоуровневого проектирования архитектуры данной системы использован язык моделирования архитектуры ArchiMate. Для проектирования статической структуры системы в виде основных сущностей, реализующих функционал системы по поддержке принятия решений о допустимости использования критического футерованного оборудования, применены объектно-ориентированный подход (включая объектно-ориентированный анализ, объектно-ориентированное проектирование и программирование) и унифицированный язык моделирования UML. В ходе работы построена модель верхнеуровневой архитектуры системы поддержки принятия решений при мониторинге технического состояния критического футерованного оборудования. Выполнен анализ функционала, проведено объектно-ориентированное проектирование статической структуры системы поддержки принятия решений. Разработано ПО системы поддержки принятия решений в процессе мониторинга технического состояния критического футерованного оборудования, позволяющее генерировать рекомендации относительно режимов его эксплуатации и осуществлять профилактическую диагностику данного оборудования. Разработанное ПО протестировано в условиях металлургического производства в технологическом процессе диагностики передвижных миксеров ПМ350 на Алчевском металлургическом комбинате.

**Ключевые слова:** система поддержки принятия решений, архитектура, объектная модель, ПО, футерованное оборудование

Критически важным для металлургических и машиностроительных предприятий является футерованное оборудование, такое как стационарные и передвижные миксеры, сталеразливочные ковши. От его мониторинга и своевременной диагностики во многом зависит способность предприятия предотвращать аварии и обеспечивать непрерывность технологического процесса выпуска металлов. Этим и обусловлена необходимость разработки и совершенствования систем диагностики футерованного оборудования.

### Анализ источников и постановка задачи

В работах [1, 2] наглядно продемонстрировано выполнение диагностики и мониторинга технического состояния критического футерованного оборудования с помощью средств, функционирующие которых характеризуется недопустимым уровнем погрешности измерения. В настоящее время существует большое разнообразие подходов и систем [3–5] в обла-

сти автоматизации контроля технического состояния такого футерованного оборудования, как сталь-ковши. В работе [6] предлагается система диагностики футеровки сталь-ковша на основе инфракрасной термометрии. Авторы исследования [7] представляют лазерный измеритель и термографию в сталеразливочном ковше в качестве методов оценки, внедренных в Nippon Steel Corporation. В [8] предложена система диагностики сталь-ковша на основе программной платформы LabVIEW, позволяющая осуществлять мониторинг состояния рабочего слоя внутренней поверхности ковша. Все эти работы направлены на решение общей проблемы при эксплуатации футерованного оборудования, заключающейся в низком уровне автоматизации диагностики и контроля состояния оборудования, о чем также свидетельствуют исследования [9, 10]. Анализ приведенных источников позволяет сделать вывод о том, что существующие модели, методы и системы не позволяют выполнять диагностику и мониторинг футерованного оборудования без вы-

вода его из штатного режима работы, то есть в системах отсутствует возможность профилактической диагностики. Во многих системах диагностики футерованного оборудования задача определения режима его эксплуатации возлагается на технолога, что в силу субъективного фактора и низкого уровня объективности принятия решений приводит к проблемным и аварийным ситуациям [11, 12]. Таким образом, возникает необходимость совершенствования существующих моделей и систем диагностики критического футерованного оборудования без его вывода из эксплуатации. Поэтому актуально проведение научных исследований, направленных на разработку систем и средств для автоматизации оценки технического состояния и поддержки принятия решений в процессе работы такого оборудования.

### Проектирование архитектуры системы

В данном исследовании предлагается новая система, позволяющая осуществлять поддержку принятия решений о допустимости использования оборудования без вывода его из штатных режимов эксплуатации. Для разработки такой системы необходимо ее детальное проектирование.

Авторами выполнено высокоуровневое проектирование архитектуры системы поддержки принятия решений в процессе мониторинга технического состояния критического футерованного оборудования, которая представляет собой описание функциональных модулей и компонент ПО системы, а также связей между ними. Проектируемая система основана на разработанном ранее подходе и методах автоматизированной диагностики из работы [13]. Реализация методов дает возможность системе осуществлять поддержку принятия решений о допустимости использования футерованного оборудования без вывода его из штатных режимов эксплуатации.

Для отражения архитектуры системы построены следующие модели:

- модель использования системы поддержки принятия решений в процессе мониторинга технического состояния критического футерованного оборудования в виде диаграммы вариантов использования;
- концептуальная модель для отражения статической структуры системы в виде основных сущностей, на которые возлагается реализация функциональности системы;
- верхнеуровневая модель архитектуры системы поддержки принятия решений в про-

цессе мониторинга технического состояния критического футерованного оборудования.

Проектирование функционала системы поддержки принятия решений выполнено с помощью унифицированного языка UML в ПО StarUML. В процессе построено 11 диаграмм вариантов использования. Диаграмма, приведенная на рисунке 1, отражает обобщенную модель использования системы.

В модели выделены роль пользователя системы «Технолог» и шесть основных вариантов ее использования: «Формирование термограммы», «Распознавание термограмм», «Оценить состояние футеровки», «Сформировать рекомендации», «Оценить состояние корпуса», «Создать отчет». Варианты использования имеют свою декомпозицию и сценарии по каждой функции.

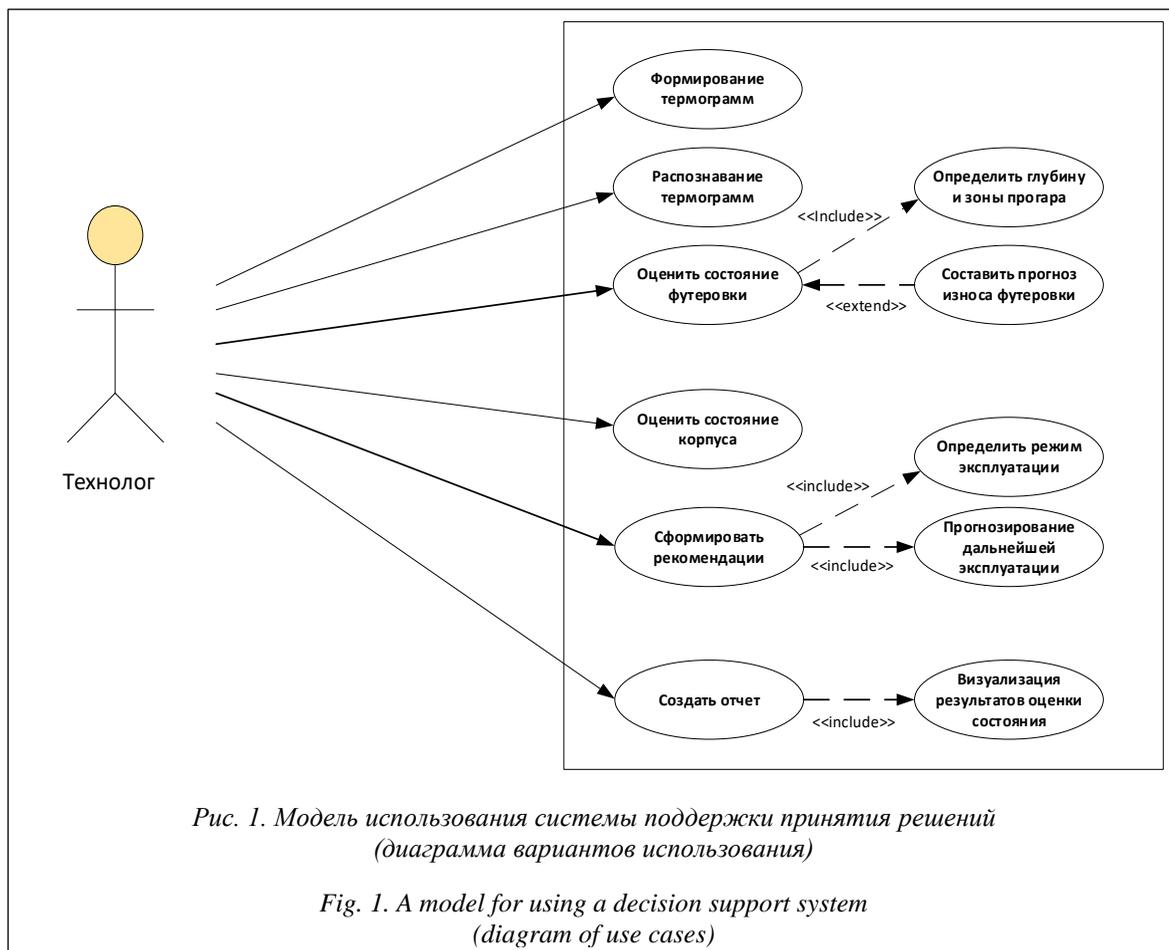
Для отображения статической структуры системы поддержки принятия решений построено восемь диаграмм классов этапа анализа, отражающих основные сущности (классы), на которые возлагается реализация функций системы поддержки принятия решений. Общая диаграмма классов приведена на рисунке 2.

На диаграмме классов системы поддержки принятия решений выделены описанные далее сущности.

В «Личном кабинете технолога» выполняются операции по регистрации нового пользователя-технолога и по входу в личный кабинет уже существующего.

«Конфигуратор» содержит операции для конфигурирования исследования критического футерованного оборудования: `ImportDiagnosingTask()` загружает в систему поддержки принятия решений исходные данные по проводимому анализу критического футерованного оборудования (цех, описание оборудования, номер и т.д.); `CreateDiagnosingReport()` реализует функцию создания отчета о проведенном исследовании критического футерованного оборудования; `SendReport()` реализует функцию «Отправить результаты в цех»; `ChangeAnalyseSettings()` изменяет параметры проводимого исследования (например, добавить оценку свойств металла и др. настройки); `ChooseConfiguration()` позволяет выбрать один из предустановленных профилей диагностики критического футерованного оборудования.

«Модуль оценки технического состояния» содержит операции для оценки износа футеровки и технического состояния корпуса футерованного оборудования: `GetBurnOutZones()` определяет глубину и зоны прогара футеровки;



LiningPrediction() составляет прогноз износа футеровки; LiningAssesment() оценивает состояние футеровки; MetalAssesment() оценивает состояние корпуса.

В составе «Анализатора термограмм» операции для обработки изображений термограмм критического футерованного оборудования. Он позволяет выполнять набор функций, связанных с вариантом использования «Распознавание термограмм». LoadThermogram() осуществляет загрузку сформированного изображения термограммы критического футерованного оборудования. Для выполнения операций по предварительной обработке изображения термограммы критического футерованного оборудования используются операции BinarizeImage(), Filtration(), Vectorization, предназначенные для бинаризации, фильтрации (разными методами) и векторизации изображения соответственно. Для распознавания термограммы критического футерованного оборудования используется операция RecognizeThermogram(). Сохранение обработанного изображения с вектором числовых характеристик этого изображения реализуется операцией SaveResult().

Поскольку для оценки футеровки критического футерованного оборудования система поддержки принятия решений предполагает использовать нейронные сети, в модели выделена сущность «Нейронная сеть», которая содержит атрибуты и операции для создания и обучения нейронных сетей.

«Модуль ППР» содержит операции для формирования рекомендаций относительно исследуемого критического футерованного оборудования: GetOperationMode() определяет наиболее рациональный режим эксплуатации критического футерованного оборудования; SavePrecedent() сохраняет в базу прецедентов новый прецедент, для которого не был найден подобный прецедент в базе; CreateRecomendations() генерирует для технолога рекомендации о дальнейшем использовании критического футерованного оборудования; SaveKnowledge() сохраняет новые знания в предметной области, которые могут появляться в процессе исследования; ChangeExpertSettings() изменяет настройки модуля поддержки принятия решений.

На рисунке 3 представлена верхнеуровневая модель архитектуры системы поддержки

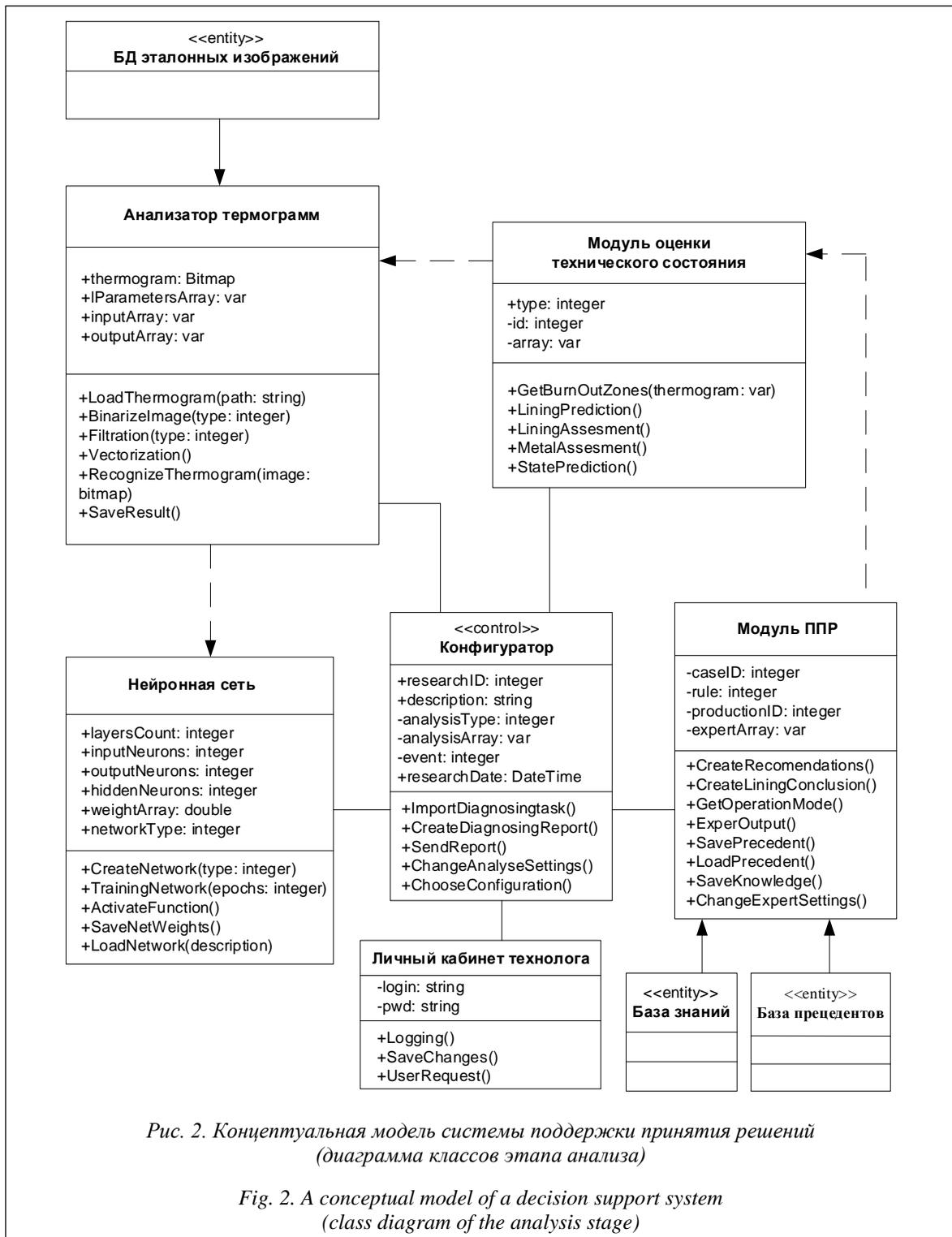


Рис. 2. Концептуальная модель системы поддержки принятия решений (диаграмма классов этапа анализа)

Fig. 2. A conceptual model of a decision support system (class diagram of the analysis stage)

принятия решений, выполненная в программном средстве Archi на языке моделирования ArchiMate.

В архитектуре системы поддержки принятия решений были выделены модули:

- обработки термограмм футерованного оборудования;

- оценки технического состояния футерованного оборудования;
- поддержки принятия решений;
- нейронных сетей;
- управления.

Пользователю системы (технологу) доступны следующие сервисы приложений:

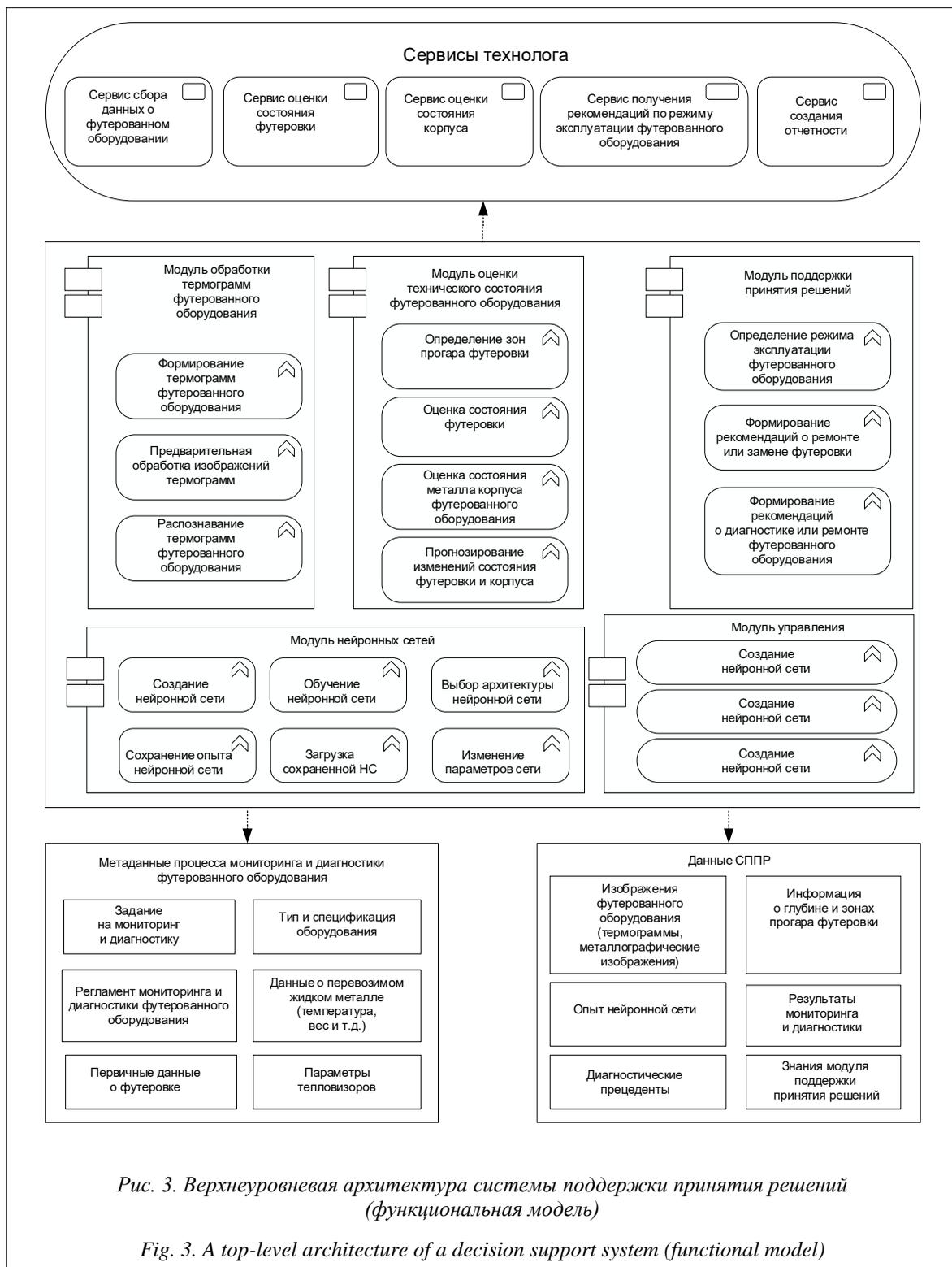


Рис. 3. Верхнеуровневая архитектура системы поддержки принятия решений (функциональная модель)

Fig. 3. A top-level architecture of a decision support system (functional model)

- сбор данных о футерованном оборудовании;
- оценка состояния футеровки;
- оценка состояния корпуса;
- получение рекомендаций по режиму эксплуатации футерованного оборудования;
- создание отчетности.

В процессе проектирования выделены отдельные категории метаданных, которые будут использоваться системой поддержки принятия решений. К основным данным отнесены:

- документ-задание на мониторинг и диагностику футерованного оборудования;

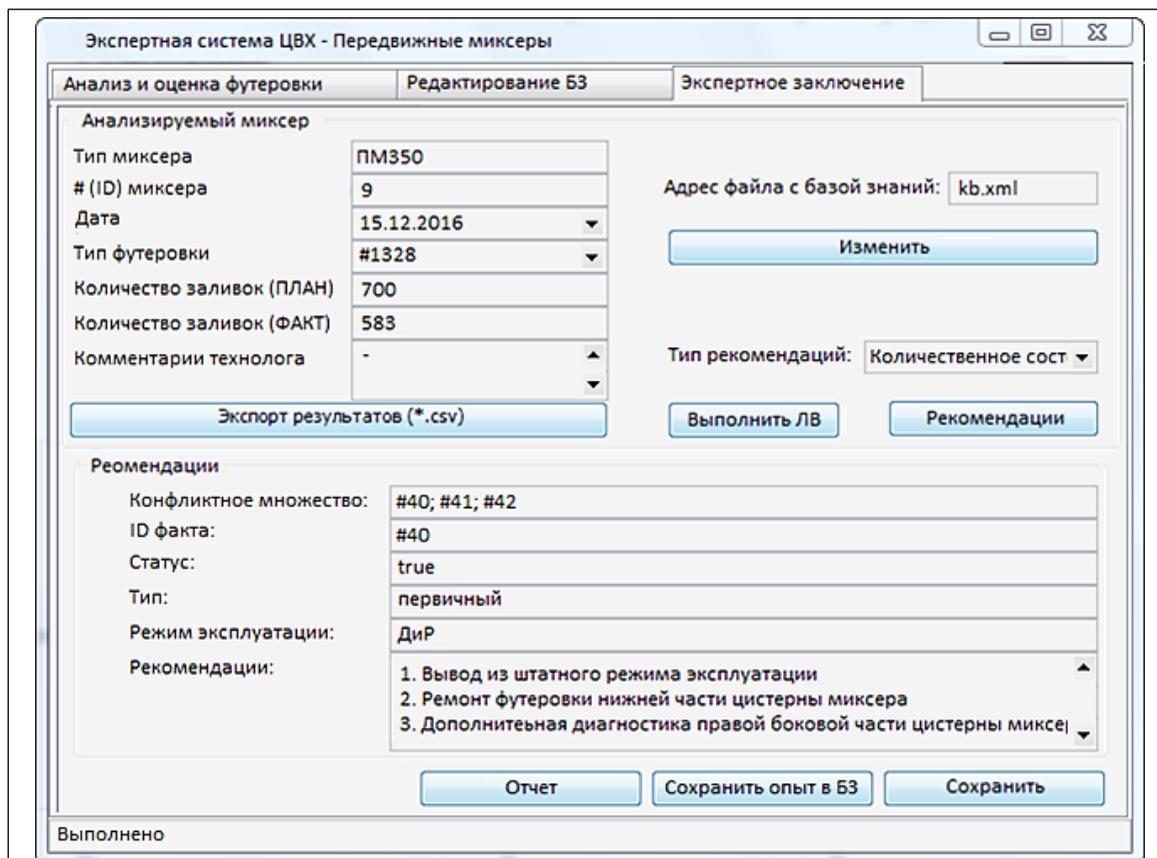


Рис. 4. ПО системы поддержки принятия решений в процессе мониторинга технического состояния критического футерованного оборудования

Fig. 4. Decision support system software in the process of monitoring the technical condition of critical lined equipment

- тип и спецификация футерованного оборудования;
- регламент мониторинга и диагностики футерованного оборудования;
- данные о перевозимом в футерованном оборудовании жидком металле (температура металла, вес жидкого металла и др.);
- параметры тепловизоров, с помощью которых формируются термограммы футерованного оборудования;
- первичные данные о футеровке (материал, состав и др.).

### Разработка ПО системы

Созданное ПО системы поддержки принятия решений в процессе мониторинга технического состояния критического футерованного оборудования (рис. 4) функционирует в двух режимах: анализа и обучения.

Разработанное ПО успешно апробировано на Алчевском металлургическом комбинате

для диагностики передвижных миксеров типа ПМ350.

### Выводы

В ходе исследования создана модель верхнеуровневой архитектуры системы поддержки принятия решений в процессе мониторинга технического состояния критического футерованного оборудования. Отличительной особенностью данной системы является возможность поддержки принятия решений о допустимости использования футерованного оборудования без вывода его из штатных режимов эксплуатации, которая достигается благодаря реализации разработанных методов автоматизированной диагностики, основанных на использовании нейронных сетей и теории прецедентов для диагностики технического состояния футерованного оборудования.

На основе модели разработано ПО для распознавания изображений термограмм критиче-

ского футерованного оборудования, которое позволяет автоматически определять режим эксплуатации оборудования и генерировать для

технолога рекомендации относительно технического состояния футеровки и вида ремонта критического футерованного оборудования.

#### Список литературы

1. Jančar D., Machů M., Velička M., Tvardek P., Kocián L., Vlček J. Use of neural networks for lifetime analysis of teeming ladles. *Materials*, 2022, vol. 15, art. 8234. doi: 10.3390/ma15228234.
2. Branca T.A., Fornai B., Colla V., Murri M.M., Streppe E., Schröder A.J. The challenge of digitalization in the steel sector. *Metals*, 2020, vol. 10, no. 2, art. 288. doi: 10.3390/met10020288.
3. Chakraborty B., Sinha B. Process-integrated steel ladle monitoring, based on infrared imaging – a robust approach to avoid ladle breakout. *QIRT*, 2019, vol. 17, no. 3, pp. 169–191. doi: 10.1080/17686733.2019.1639112.
4. Li G., Jiang D., Sun Y., Jiang G., Tao B. Life prediction mechanism of ladle composite structure body based on simulation technology. *Arch. Metall. Mater.*, 2019, vol. 64, no. 4, pp. 1555–1562.
5. Vannucci M., Colla V., Chini M., Gaspardo D., Palm B. Artificial intelligence approaches for the ladle predictive maintenance in electric steel plant. *IFAC-PapersOnLine*, 2022, vol. 55, no. 2, pp. 331–336. doi: 10.1016/j.ifacol.2022.04.215.
6. Lee S.J., Jeon Y.J., Kim S.W. Analyzing refractory bricks of ladles using infrared images. *J. ICROS*, 2015, vol. 21, no. 4, pp. 291–300. doi: 10.5302/j.icros.2015.14.8036.
7. Sawai S., Shiokawa M., Hashimoto K., Yamada Y. Methods of evaluating the damage of steelmaking refractories. *Nippon Steel Tech. Report*, 2020, no. 125, pp. 81–86.
8. Chang W., Sun Y., Li G., Jiang G., Kong J., Jiang D., Liu H. Ladle health monitoring system based on LabVIEW. *IJCSM*, 2018, vol. 9, no. 6, pp. 566–576. doi: 10.1504/IJCSM.2018.096311.
9. Mihailov E., Petkov V., Doichev I., Boshnakov K. Model-based approach for investigation of ladle lining damages. *IREME*, 2013, vol. 7, no. 2, pp. 122–130.
10. Lamm R., Kirchhoff S. Optimization of ladle refractory lining, gap and crack detection, lining surface temperature and sand-filling of the ladle-tap hole by means of a 3d-laserprofile-measurement system that is immersed into a hot ladle to evaluate the entire condition. *Proc. UNITECR*, 2017. URL: [http://www.unitecr2017.mundodecongresos.com/abstracts/Paper\\_rbofbhfcxshpgipoispm.pdf](http://www.unitecr2017.mundodecongresos.com/abstracts/Paper_rbofbhfcxshpgipoispm.pdf) (дата обращения: 10.10.2022).
11. Colpaert H. *Metallography of Steels: Interpretation of Structure and the Effects of Processing*. USA, ASM Int. Publ., 2018, 699 p. doi: 10.31399/asm.tb.msisep.9781627082594.
12. Martyushev N., Skeebe V. The method of quantitative automatic metallographic analysis. *JPCS*, 2017, vol. 803, art. 012094. doi: 10.1088/1742-6596/803/1/012094.
13. Chernyi S., Yemelyanov V., Zinchenko E., Zinchenko A., Tsvetkova O., Mishin A. Application of artificial intelligence technologies for diagnostics of production structures. *JMSE*, 2022, vol. 10, no. 2, art. 259. doi: 10.3390/jmse10020259.

### An architecture of the decision support system for monitoring a technical condition of critical equipment

Vitaliy A. Yemelyanov  
Sergei G. Chernyi  
Nataliya Yu. Yemelyanova

#### For citation

Yemelyanov, V.A., Chernyi, S.G., Yemelyanova, N.Yu. (2023) 'An architecture of the decision support system for monitoring a technical condition of critical equipment', *Software & Systems*, 36(3), pp. 451–458 (in Russ.). doi: 10.15827/0236-235X.142.451-458

#### Article info

Received: 30.01.2023

After revision: 24.03.2023

Accepted: 28.03.2023

**Abstract.** The paper solves an applied problem of improving the existing system for diagnosing critical lined equipment without its decommissioning. The relevance of the work is due to the need to increase the level of automation and objectivity of decision-making during operation of critical lined equipment to prevent accidents in production. The purpose of the study is designing the architecture of a decision support system for monitoring the technical condition of critical lined equipment. As for research methods, for high-level architecture design of the decision support system in the process of monitoring the technical condition of critical lined equipment, we used the architecture modeling language ArchiMate. An object-oriented approach (including object-oriented analysis, object-oriented design and programming) and the unified modelling language UML were used to design the static system structure in the form of the main entities that are responsible for implementing the system functionality to support decision making on the admissibility of using critical lined equipment.

As a result, there is a developed model of the top-level architecture of the decision support system for monitoring the technical condition of critical lined equipment. The developed software for a decision support system for monitoring the technical condition of critical lined equipment makes it possible to generate recommendations regarding the operating modes of critical lined equipment. The developed software was tested in the conditions of metallurgical production in the technological process of diagnosing mobile mixers PM350 at the Alchevsk Iron and Steel Works.

**Keywords:** decision support system, architecture, object model, lined equipment, software

### Reference List

1. Jančar, D., Machů, M., Velička, M., Tvardek, P., Kocián, L., Vlček, J. (2022) 'Use of neural networks for lifetime analysis of teeming ladles', *Materials*, 15, art. 8234. doi: 10.3390/ma15228234.
2. Branca, T.A., Fornai, B., Colla, V., Murri, M.M., Streppa, E., Schröder, A.J. (2020) 'The challenge of digitalization in the steel sector', *Metals*, 10(2), art. 288. doi: 10.3390/met10020288.
3. Chakraborty, B., Sinha, B. (2019) 'Process-integrated steel ladle monitoring, based on infrared imaging – a robust approach to avoid ladle breakout', *QIRT*, 17(3), pp. 169–191. doi: 10.1080/17686733.2019.1639112.
4. Li, G., Jiang, D., Sun, Y., Jiang, G., Tao, B. (2019) 'Life prediction mechanism of ladle composite structure body based on simulation technology', *Arch. Metall. Mater.*, 64(4), pp. 1555–1562.
5. Vannucci, M., Colla, V., Chini, M., Gaspario, D., Palm, B. (2022) 'Artificial intelligence approaches for the ladle predictive maintenance in electric steel plant', *IFAC-PapersOnLine*, 55(2), pp. 331–336. doi: 10.1016/j.ifacol.2022.04.215.
6. Lee, S.J., Jeon, Y.J., Kim, S.W. (2015) 'Analyzing refractory bricks of ladles using infrared images', *J. ICROS*, 21(4), pp. 291–300. doi: 10.5302/j.icros.2015.14.8036.
7. Sawai, S., Shiokawa, M., Hashimoto, K., Yamada, Y. (2020) 'Methods of evaluating the damage of steelmaking refractories', *Nippon Steel Tech. Report*, (125), pp. 81–86.
8. Chang, W., Sun, Y., Li, G., Jiang, G., Kong, J., Jiang, D., Liu, H. (2018) 'Ladle health monitoring system based on LabVIEW', *IJCSM*, 9(6), pp. 566–576. doi: 10.1504/IJCSM.2018.096311.
9. Mikhailov, E., Petkov, V., Doichev, I., Boshnakov, K. (2013) 'Model-based approach for investigation of ladle lining damages', *IREME*, 7(2), pp. 122–130.
10. Lamm, R., Kirchhoff, S. (2017) 'Optimization of ladle refractory lining, gap and crack detection, lining surface temperature and sand-filling of the ladle-tap hole by means of a 3d-laserprofile-measurement system that is immersed into a hot ladle to evaluate the entire condition', *Proc. UNITECR*, available at: [http://www.unitecr2017.mundodecongressos.com/abstracts/Paper\\_rbofbhfxcsxhpgipoispm.pdf](http://www.unitecr2017.mundodecongressos.com/abstracts/Paper_rbofbhfxcsxhpgipoispm.pdf) (accessed October 10, 2022).
11. Colpaert, H. (2018) *Metallography of Steels: Interpretation of Structure and the Effects of Processing*. USA: ASM Int. Publ., 699 p. doi: 10.31399/asm.tb.msisep.9781627082594.
12. Martyshev, N., Skeeba, V. (2017) 'The method of quantitative automatic metallographic analysis', *JPCS*, 803, art. 012094. doi: 10.1088/1742-6596/803/1/012094.
13. Chernyi, S., Emelianov, V., Zinchenko, E., Zinchenko, A., Tsvetkova, O., Mishin, A. (2022) 'Application of artificial intelligence technologies for diagnostics of production structures', *JMSE*, 10(2), art. 259. doi: 10.3390/jmse10020259.

### Авторы

**Емельянов Виталий Александрович**<sup>1</sup>,  
д.т.н., профессор, v.yemelyanov@gmail.com  
**Черный Сергей Григорьевич**<sup>2,3</sup>, к.т.н., доцент,  
зав. кафедрой, научный сотрудник,  
sergiiblack@gmail.com  
**Емельянова Наталия Юрьевна**<sup>1</sup>, к.т.н., доцент,  
n.yemelianova@gmail.com

<sup>1</sup> Финансовый университет  
при Правительстве Российской Федерации,  
г. Москва, 125167, Россия

<sup>2</sup> Керченский государственный  
морской технологический университет,  
г. Керчь, 298300, Россия

<sup>3</sup> Санкт-Петербургский государственный  
морской технический университет,  
г. Санкт-Петербург, 198035, Россия

### Authors

**Vitaliy A. Yemelyanov**<sup>1</sup>, Dr.Sc. (Engineering),  
Professor, v.yemelyanov@gmail.com  
**Sergei G. Chernyi**<sup>2,3</sup>, Ph.D. (Engineering),  
Associate Professor, Chief of Department,  
Research Associate, sergiiblack@gmail.com  
**Nataliya Yu. Yemelyanova**<sup>1</sup>, Ph.D. (Engineering),  
Associate Professor, n.yemelianova@gmail.com

<sup>1</sup> Financial University under the Government  
of the Russian Federation,  
Moscow, 125167, Russian Federation

<sup>2</sup> Kerch State Maritime Technological University,  
Kerch, 298300,  
Russian Federation

<sup>3</sup> St. Petersburg State Marine Technical University,  
St. Petersburg, 198035,  
Russian Federation

## Программный комплекс планирования производства и управления запасами

Г.М. Соломаха  
В.А. Тулуева  
С.В. Хижняк

### Ссылка для цитирования

Соломаха Г.М., Тулуева В.А., Хижняк С.В. Программный комплекс планирования производства и управления запасами // Программные продукты и системы. 2023. Т. 36. № 3. С. 459–465. doi: 10.15827/0236-235X.142.459-465

### Информация о статье

Поступила в редакцию: 16.02.2023

После доработки: 12.03.2023

Принята к публикации: 14.03.2023

**Аннотация.** В работе представлен процесс взаимодействия информационных систем в ходе планирования производства и управления запасами с учетом динамичности производственной среды и ограниченности возможностей предприятия. Рассмотрены особенности промышленных предприятий, не позволяющие использовать стандартные реализации предлагаемых на рынке программных продуктов и систем. Основным модулем для оптимизации процесса планирования и управления запасами является ERP-система. Построение прогноза продаж и уровня страховых запасов выполняется при помощи отечественного программного продукта Forecast NOW, планирование производства осуществляется в SCM-системе SAP APO. Результатом работы и взаимодействия данных систем являются план-график производства, который, в свою очередь, передается в систему MES для производства, а также корректные данные о потребностях закупки продуктов, сырья, материалов и полуфабрикатов. В качестве методов используются системный подход, а также анализ и синтез исследований отечественных ученых в области планирования и управления запасами. Сформулированы требования к разрабатываемым в рамках интеграции систем программным модулям. Показано, что предложенный программный комплекс удовлетворяет этим требованиям. Приведен пример возможной интеграции разработанного программного комплекса с системой мониторинга производственных процессов, представлена соответствующая схема интеграции. Показано, что область применения данного подхода не ограничивается крупными предприятиями и может быть скорректирована для использования с соответствующими программными комплексами предприятий меньшего масштаба. Гибкость полученного программного решения открывает возможности для дальнейшего расширения области применения и реализации более масштабных самостоятельных производственных систем и комплексов программ.

**Ключевые слова:** информационная система, производственное планирование, управление запасами, производственные запасы, планирование продаж, мониторинг на производстве, автоматизация

Обеспечение единого непрерывного снабжения всех систем производственного процесса необходимыми запасами в оптимальном количестве и заданного качества – важнейшая гарантия эффективного функционирования промышленных предприятий.

Существует множество исследований по теории и практике управления запасами, результатом которых являются различные модели и примеры их применения, однако каждая из этих моделей решает лишь отдельную задачу – или управления запасами [1, 2], или планирования производства [3].

В связи с этим остается спорной сформированность самой системы управления запасами, выполняющей роль регулятора процесса движения материального потока и позволяющей свести к минимуму его случайность [4]. К тому же существующие на рынке программные продукты не позволяют совместно решать задачи и управления запасами, и планирования производства [5–7].

В рамках данного исследования предлагается разработка, объединяющая решение обеих

задач и ряда проблем, возникающих из-за усложнения процесса управления запасами особенностями промышленных предприятий, которые не позволяют использовать стандартные реализации предлагаемых на рынке программных продуктов и систем, требующих интеграции различных систем в единое целое.

Основными из таких особенностей являются следующие:

- неопределенность спроса;
- непостоянность затрат на поставку и хранение продуктов, стоимости хранимых продуктов, скидок клиентам;
- рассмотрение плановых периодов различной длины;
- наличие страховых запасов на востребованные продукты, которые необходимо хранить в определенных условиях для исключения возможной порчи продукции, морально и физически устаревших ресурсов, минимальных производственных партий (партий заказа), остатки от которых также необходимо хранить на складе, поставок продуктов и комплектующих из других стран;

– возможность отказа клиентов от заказанной продукции, что увеличивает остаток неликвидной продукции на складе.

Развитие производства, а также бизнеса в целом, его планирование и управление запасами невозможно представить без автоматизации. Внедрение ERP-системы и ее интеграция с системами MES и SCM позволяют существенно снизить издержки предприятия, повысить уровень обслуживания, оптимизировать производственный цикл, уменьшить складские запасы и, соответственно, повысить производительность предприятия, его рентабельность и уровень контроля производственного процесса. Однако внедрение данных систем и их поддержка довольно затратны, что снижает интерес к ним со стороны малых предприятий. В связи с этим необходимо заранее оценить целесообразность использования указанных систем управления запасами для конкретного предприятия.

Информационные технологии, являясь основным ресурсом интеграционного процесса, подразумевают необходимость создания банков данных, которые дают возможность организовать информационный обмен на постоянной основе. Обобщенная схема взаимосвязей компонентов программного комплекса планирования производства и управления запасами предложена в [8].

Данная схема включает в себя БД, содержащую нормативно-справочную информацию, которая подразделяется на статическую и динамическую, а также модули формирования производственной программы и плана характеристик движения производства, построения оптимальных производственных расписаний. Однако указанная схема не подразумевает получение обратной информации о производстве продукции, упрощая процесс учета сырья, полуфабрикатов и готовой продукции и, соответственно, управление запасами в целом.

Вместе с тем при оптимизации процесса планирования и управления запасами может быть использована ERP-система, имеющая доступ к массиву операционных данных, на основе которых проводится планирование потребности в материально-технических ресурсах, мощностей (средств производства, станков и оборудования, складов и мест хранения), денежных средств и прочего. Основными элементами данных являются материалы или продукты, заводы или местоположение, склады, поставщики, контрагенты, заказы клиентов (проектные, срочные и стандартные), соглаше-

ния о поставках, заказы и заявки на перемещение, исходящие и входящие поставки, резервирования перемещаемого запаса, страховой запас, плановые первичные потребности, запасы в пути, плановые заказы, вторичные потребности, запасы. Данная подсистема обладает функциональностью, позволяющей осуществлять не только планирование материально-технических ресурсов и мощностей, но и перевод соответствующих потребностей в денежные ресурсы [9].

Процесс планирования производства заключается в формировании планов-графиков на основе математической модели, разработанной для конкретного промышленного предприятия. Однако этот процесс усложняется большой номенклатурой товаров, наличием мелких производственных партий, а также несерийной продукции, требующей больших затрат на производство и имеющей высокую стоимость.

С целью минимизации издержек осуществляются укрупнение производственных партий и расчет величины страхового запаса, объем которого зависит от объема партии и времени поставки.

Для расчета страховых запасов целесообразно использовать SCM-системы. Например, отечественный продукт Forecast NOW позволяет строить прогноз потребности в продукции на складах либо с ежедневным контролем, либо по фиксированному расписанию, проводить расчет страховых запасов с учетом рисков дефицита, штрафов за недопоставку, затрат на хранение запасов, сезонности с точностью от месяца до дня, дат поставки товаров и сроков годности поступивших партий, наличия проектных продаж (большие разовые объемы, серьезно искажающие реальные средние показатели). Forecast NOW помогает распределять запасы между складами в условиях дефицита, а также переводить продукты в складские или заказные в зависимости от ситуации на рынке.

Ручной перенос данных между системами SCM, SAP и MES предполагает большие трудозатраты, а также увеличивает значение человеческого фактора. Кроме того, ручной перенос может быть осуществлен только в рабочее время ответственного сотрудника, что может существенно затруднить планирование производства. Соответственно, одной из основных целей предлагаемой интеграции систем является снижение трудозатрат на перенос данных между интегрируемыми системами и зависимости от рабочего расписания. Поэтому си-

стема должна отвечать требованию по автоматизации процессов загрузки и выгрузки данных.

Неизбежно возникает вопрос планирования выгрузки, а значит, конечная реализация должна обладать такой возможностью и при этом сохранять функцию переноса данных по требованию.

Чтобы использовать программные компоненты для широкого спектра производственных программных конфигураций, итоговые модули должны быть переносимыми с минимальными трудозатратами, а значит, их структура и взаимодействие должны обеспечивать гибкость и возможность замены реализации одного компонента без вмешательства в программный код других.

Для внешнего взаимодействия с различными конфигурациями производственных систем программные модули должны иметь возможность расширения или точечной замены реализации отдельных модулей для обеспечения работы с различными форматами представления данных и протоколами их передачи.

В ходе исследования существующих проблем и их возможных решений можно сформировать следующие требования к разрабатываемым в рамках интеграции программным модулям:

- возможность автоматической загрузки/выгрузки данных, загрузки/выгрузки данных по требованию и по расписанию, расширения и модификации для использования других форматов и протоколов передачи данных;
- переносимость разработанных программных модулей;
- поддержка стандартных форматов и протоколов внешнего и внутреннего взаимодействия.

На рисунке 1 представлена схема взаимодействия систем в рамках планирования производства.

Рассмотрим элементы схемы и их функции.

Системы Forecast NOW и ERP взаимодействуют в двустороннем порядке через экспорт, копирование и импорт информации. Для взаимодействия с сервером системы Forecast NOW разработан сервер Interactionserver. Взаимодействие происходит согласно ее спецификации.

В соответствии с установленным расписанием из ERP-системы модулем DataExportModule на сервер InteractionServer экспортируются фактические данные, после этого они копируются на сервер Forecast NOW. В си-

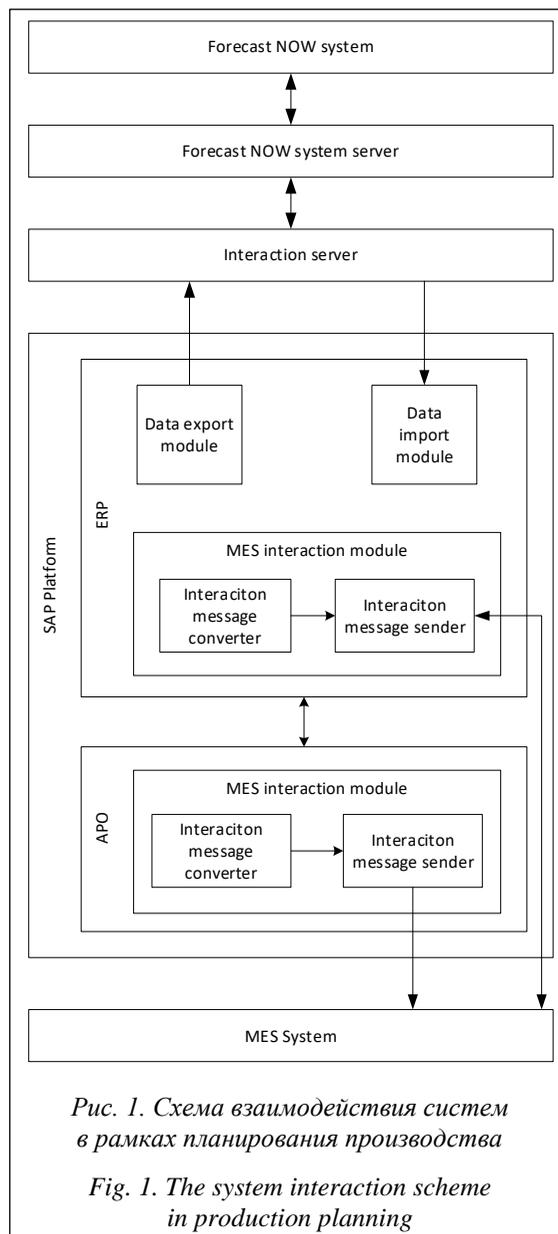


Рис. 1. Схема взаимодействия систем в рамках планирования производства

Fig. 1. The system interaction scheme in production planning

стему Forecast NOW осуществляется импорт данных и формируются прогнозные данные продаж, на основании которых рассчитывается необходимый уровень страхового запаса. Прогнозные данные продаж и страхового запаса передаются в ERP-систему по той же схеме экспорта, копирования и импорта через серверы с установленной периодичностью с помощью модуля DataImportModule. Таким образом реализуются возможности загрузки/выгрузки данных и загрузки/выгрузки данных по расписанию.

В ERP-системе данные обрабатываются и рассчитываются необходимые объемы продукции на складах отгрузки и производства, а также для закупки и производства.

Планирование производства на основании данных, полученных из системы Forecast NOW, осуществляется в SCM-системе SAP APO. Системы APO и ERP взаимодействуют друг с другом напрямую, так как имеют общую платформу SAP. Передача данных осуществляется в двустороннем порядке. Согласно установленному расписанию фактические данные интегрируются из ERP-системы в систему APO. С установленной периодичностью выполняется прогон производственного планирования, который формирует планы-графики производства, заказы на сырье и комплектующие, а также план переналадок оборудования.

На основе полученной информации после внесения необходимых корректировок строится план-график производства для соответствующих рабочих центров и передается обратно в систему SAP ERP, в которой формируются производственные заказы для дальнейшей передачи в MES-систему, а также заказы на поставку товаров, сырья и комплектующих.

Производственные заказы передаются в MES-систему преимущественно из системы производственного планирования SAP APO. Передача нарядов из этой системы позволяет автоматически передавать обновления производственных заказов в системе SAP ERP непосредственно в MES-систему при условии, что наряд не взят ею в работу. Также есть возможность передачи производственных заказов в MES-систему из системы SAP ERP по требованию, что обеспечивает загрузку/выгрузку данных по требованию.

Система MES позволяет сотрудникам производственных участков без отрыва от производства контролировать план-график и его изменения, оперативно передавать информацию о процессе в ERP-систему, а также оценивать загруженность производственных мощностей и эффективность работы сотрудников на соответствующих участках.

Системы APO и ERP взаимодействуют с MES-системой напрямую посредством IDOC-сообщений. Система MES отправляет в систему SAP ERP ответные IDOC-сообщения, а также сообщения о ходе выполнения производственного плана. Модуль Interaction Message Converter осуществляет преобразование данных, а модуль Interaction Message Sender их непосредственную отправку. Подобное разделение позволяет при необходимости расширить модуль InteractionMessageConverter для использования других форматов или модуль InteractionMessageSender для использования

других протоколов передачи данных. Формат IDOC, имеющий спецификацию и широко применяемый в системах на платформе SAP, обеспечивает поддержку стандартных форматов и протоколов внешнего и внутреннего взаимодействия.

Система ERP открыта для модификации, соответственно, имеется возможность модернизации существующих стандартных функций и создания новых программных комплексов. Именно это позволило реализовать описанный выше процесс взаимодействия информационных систем в ходе планирования производства и управления запасами на крупном промышленном предприятии с учетом динамичности производственной среды и ограниченности его возможностей. Это стало реальным благодаря переносимости программных модулей и возможности расширения и модификации для использования других форматов и протоколов передачи данных.

Разработанный программный комплекс внедрен на АО ДКС (г. Тверь). Компания производит продукцию для систем электроснабжения, автоматизации и распределения энергии на объектах любого назначения. Реализация проекта позволила оптимизировать производственные показатели и снизить трудозатраты сотрудников различных подразделений, в частности, отдела продуктового маркетинга и планово-экономической службы, систематизировать процесс управления запасами, повысить уровень клиентского сервиса.

Благодаря соответствию системы предъявляемым к ней требованиям переносимости и универсальности относительно протоколов и форматов разработанные в ходе исследования программные модули могут быть адаптированы к работе с другими программными и аппаратными производственными конфигурациями, в том числе существенно меньшего масштаба.

Рассмотрим пример возможной интеграции с системой мониторинга производственных процессов, приведенной в [10]. В силу своей масштабируемости и гибкости данная система может быть внедрена на производствах различного масштаба.

Выделим основные отличия во взаимодействии с данной системой от рассмотренной выше интеграции с использованием платформы SAP:

- основным протоколом передачи данных является WebSocket;
- система имеет свою сигнатуру входящих сообщений;

– отсутствуют платформа SAP и ее ERP-система для реализации модулей на их базе.

Соответственно, для успешной интеграции необходимо решить следующие задачи:

– модификация Interaction Message Sender для работы с WebSocket;

– модификация Interaction Message Converter для преобразования сообщений в формат системы;

– перенос модулей Interaction Message Sender и Interaction Message Converter в рамки Interaction Server для самостоятельного использования.

Поскольку взаимодействие с системой происходит посредством отправки и принятия сообщений и Interaction Message Sender в любом случае устанавливает WebSocket-соединение с системой, в рамках данного модуля могут быть реализованы как отправка, так и получение сообщений, содержащих необходимые данные. Для их обработки необходима реализация отдельного модуля Data Processing Module, которая напрямую зависит от характера и конфигурации производства. Однако для обработки сообщений может быть использован подход, аналогичный используемому в самой системе мониторинга в ее компоненте HandlerHub, а именно совокупность обработчиков на языке JavaScript.

Таким образом, итоговая схема взаимодействия с системой мониторинга будет иметь вид, представленный на рисунке 2.

### Заключение

Поставленная в ходе исследования задача разработки программного продукта, обеспечивающего совместное планирование производ-

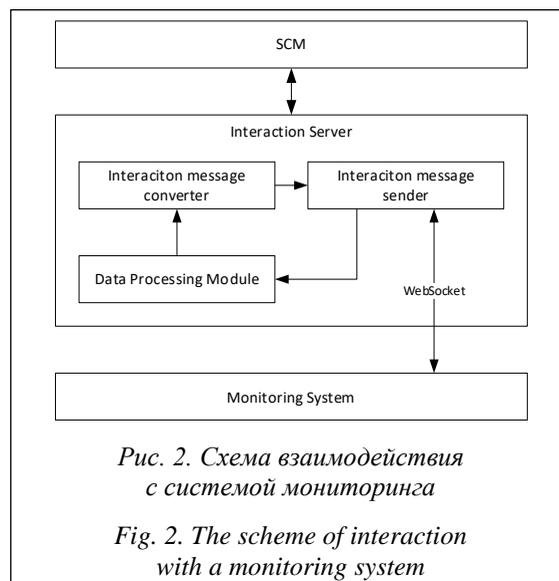


Рис. 2. Схема взаимодействия с системой мониторинга

Fig. 2. The scheme of interaction with a monitoring system

ства и управление запасами, решена в полном объеме. Внедрение разработанных программных модулей позволило на практике подтвердить способность данного решения снизить трудозатраты сотрудников отделов продуктового маркетинга и планово-экономической службы. Показано, что область применения данного подхода не ограничивается крупными предприятиями и может быть скорректирована для использования с соответствующими программными комплексами предприятий меньшего масштаба. Возможна интеграция предложенного программного комплекса с системой мониторинга производственных процессов. Гибкость полученного программного решения открывает возможности для дальнейшего расширения области применения и реализации более масштабных самостоятельных производственных систем и комплексов программ.

### Список литературы

1. Соломаха Г.М., Тулуева В.А. Модель управления запасами на предприятии при изменяющихся стоимости поставки и затратах на хранение // Проблемы управления социально-экономическими системами: теория и практика: матер. VIII Междунар. науч.-практич. конф. 2020. С. 94–99.
2. Новосельцев В.И., Шугай О.Е., Попов Н.Н. Модель управления производственными запасами в условиях неопределенности спроса на выпускаемую продукцию // Вестн. ВИВТ. 2019. № 1. С. 51–52.
3. Слюян С.Х., Новопашин К.С. Управление запасами в корпорации // Наука и бизнес: пути развития. 2019. № 4. С. 172–174.
4. Бабенко И.В. Современные тенденции формирования системы интегрированного управления запасами // Изв. ЮЗГУ. Сер.: Экономика. Социология. Менеджмент. 2019. Т. 9. № 6. С. 135–146.
5. Дирко С.В., Лозко О.И. Автоматизация управления запасами в торговых организациях // Интекс-2021: сб. матер. науч. конф. Т. 6. 2021. С. 62–65.
6. Валиахметова Р.Э. Программа для ЭВМ "INVENTORY MANAGEMENT": Свид. о регистр. ПрЭВМ № 2022665246. Рос. Федерация, 2022.
7. Чистякова Т.Б., Шашихина О.Е. Интеллектуальный программный комплекс моделирования процесса планирования многоассортиментных промышленных производств // Прикладная информатика. 2022. Т. 17. № 5. С. 41–50. doi: 10.37791/2687-0649-2022-17-5-41-50.

8. Арденталева С.И. Программный комплекс планирования производства на малом предприятии // Программные продукты и системы. 2010. Т. 23. № 2. С.127–129.

9. Тимошкина Е.В., Семенова А.Г., Абышева И.Г. Современные программные продукты для оптимизации процесса управления товарно-материальными ресурсами на предприятии // Социальные науки. 2018. № 1. С. 114–123.

10. Соломаха Г.М., Хижняк С.В. Архитектура системы мониторинга производственных процессов в условиях географической распределенности производства // Программные продукты и системы. 2019. Т. 32. № 2. С. 251–257. doi: 10.15827/0236-235X.126.251-257.

Software &amp; Systems

doi: 10.15827/0236-235X.142.459-465

2023, vol. 36, no. 3, pp. 459–465

### Software system for production planning and inventory management

Gennadiy M. Solomakha  
Victoria A. Tulueva  
Stanislav V. Khizhnyak

#### For citation

Solomakha, G.M., Tulueva, V.A., Khizhnyak, S.V. (2023) ‘Software system for production planning and inventory management’, *Software & Systems*, 36(3), pp. 459–465 (in Russ.). doi: 10.15827/0236-235X.142.459-465

#### Article info

Received: 16.02.2023

After revision: 12.03.2023

Accepted: 14.03.2023

**Abstract.** The paper presents the process of interaction between information systems during production planning and inventory management, taking into account the dynamic production environment and the limited capabilities of the enterprise. The paper considers the features of industrial enterprises that do not allow using standard implementations of software products and systems offered on the market. The main module for optimizing the process of planning and inventory management is an ERP system. The Forecast NOW software product of the Russian company “Ingenius Team” LLC helps in sales forecasting and safety stock level estimation; production planning is carried out in the SAP APO SCM system. The result of these systems work and interaction is a production schedule, which is then passed to the MES system for production, as well as accurate data on the needs for purchasing products, raw materials, materials, and semi-finished products. The research methodology uses a system approach, as well as methods of analysis and synthesis of domestic scientists’ studies in the field of planning and inventory management. The paper lists the requirements for software modules developed in process of system integration. It is shown that the proposed software system meets these requirements. The paper presents an example of possible integration of the developed software complex with a system for monitoring production processes, and shows the corresponding integration scheme. It is shown that the application scope of this approach is not limited to large enterprises and can be adjusted for use with appropriate software systems of smaller scale enterprises. The flexibility of the resulting software solution opens up opportunities for further expansion of the application and implementation scope of more large-scale independent production software systems.

**Keywords:** information systems, production planning, inventory management, production inventory, sales forecasting, production monitoring, automation

#### Reference List

1. Solomakha, G.M., Tulueva, V.A. (2020) ‘Enterprise resource management model with variable delivery costs and storage costs’, *Proc. VIII Int. Sci. and Pract. Conf. Problems of Managing Socio-Economic Systems: Theory and Practice*, pp. 94–99.

2. Novoseltsev, V.I., Shugai, O.E., Popov, N.N. (2019) ‘Model of management of production reserves in the conditions of uncertainty of demand for produced products’, *Bull. of the VIHT*, (1), pp. 51–52 (in Russ.).

3. Sloyan, S.Kh., Novopashin, K.S. (2019) ‘Corporate stock management’, *Sci. and Business: Development Ways*, (4), pp. 172–174 (in Russ.).

4. Babenko, I.V. (2019) ‘Current trends in the formation of an integrated inventory management system’, *Proc. SSU. Ser. Economy. Sociology. Management*, 9(6), pp. 135–146 (in Russ.).

5. Dirko, S.V., Lozko, O.I. (2021) ‘Automation of inventory management in trade organizations’, *Proc. Sci. Conf. Intex-2021*, 6, pp. 62–65 (in Russ.).

6. Valiakhmetova, R.E. (2022) *Computer Program "INVENTORY MANAGEMENT"*, Pat RF, № 2022665246.
7. Chistyakova, T.B., Shashikhina, O.E. (2022) 'Intelligent soft package for modeling the planning process of multi-assortment industrial production', *J of Applied Informatics*, 17(5), pp. 41–50 (in Russ.). doi: 10.37791/2687-0649-2022-17-5-41-50.
8. Arendateleva, S.I. (2010) 'Program complex planning production on small business', *Software & Systems*, 23(2), pp. 127–129 (in Russ.).
9. Timoshkina, E.V., Semenova, A.G., Abysheva, I.G. (2018) 'Modern software products for optimizing the process of managing of commodity and material resources at enterprise', *Social-Economic Sci.*, (1), pp. 114–123 (in Russ.).
10. Solomakha, G.M., Khizhnyak, S.V. (2019) 'The architecture of a production processes monitoring system in terms of geographically distributed production', *Software & Systems*, 32(2), pp. 251–257 (in Russ.). doi: 10.15827/0236-235X.126.251-257.

**Авторы****Соломаха Геннадий Михайлович**<sup>1</sup>,

д.ф.-м.н.,

gsolomakha@yandex.ru

**Тулueva Виктория Алексеевна**<sup>1</sup>, аспирант,

viktoriya-kisk@mail.ru

**Хижняк Станислав Виталиевич**<sup>1</sup>, аспирант,

stanislav.khizhnyak@gmail.com

**Authors****Gennadiy M. Solomakha**<sup>1</sup>,

Dr.Sc. (Physics and Mathematics),

gsolomakha@yandex.ru

**Victoria A. Tulueva**<sup>1</sup>, Postgraduate Student,

viktoriya-kisk@mail.ru

**Stanislav V. Khizhnyak**<sup>1</sup>, Postgraduate Student,

stanislav.khizhnyak@gmail.com

<sup>1</sup> Тверской государственный университет,  
г. Тверь, 170100, Россия<sup>1</sup> Tver State University,  
Tver, 170100, Russian Federation

## Разработка чат-бота для отслеживания расписания учебных занятий в университете

В.А. Сазанов  
А.О. Хлобыстова  
М.В. Абрамов

### Ссылка для цитирования

Сазанов В.А., Хлобыстова А.О., Абрамов М.В. Разработка чат-бота для отслеживания расписания учебных занятий в университете // Программные продукты и системы. 2023. Т. 36. № 3. С. 466–473. doi: 10.15827/0236-235X.142.466-473

### Информация о статье

Поступила в редакцию: 07.03.2023

После доработки: 05.04.2023

Принята к публикации: 10.04.2023

**Аннотация.** Высшие учебные заведения нередко сталкиваются с проблемами обеспечения цифровой поддержки учебного процесса, в частности, со своевременным информированием о происходящих в расписании занятий изменениях. Одним из решений может быть Telegram-бот. В данной статье описывается Telegram-бот, разработанный для повышения эффективности оповещения участников учебного процесса о расписании занятий, его изменениях на примере Санкт-Петербургского государственного университета. Методы исследования основаны на выявлении проблем текущего инструмента информирования о расписании, проектировании структуры и последующей разработке клиент-серверного приложения для их решения. Предметом исследования являются инструменты для взаимодействия участников учебного процесса с расписанием учебных занятий. Основным результатом заключается в реализации возможности быстрого отображения расписания с тремя различными вариантами запросов к нему (по названию группы, через поиск преподавателя и путем навигации по всем программам), подписки на уведомления с актуальным расписанием определенной группы или преподавателя с настройкой времени получения уведомлений и просмотра запрошенного расписания в формате текста или сгенерированной на его основе картинки. Кроме того, бот уведомляет об изменении расписания. Практическая значимость заключается в упрощении организационных процессов за счет расширения возможностей своевременного информирования и сокращения времени на получение необходимых сведений. Разработанная система уже используется обучающимися, преподавателями и работниками университета, обеспечивающими учебный процесс.

**Ключевые слова:** Telegram-бот, чат-бот, академическое расписание, информационная система, клиент-серверное приложение

**Благодарности.** Работа выполнена в рамках проекта по государственному заданию Санкт-Петербургского Федерального исследовательского центра РАН СПИИРАН № FFZF-2022-0003, поддержана Санкт-Петербургским государственным университетом, проект № 75254082

С каждым годом информационные технологии все активнее применяются в различных задачах, возникающих в самых разных областях жизни современного общества, в том числе и в сфере высшего образования [1–3]. Одним из элементов организации образовательного процесса в вузе является расписание занятий. При этом немаловажно, как именно обучающиеся получают информацию о расписании, удобство его просмотра и простота в навигации по программам, а также своевременное уведомление участников образовательного процесса о происходящих в расписании изменениях. Все эти факторы так или иначе влияют на качество организации образовательного процесса, закладывают основу для обеспечения более широкого спектра возможностей самореализации граждан.

В большинстве вузов информация о расписании в цифровом формате представляется в виде веб-страницы, что, в частности, не предполагает наличия возможности подписки на

его изменения, уведомления о занятиях, высокой оперативности доступа. В настоящей статье данный процесс рассмотрен на примере *Санкт-Петербургского государственного университета* (СПбГУ). Как правило, обучающиеся могут просмотреть расписание в любое удобное для них время, но при этом возникает ряд сложностей: каждый раз при открытии расписания приходится искать нужный раздел с информацией заново, часто информация выглядит громоздкой из-за дублирования сведений, а различного рода изменения в расписании приходится мониторить самостоятельно, что может обусловить несвоевременность информирования и т.п.

Таким образом, актуально создание дополнительного инструмента для просмотра расписания учебных занятий. В качестве такого инструмента предлагается рассмотреть Telegram-бот. К преимуществам реализации просмотра расписания и уведомления об изменениях через Telegram-бот можно отнести отсутствие

необходимости установки пользователем сторонних приложений, возможность использования как мобильной, так и десктопной версии, удобные инструменты для разработчика.

Целью работы стало повышение оперативности уведомления участников образовательного процесса о расписании учебных занятий за счет создания дополнительной системы – Telegram-бота. Практическая значимость заключается в расширении способов получения сведений об учебном процессе, что, в свою очередь, способствует повышению качества образовательного процесса.

### Релевантные работы

В результате анализа предметной области на наличие схожих концепций найдены работы, описывающие программные разработки в виде Telegram-бота для помощи обучающимся в получении быстрых и точных ответов на вопросы, связанные с расписанием занятий и учебным планом. В частности, в [4, 5] описывается созданный Telegram-бот с возможностью просмотра расписания преподавателя, группы или аудитории по данным, извлекаемым из web-страницы с официальным расписанием университета. Недостатком данного решения является существенная зависимость от структуры сайта. В [6] предлагается бот с использованием моделей глубокого обучения: именно бот классифицирует сообщения пользователя (относительно расписания учебных занятий) и выдает нужный ответ. Это решение имеет практическую значимость, однако повторение его реализации возможно только при наличии набора данных, содержащего широкий спектр вопросов от студентов и преподавателей. В публикациях [7, 8] описываются чат-боты, близкие к цели авторов настоящей статьи, но они предназначены для сопровождения занятий только по одной определенной дисциплине и включают возможность не только просмотра расписания, но и размещения учебных материалов. Вместе с тем в результате данных исследований была подтверждена гипотеза о целесообразности использования чат-ботов для сопровождения образовательного процесса. Положительный эффект от использования чат-ботов в академической среде был отмечен и в исследованиях [9–11].

Для проектирования Telegram-бота были изучены работы, посвященные анализу требований к чат-ботам и оценке удобства их использования [12, 13], вопросам корректности

отображаемой информации [14] и легкого доступа к ней [15].

### Постановка задачи

Предлагаемая разработка направлена на создание Telegram-бота – дополнительной системы для повышения оперативности уведомления участников образовательного процесса об изменениях в расписании учебных занятий. Задача состояла в разработке инструмента, реализующего следующую функциональность:

- выбор расписания группы по названию или при помощи навигации по всем программам;
- выбор расписания преподавателя по его фамилии;
- подписка на расписание определенной группы или преподавателя для быстрого доступа к расписанию и получения ежедневных уведомлений с информацией о предстоящих занятиях;
- настройка уведомлений, содержащих актуальное расписание, с возможностью выбора времени и формата отображения (текстовое сообщение или изображение).

### Программная реализация

Разработка Telegram-бота ([https://t.me/timetable\\_SPBU\\_bot](https://t.me/timetable_SPBU_bot)) велась на языке программирования Python с применением расширений Poetry, AIogram, Pylint, библиотек Pydantic, Cashews, SQLAlchemy, Gino, Babel, APScheduler, Jinja2, Pyppeteer, Pdoc, а также инструментов GitHub Actions и Docker Compose. Для хранения информации о состоянии пользователей использована система Redis, для хранения долговременной информации о пользователях и расписании – PostgreSQL.

**Серверная часть.** Источником информации для наполнения бота послужило официальное API сайта [timetable.spbu.ru](https://timetable.spbu.ru) (<https://timetable.spbu.ru/help>), в котором реализованы функции StudyDivisions (для получения информации об учебных подразделениях) и Programs (для получения информации о реализуемых образовательных программах).

Для реализации поиска расписания группы по названию в БД собирается информация о всех студенческих группах путем последовательного вызова методов StudyDivisions и Programs:

- вызов метода StudyDivisions (без параметров) для получения списка сокращенных названий (alias) всех учебных подразделений;

– циклический вызов метода StudyDivisions (с параметром alias) для получения идентификаторов образовательных программ (program id) каждого учебного подразделения;

– циклический вызов метода Programs (с параметром program\_id) для получения названий и идентификаторов групп, соответствующих каждой из программ.

Поиск расписания преподавателя реализован через отправку GET-запроса на адрес <https://timetable.spbu.ru/api/v1/educators/search/{query}>, где query – фамилия преподавателя (или ее часть), и возвращает список преподавателей, удовлетворяющих параметру запроса.

**БД системы.** Для повышения скорости отправки нужного расписания конечному пользователю и снижения нагрузки на сервер расписания было решено сохранять необходимую информацию в своей БД, а не обращаться постоянно к серверу с запросами. Для этого при первом запросе собирается информация о расписании, после чего с заданной периодичностью осуществляется ее актуализация. На рисунке 1 изображена структура БД, в которой содержится базовая информация об основном расписании пользователя.

**Генерация изображения для удобства просмотра расписания.** При решении задачи генерации изображения была выявлена необходимость настройки его размера в зависимости от количества занятий. Представим этапы разработанного алгоритма генерации отображения расписания.

0. Преобразование ответа от API timetable в объект класса Schedule.

1. Заполнение разработанных html-шаблонов информацией о расписании, используя Jinja2.

2. При помощи библиотеки Puppeteer происходят

2.1. запуск (если это не было сделано ранее) браузера, установленного в Docker;

2.2. открытие в новой вкладке html-файла, сгенерированного на шаге 1, с примененными css-стилями;

2.3. создание скриншота всей страницы и сохранение его в формате jpeg.

Таким образом, задача генерации изображения с расписанием была сведена к разработке html-шаблонов и написанию css-стилей для их внешнего оформления (см. <http://www.swsys.ru/uploaded/image/2023-3/2023-3-dop/31.jpg>).

Для улучшения визуального восприятия информации были разработаны функции по уда-

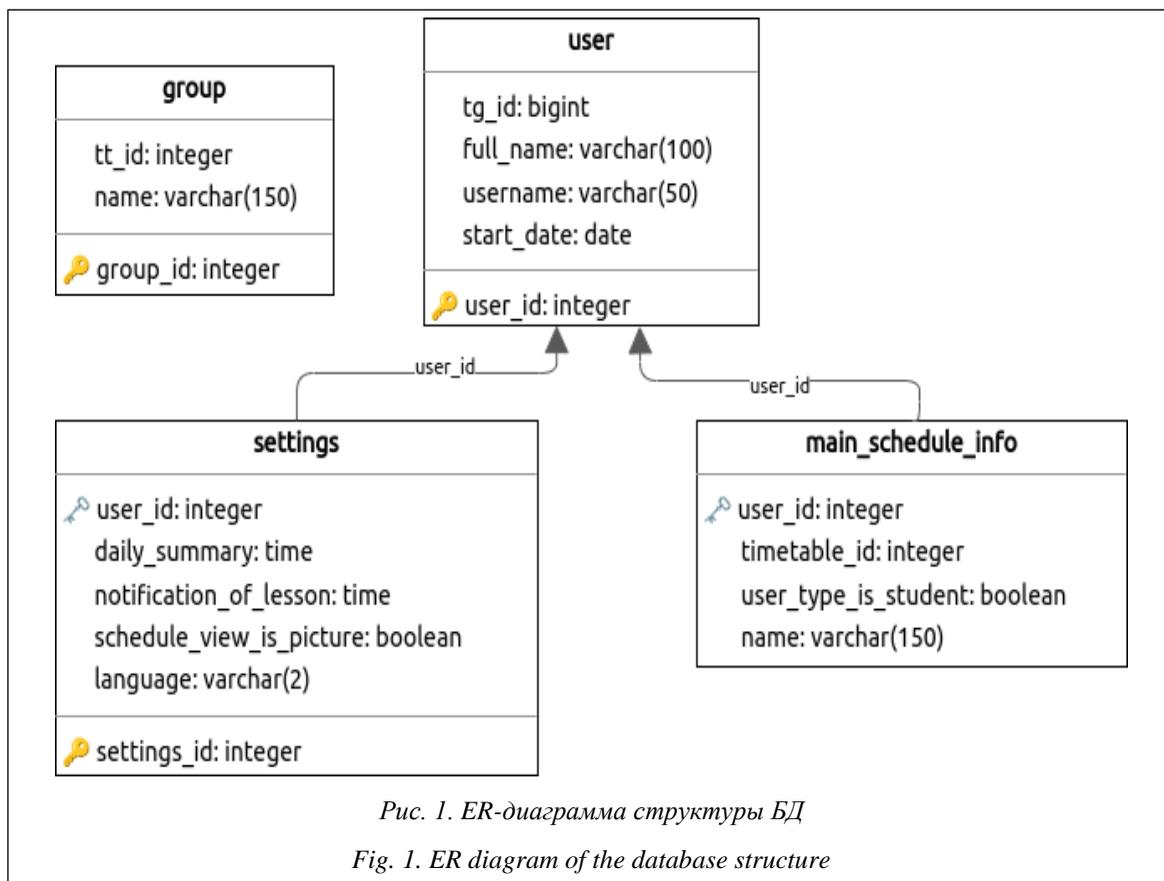


Рис. 1. ER-диаграмма структуры БД

Fig. 1. ER diagram of the database structure

лению повторяющейся информации: время проведения занятия и название предмета.

**Обработка действий пользователя.** Взаимодействие с пользователем основано на двух типах действий-обновлений: получение нового входящего сообщения и входящий запрос обратного вызова от кнопки на встроенной клавиатуре, для которых в ходе разработки бота были написаны функции-обработчики. Реализовано следующее: приветствие пользователя с предложением способов получения расписания (по названию группы, через навигацию по программам, по Ф.И.О. преподавателя), уточнение выбора, отображение запрашиваемого расписания, предложение сделать расписание основным (с последующей возможностью настроить сообщения-уведомления) / только посмотреть, настройка уведомлений (для расписания, указанного в качестве основного (рис. 2)), настройка языка расписания, настройка отображения – текст или картинка.

Чтобы после обновлений и перезапуска бота состояния пользователей не исчезали,

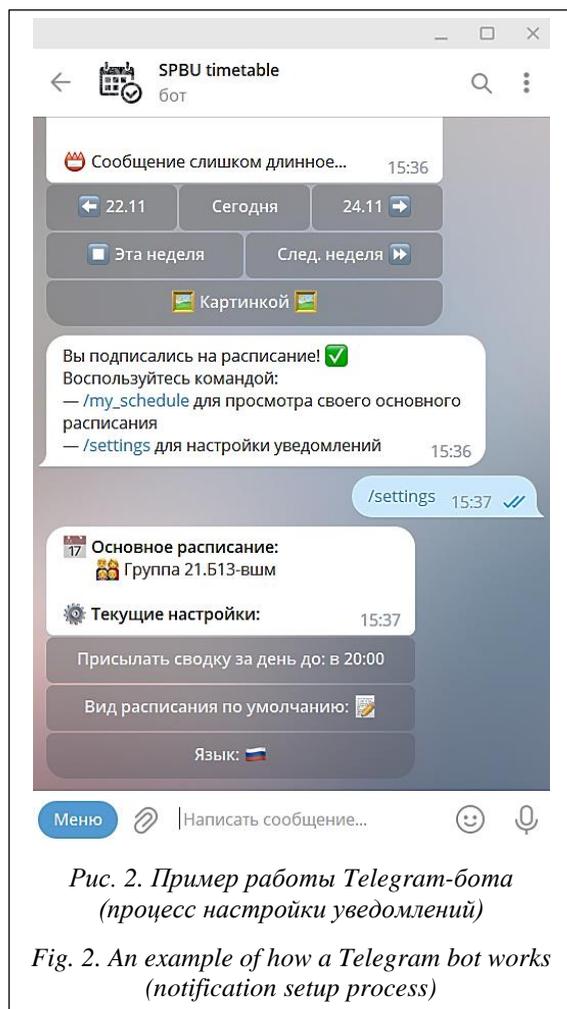


Рис. 2. Пример работы Telegram-бота (процесс настройки уведомлений)

Fig. 2. An example of how a Telegram bot works (notification setup process)

было организовано хранение временной информации в БД на основе системы Redis.

**Генерация документации для разработчиков.** Для дальнейшего сопровождения бота была создана техническая документация при помощи docstrings – строкового литерала, указываемого в исходном коде для документирования его определенного фрагмента. В строках документации описываются назначение объекта программы, смысл входных и выходных параметров – для функции/процедуры, примеры использования, возможные исключительные ситуации, особенности реализации (<http://www.swsys.ru/uploaded/image/2023-3/2023-3-dop/12.jpg>). Формирование документации из docstrings осуществляется при помощи инструмента Pdoc.

**Разработка системы сбора статистики использования бота.** С целью дальнейшего развития Telegram-бота была реализована команда /statistics, доступная только администраторам бота, с помощью которой можно получить статистику о пользователях: никнейм в Telegram, дата начала пользования ботом, расписание, на которое подписан пользователь, настройки уведомлений.

## Выводы

В работе описано разработанное клиент-серверное приложение – Telegram-бот, призванное устранить недостатки существующего способа получения сведений о графике проводимых занятий. Основная проблема заключалась в необходимости использовать браузер и каждый раз заново искать нужное расписание через навигацию по программам или по фамилии преподавателя. Решением стало создание дополнительного инструмента, встраиваемого в популярный мессенджер Telegram и позволяющего избежать повторяющихся действий путем подписки на определенное расписание. Разработанное решение позволяет находить нужное расписание по номеру группы, что, в свою очередь, существенно сократит время на его поиски. Кроме того, в работе были предложены подходы к решению проблемы громоздкости расписания при помощи удаления дублирующейся информации, а также добавления способа отображения расписания в виде сгенерированной картинка.

В качестве дальнейших направлений исследований могут быть рассмотрены задачи по внедрению методов искусственного интеллекта, а именно классификации сообщений

пользователей и выдачи нужной информации на их основе [16–18], а также прогнозирования поведения пользователей по сведениям об их взаимодействии с Telegram-ботом [19–21]. Кроме того, с целью адаптации предлагаемого решения под расписания других университетов планируется рассмотреть вопросы веб-скрейпинга [22–24].

### Заключение

Таким образом, в статье был описан процесс разработки Telegram-бота – дополнительной системы, созданной с целью повышения оперативности уведомления участников образовательного процесса о расписании учебных занятий. В работе описаны структура предлагаемого решения и особенности его реализации.

В чат-боте существует возможность выбора расписания группы по названию или при помощи навигации по всем программам либо просмотра расписания преподавателя; кроме просмотра расписания, для быстрого доступа и получения ежедневных уведомлений с информацией о предстоящих занятиях пользователь может подписаться на него, настроить уведомления и вид отображения сведений.

Проведена апробация бота для СПбГУ, на него подписались уже более 400 обучающихся и преподавателей. Полученные результаты имеют практическую значимость для упрощения организационных процессов, сопровождающих основную цель образования – получение новых знаний, навыков и компетенций, способствуя сокращению времени получения актуального расписания занятий.

### Список литературы

1. Fährndrich J. A literature review on the impact of digitalisation on management control. *J. of Management Control*, 2022, pp. 1–57. doi: 10.1007/s00187-022-00349-4.
2. Gupta R., Seetharaman A., Maddulety K. Critical success factors influencing the adoption of digitalisation for teaching and learning by business schools. *Education and Inform. Tech.*, 2020, vol. 25, no. 5, pp. 3481–3502. doi: 10.1007/s10639-020-10246-9.
3. Xie P., Zhang H., Xu W., Hu Y. The Construction of a smart campus model based on big data. *Proc. 6th ICSGEA*, 2021, pp. 536–539. doi: 10.1109/ICSGEA53208.2021.00127.
4. Шипунов А.В., Жилина Е.В. Разработка Telegram-бота "Rsuessedulebot" // Информатизация в цифровой экономике. 2021. Т. 2. № 3. С. 113–124. doi: 10.18334/ide.2.3.113390.
5. Priadko A.O., Osadcha K.P., Kruhlyk V.S., Rakovych V.A. Development of a chatbot for informing students of the schedule. *CEUR Workshop Proc. Proc. 2nd Student Workshop on Comput. Sci. & Software Engineering*, 2019, vol. 2546, pp. 128–137.
6. Dinesh T., Anala M.R., Newton T.T., Smitha G.R. AI Bot for academic schedules using Rasa. *Proc. ICSES*, 2021, pp. 1–6. doi: 10.1109/ICSES52305.2021.9633799.
7. Khalil M., Ramech M. Eduino: A telegram learning-based platform and chatbot in higher education. In: *LNCS. Proc. HCII*, 2022, pp. 188–204. doi: 10.1007/978-3-031-05675-8\_15.
8. Nosenko O., Nosenko Y., Shevchuk R. Telegram messenger for supporting educational process under the conditions of quarantine restrictions. In: *CCIS. Proc. ICTERI*, 2022, pp. 308–319. doi: 10.1007/978-3-031-14841-5\_20.
9. Abbas N., Pickard T., Atwell E., Walker A. University student surveys using chatbots: artificial intelligence conversational agents. In: *LNISA. Proc. HCII*, 2021, pp. 155–169. doi: 10.1007/978-3-030-77943-6\_10.
10. Mora A.M., Guillén A., Barranco F., Castillo P.A., Merelo J.J. Studying how to apply chatbots technology in higher-education: first results and future strategies. In: *LNISA. Proc. HCII*, 2021, pp. 185–198. doi: 10.1007/978-3-030-77943-6\_12.
11. Tsivitanidou O., Ioannou A. Envisioned pedagogical uses of chatbots in higher education and perceived benefits and challenges. In: *LNISA. Proc. HCII*, 2021, pp. 230–250. doi: 10.1007/978-3-030-77943-6\_15.
12. Mafra M., Nunes K., Castro A., Lopes A. et al. Defining requirements for the development of useful and usable chatbots: An analysis of quality attributes from academy and industry. In: *LNCS. Proc. HCII*, 2022, pp. 479–493. doi: 10.1007/978-3-031-05412-9\_33.
13. Plantak Vukovac D., Horvat A., Čizmešija A. Usability and user experience of a chat application with integrated educational chatbot functionalities. In: *LNISA. Proc. HCII*, 2021, pp. 216–229. doi: 10.1007/978-3-030-77943-6\_14.
14. Abdou N., Karimi A., Murarka R., Swarat S. Rigorous data validation for accurate dashboards: Experience from a higher education institution. *IT Professional*, 2021, vol. 23, no. 3, pp. 95–101. doi: 10.1109/MITP.2021.3073799.
15. König C.M., Karrenbauer C., Breitner M.H. Critical success factors and challenges for individual digital study assistants in higher education: A mixed methods analysis. *Educ. and Inform. Tech.*, 2022, vol. 28, no. 4, pp. 4475–4503. doi: 10.1007/s10639-022-11394-w.
16. Sáiz-Manzanares M.C., Marticorena-Sánchez R., Martín-Antón L.J., Díez I.G., Almeida L. Perceived satisfaction of university students with the use of chatbots as a tool for self-regulated learning. *Heliyon*, 2023, vol. 9, no. 1, art. e12843. doi: 10.1016/j.heliyon.2023.e12843.
17. Peyton K., Unnikrishnan S. A comparison of chatbot platforms with the state-of-the-art sentence BERT for answering online student FAQs. *Results in Engineering*, 2023, vol. 17, art. 100856. doi: 10.1016/j.rineng.2022.100856.

18. Oliseenko V.D., Tulupyeva T.V., Abramov M.V. Online social network post classification: a multiclass approach. In: LNNS. Proc. ИТИ, 2021, pp. 207–215. doi: 10.1007/978-3-030-87178-9\_21.
19. Stoliarova V.F., Tulupyeu A.L. Regression model for the problem of parameter estimation in the gamma Poisson model of behavior: an application to the online social media posting data. Proc. XXIV SCM, 2021, pp. 24–27. doi: 10.1109/SCM52931.2021.9507187.
20. Столярова В.Ф., Тулупьев А.Л. Регрессия кокса в задаче оценки параметров рискообразующего поведения индивида по данным о последних эпизодах // ИТВ СПбГПУ. Физико-математ. науки. 2021. Т. 14. № 4. С. 202–217 (in Eng.).
21. Frolova M.S., Korepanova A.A., Abramov M.V. Assessing the degree of the social media user's openness using an expert model based on the Bayesian network. Proc. XXIV SCM, 2021, pp. 52–55. doi: 10.1109/SCM52931.2021.9507111.
22. Корепанова А.А., Бушмелев Ф.В., Сабреков А.А. Технологии парсинга на Node.js в задаче агрегации сведений и оценки параметров грузовых маршрутов посредством извлечения данных из открытых источников // Компьютерные инструменты в образовании. 2021. № 3. С. 41–56.
23. Patnaik S.K., Babu C.N., Bhave M. Intelligent and adaptive web data extraction system using convolutional and long short-term memory deep learning networks. Big Data Mining and Analytics, 2021, vol. 4, no. 4, pp. 279–297. doi: 10.26599/BDMA.2021.9020012.
24. Bale A.S., Ghorpade N., Rohith S., Kamalesh S., Rohith R., Rohan B.S. Web scraping approaches and their performance on modern websites. Proc. ICESc, 2022, pp. 956–959. doi: 10.1109/ICESc54411.2022.9885689.

### Developing a chatbot to monitor a university's academic timetable

Vadim A. Sazanov  
Anastasia O. Khlobystova  
Maxim V. Abramov

#### For citation

Sazanov, V.A., Khlobystova, A.O., Abramov, M.V. (2023) 'Developing a chatbot to monitor a university's academic timetable', *Software & Systems*, 36(3), pp. 466–473 (in Russ.). doi: 10.15827/0236-235X.142.466-473

#### Article info

Received: 07.03.2023

After revision: 05.04.2023

Accepted: 10.04.2023

**Abstract.** Higher education institutions often face problems in providing digital support for the educational process, in particular, the problems associated with timely informing about timetable changes. One of the solutions to this problem can be a Telegram bot. This article describes a Telegram bot designed to improve the efficiency of notifying participants about the class schedule and its changes using the example of St. Petersburg State University (SPbGU). The research methods are based on identifying the problems of the current schedule information tool, designing a structure, and then developing a client-server application to solve them. The research subject is the tools for the interaction of participants in the educational process with an academic timetable. The main result is the implementation of the ability to quickly display the timetable with three different querying options (by a group name, by searching for a teacher and by navigating through all programs), subscribing to notifications with the current timetable of a specific group or teacher with setting the time for

receiving notifications and the ability to view the requested schedule in text format or a generated image. In addition, the bot notifies a user when the timetable changes. The practical significance is in the simplification of organizational processes by expanding the possibilities of timely information and reducing the time to obtain the necessary information. The students, teachers and employees of the St. Petersburg State University, who provide the educational process, already use the developed system in their activities.

**Keywords:** Telegram bot, Chatbot, academic timetable, information system, client-server application

**Acknowledgements.** This work was carried out within the framework of the project under the state assignment of SPC RAS SPIIRAS FFZF-2022-0003, with the financial support of. St. Petersburg State University, project № 75254082

### Reference List

1. Fährndrich, J. (2022) 'A literature review on the impact of digitalisation on management control', *J. of Management Control*, pp. 1–57. doi: 10.1007/s00187-022-00349-4.
2. Gupta, R., Seetharaman, A., Maddulety, K. (2020) 'Critical success factors influencing the adoption of digitalisation for teaching and learning by business schools', *Education and Inform. Tech.*, 25(5), pp. 3481–3502. doi: 10.1007/s10639-020-10246-9.
3. Xie, P., Zhang, H., Xu, W., Hu, Y. (2021) 'The Construction of a smart campus model based on big data', *Proc. 6th ICSGEA*, pp. 536–539. doi: 10.1109/ICSGEA53208.2021.00127.
4. Shipunov, A.V., Zhilina, E.V. (2021) 'Development of the Telegram bot "Rsueschedulebot"', *Informatization in Digital Economy*, 2(3), pp. 113–124 (in Russ.). doi: 10.18334/ide.2.3.113390
5. Priadko, A.O., Osadcha, K.P., Kruhlyk, V.S., Rakovych, V.A. (2019) 'Development of a chatbot for informing students of the schedule', *CEUR Workshop Proc. Proc. 2nd Student Workshop on Comput. Sci. & Software Engineering*, 2546, pp. 128–137.
6. Dinesh, T., Anala, M.R., Newton, T.T., Smitha, G.R. (2021) 'AI Bot for academic schedules using Rasa', *Proc. ICSES*, pp. 1–6. doi: 10.1109/ICSES52305.2021.9633799.
7. Khalil, M., Rambech, M. (2022) 'Eduino: A telegram learning-based platform and chatbot in higher education', in *LNCS. Proc. HCII*, pp. 188–204. doi: 10.1007/978-3-031-05675-8\_15.
8. Nosenko, O., Nosenko, Y., Shevchuk, R. (2022) 'Telegram messenger for supporting educational process under the conditions of quarantine restrictions', in *CCIS. Proc. ICTERI*, pp. 308–319. doi: 10.1007/978-3-031-14841-5\_20.
9. Abbas, N., Pickard, T., Atwell, E., Walker, A. (2021) 'University student surveys using chatbots: artificial intelligence conversational agents', in *LNISA. Proc. HCII*, pp. 155–169. doi: 10.1007/978-3-030-77943-6\_10.
10. Mora, A.M., Guillén, A., Barranco, F., Castillo, P.A., Merelo, J.J. (2021) 'Studying how to apply chatbots technology in higher-education: first results and future strategies', in *LNISA. Proc. HCII*, pp. 185–198. doi: 10.1007/978-3-030-77943-6\_12.
11. Tsivitanidou, O., Ioannou, A. (2021) 'Envisioned pedagogical uses of chatbots in higher education and perceived benefits and challenges', in *LNISA. Proc. HCII*, pp. 230–250. doi: 10.1007/978-3-030-77943-6\_15.
12. Mafra, M., Nunes, K., Castro, A., Lopes, A. et al. (2022) 'Defining requirements for the development of useful and usable chatbots: An analysis of quality attributes from academy and industry', in *LNCS. Proc. HCII*, pp. 479–493. doi: 10.1007/978-3-031-05412-9\_33.
13. Plantak Vukovac, D., Horvat, A., Čizmešija, A. (2021) 'Usability and user experience of a chat application with integrated educational chatbot functionalities', in *LNISA. Proc. HCII*, pp. 216–229. doi: 10.1007/978-3-030-77943-6\_14.
14. Abdou, N., Karimi, A., Murarka, R., Swarat, S. (2021) 'Rigorous data validation for accurate dashboards: Experience from a higher education institution', *IT Professional*, 23(3), pp. 95–101. doi: 10.1109/MITP.2021.3073799.
15. König, C.M., Karrenbauer, C., Breitner, M.H. (2022) 'Critical success factors and challenges for individual digital study assistants in higher education: A mixed methods analysis', *Educ. and Inform. Tech.*, 28(4), pp. 4475–4503. doi: 10.1007/s10639-022-11394-w.
16. Sáiz-Manzanares, M.C., Marticorena-Sánchez, R., Martín-Antón, L.J., Díez, I.G., Almeida, L. (2023) 'Perceived satisfaction of university students with the use of chatbots as a tool for self-regulated learning', *Heliyon*, 9(1), art. e12843. doi: 10.1016/j.heliyon.2023.e12843.
17. Peyton, K., Unnikrishnan, S. (2023) 'A comparison of chatbot platforms with the state-of-the-art sentence BERT for answering online student FAQs', *Results in Engineering*, 17, art. 100856. doi: 10.1016/j.rineng.2022.100856.
18. Oliseenko, V.D., Tulupyeva, T.V., Abramov, M.V. (2021) 'Online social network post classification: a multiclass approach', in *LNNS. Proc. ITI*, pp. 207–215. doi: 10.1007/978-3-030-87178-9\_21.
19. Stoliarova, V.F., Tulupyev, A.L. (2021) 'Regression model for the problem of parameter estimation in the gamma Poisson model of behavior: an application to the online social media posting data', *Proc. XXIV SCM*, pp. 24–27. doi: 10.1109/SCM52931.2021.9507187.
20. Stoliarova, V.F., Tulupyev, A.L. (2021) 'Cox regression in the problem of risky behavior parameter estimation based on the last episodes' data', *St. Petersburg State Polytechnical University J. Physics and Math.*, 14(4), pp. 202–217.
21. Frolova, M.S., Korepanova, A.A., Abramov, M.V. (2021) 'Assessing the degree of the social media user's openness using an expert model based on the Bayesian network', *Proc. XXIV SCM*, pp. 52–55. doi: 10.1109/SCM52931.2021.9507111.
22. Korepanova, A.A., Bushmelev, F.V., Sabrekov, A.A. (2021) 'Node.js parsing technologies in the task of aggregating information and evaluating the parameters of cargo routes by extracting data from open sources', *Comput. Tools in Education*, (3), pp. 41–56 (in Russ.).

23. Patnaik, S.K., Babu, C.N., Bhave, M. (2021) 'Intelligent and adaptive web data extraction system using convolutional and long short-term memory deep learning networks', *Big Data Mining and Analytics*, 4(4), pp. 279–297. doi: 10.26599/BDMA.2021.9020012.

24. Bale, A.S., Ghorpade, N., Rohith, S., Kamalesh, S., Rohith, R., Rohan, B.S. (2022) 'Web scraping approaches and their performance on modern websites', *Proc. ICESC*, pp. 956–959. doi: 10.1109/ICESC54411.2022.9885689.

#### Авторы

**Сазанов Вадим Алексеевич**<sup>1</sup>, студент,  
mail@dscs.pro

**Хлобыстова Анастасия Олеговна**<sup>1,2</sup>, м.н.с.  
лаборатории теоретических и междисциплинарных  
проблем информатики, aok@dscs.pro

**Абрамов Максим Викторович**<sup>2</sup>, к.т.н., руководитель  
лаборатории теоретических и междисциплинарных  
проблем информатики, mva@dscs.pro

<sup>1</sup> Санкт-Петербургский государственный университет,  
г. Санкт-Петербург, 199034, Россия

<sup>2</sup> Санкт-Петербургский Федеральный  
исследовательский центр РАН,  
г. Санкт-Петербург, 199178, Россия

#### Authors

**Vadim A. Sazanov**<sup>1</sup>, Student,  
mail@dscs.pro

**Anastasia O. Khlobystova**<sup>1,2</sup>, Junior Researcher  
Laboratory of Theoretical and Interdisciplinary  
Problems of Informatics, aok@dscs.pro

**Maxim V. Abramov**<sup>2</sup>, Ph.D. (Engineering), Head of  
Laboratory of Theoretical and Interdisciplinary  
Problems of Informatics, mva@dscs.pro

<sup>1</sup> St. Petersburg State University,  
St. Petersburg, 199034, Russian Federation

<sup>2</sup> St. Petersburg Federal Research Center  
of the Russian Academy of Sciences,  
St. Petersburg, 199178, Russian Federation

## Программное обеспечение для автоматизированного обнаружения и оценки разрушений соединительных швов зданий

В.В. Кабанова  
О.С. Логунова  
М.Ю. Наркевич  
В.Д. Корниенко

### Ссылка для цитирования

Кабанова В.В., Логунова О.С., Наркевич М.Ю., Корниенко В.Д. Программное обеспечение для автоматизированного обнаружения и оценки разрушений соединительных швов зданий // Программные продукты и системы. 2023. Т. 36. № 3. С. 474–485. doi: 10.15827/0236-235X.142.474-485

### Информация о статье

Поступила в редакцию: 20.01.23

После доработки: 14.04.23

Принята к публикации: 17.05.2023

**Аннотация.** В статье предложен метод обнаружения и оценки разрушений швов зданий, основанный на изменении контраста изображения, адаптивном пороге, алгоритме Хафа и нескольких дополняющих его алгоритмах обработки данных. Для исследования состояния фасадов здания промышленного предприятия осуществлена фото- и видеосъемка с помощью беспилотного летательного аппарата. Полученная коллекция изображений использовалась для отладки работы созданного программного модуля и метода обнаружения швов зданий и разрушений на них. Предложенный метод реализован в программном модуле и позволяет обрабатывать изображения разных размера и яркости пикселей. Предусмотрено применение данного метода с исключенными на изображении областями. В работе рассмотрены текущие способы визуального обследования здания, приведены примеры существующего ПО для контроля состояния объектов. Описан порядок выполнения исследования, представлены схемы работы реализованных в программном модуле процедур по обнаружению швов и их разрушений. Рассмотрены особенности работы метода, предложенного авторами данной статьи. Приведен пример градации текущих и прогнозируемых разрушений швов здания. Группа параметров, используемая в методе, подобрана в ходе исследования, изменение их значений позволяет эксперту корректировать обнаружение швов. Данная разработка позволила оценить состояние исследуемых зданий и сооружений. Приведены примеры изображений с выделенными швами и разрушениями, а также результаты процентных оценок разрушений соединительных швов зданий на изображении. Результаты исследования подтверждают работоспособность программного модуля и необходимость внедрения таких модулей для контроля состояния территорий и зданий предприятия.

**Ключевые слова:** распознавание образов, программный модуль, мониторинг состояния здания, оценка разрушений, алгоритм Хафа

**Благодарности.** Работа выполняется в МГТУ им. Г.И. Носова по договору № 247715 от 05.07.2021 г. «Разработка и применение методик контроля территорий, зданий и сооружений ПАО «ММК» с использованием беспилотных воздушных судов (БВС)»

Для бесперебойной работы предприятия требуется следить не только за технологиями производства и порядком выполнения работ, но и за состоянием сооружений, зданий, различных установок.

Целью данного исследования является разработка программного модуля по автоматизированному обнаружению и оценке разрушений соединительных швов панельных зданий. Разработанное ПО, в первую очередь, ориентировано на предприятия и компании, имеющие в материальном активе большое количество зданий и сооружений.

В соответствии с целью поставлены следующие задачи:

- определить способы оценки технического состояния зданий и сооружений;
- выполнить оценку существующего ПО, предназначенного для контроля за состоянием объектов промышленного предприятия;

- описать исходные и выходные данные, необходимые для работы программного модуля;

- разработать алгоритмы и схему работы программного модуля по обнаружению и оценке разрушений соединительных швов панельных зданий;

- разработать программную реализацию модуля для внедрения в систему экспертной оценки при принятии решений о техническом состоянии панельных зданий.

Авторами данной статьи предлагается методика обнаружения разрушений межпанельных швов здания, основанная на композиции алгоритмов изменения контраста изображения, адаптивного порога, яркости объекта разрушения, алгоритма Хафа и на нескольких других дополняющих его алгоритмах обработки данных.

## Методы

**Способы оценки состояния зданий и сооружений.** Техническое обследование конструкций зданий и сооружений проводится в несколько этапов. На первом этапе осуществляется предварительное обследование конструкций сооружения, включающее осмотр здания, сбор общих сведений о сооружении, изучение ранее проводившихся обследований его строительных конструкций. На втором этапе выполняются инструментальное обследование зданий с визуальным обследованием конструкций и фотофиксацией видимых дефектов, а также обмерные работы. Инструментальное обследование сооружения назначается в случае недостаточности результатов визуального обследования для решения задач. Третий этап проводится в лабораторных условиях, определяются физико-технические характеристики материалов обследуемых конструкций. На четвертом этапе результаты обследования обобщаются, составляются технологический отчет и заключение [1].

В настоящее время в большинстве случаев оценкой технического состояния зданий занимаются эксперты и специалисты из строительной области, имеющие необходимую аппаратуру для замеров. Такой способ является затратным с точки зрения необходимых человеко-часов для обследования здания, особенно, если объекты расположены на большой площади. Кроме того, после визуального обследования и замеров требуется обработка полученных данных, а это длительный процесс. Результат работы зависит от группы экспертов, их навыков, физического состояния и опыта.

С развитием компьютерного зрения появилась возможность автоматизации процесса визуального контроля за состоянием зданий и сооружений. Для обнаружения трещин на бетонных и других конструкциях используются сверточные [2–4], а также генеративно-состязательные [5] нейронные сети и методы обработки изображений (алгоритм Canny [6], бинаризация Otsu [7, 8], удаление шума [9] и др.).

В работе [10] предлагается метод обнаружения трещин дорожного покрытия с использованием алгоритмов предварительной обработки (выравнивание гистограммы) и адаптивного порога. Авторы протестировали метод на 509 изображениях и получили результаты с точностью 0,82. В статье [11] описаны исследования с применением метода автоматического обнаружения трещин, основанного на совмещении

изображений и сегментации на уровне пикселей (улучшенный DeepLab\_v3+). Утверждается, что метод значительно эффективнее других распространенных алгоритмов сегментации на уровне пикселей.

Внедрение прикладной цифровой платформы для контроля за состоянием объектов на металлургическом предприятии позволило сделать вывод, что использование платформы, состоящей из программных модулей, эффективно при экспертной оценке элементов производственных объектов [12, 13].

Существует трехэтапный метод автоматического обнаружения микротрещин в бетоне подводных сооружений (цветовая коррекция изображений, отправка в сверточную нейронную сеть для распознавания трещин, сегментация трещин с помощью алгоритма Otsu) с экспериментальным результатом в 93,9 % [14].

Описанное ПО позволяет быстро провести визуальное обследование объектов (по сравнению с ручным методом) и получить результат в виде изображения с обнаруженными распознанными дефектами и процентной оценкой. На основе результата делается вывод о необходимости дополнительных инструментальных замеров, а также лабораторных исследований с применением аппаратуры. Алгоритмы обработки изображений также имеют погрешность, поэтому исследователю необходимо корректировать процесс выполнения обработки, например, через изменение параметров, использующихся в алгоритме, для улучшения результата исследования.

**Описание исходных и выходных данных для работы программного модуля.** Для проведения исследования по обнаружению разрушений на швах здания проведена фото- и видеосъемка объектов на территории Магнитогорского металлургического комбината с использованием *беспилотного летательного аппарата* (БПЛА) [15]. Также допускается съемка на фотоаппарат с высоким разрешением для захвата детализированных изображений. Если исходными данными являются видеопотоки, то перед анализом изображений из них извлекаются кадры, которые затем загружаются в программный модуль. Во время съемки БПЛА или фотоаппарат следует располагать параллельно объекту исследования, то есть запрещается съемка под углом. Несоблюдение данного правила приводит к неверному обнаружению соединительных швов здания на изображении при применении серии алгоритмов, представленных в данной работе.

Для удобной и быстрой работы с программным модулем по обнаружению и оценке разрушений соединительных швов здания авторами данной статьи спроектировано и реализовано приложение с графическим интерфейсом пользователя. Однако поскольку модуль создан для использования в общей системе (веб-приложение), имеющей клиентскую часть и включающей в себя группу модулей, предусмотрена работа с модулем через командную строку посредством запуска скрипта и передачи требуемых аргументов и опций для возможности автоматического запуска.

При работе с модулем через командную строку требуется указать обязательные аргументы и опции: путь к скрипту (.sh), запускающему программный модуль, путь к исходному изображению здания и путь к директории для сохранения обработанных изображений и файла с результатом оценки разрушений в формате JSON. Кроме того, предусмотрена передача параметров для корректировки результата и исключения из области исследования фрагментов изображения. В результате выполнения скрипта в папку сохраняются обработанные изображения с постфиксами (имя изображения\_allSeams.png, имя изображения\_DamageSeams.png) для их дальнейшей идентификации.

Программный модуль обрабатывает изображения с различными разрешениями и размерами. От размера изображения зависит время его обработки. Допускается загрузка следующих видов изображений в формате png или jpg: фрагменты панелей здания, фасад здания, склейка фрагментов здания.

**Алгоритмы и схемы работы программного модуля по обнаружению и оценке разрушений соединительных швов панельных зданий.** В программном модуле реализовано несколько процедур. Основу модуля составляют процедура обнаружения и подсчета площади всех соединительных швов здания на изображении, а также площади разрушений соединительных швов здания на изображении.

Для реализации программного кода каждой процедуры составлены алгоритмы, включающие применение функций из библиотеки OpenCV и использование дополнительных алгоритмов, необходимых для обработки данных после работы стандартных функций. На рисунке 1 представлена блок-схема алгоритма поиска площади швов и разрушений соединительных швов между панелями здания.

Рассмотрим работу алгоритма подробнее. В программный модуль загружается одно изображение здания в цветовой модели RGB. Далее копия исходного изображения приводится к полутоновому (процесс № 1) для применения функции адаптивного порога. Предварительно к изображению применяется функция размытия по методу Гаусса, чтобы уменьшить количество шумов, влияющих на поиск швов здания. После работы функции адаптивного порога применяется морфологическая операция «Замыкание» для частичного замыкания контуров. На этом этапе предварительной обработки изображения завершается.

В процессе № 5 осуществляется поиск линий на бинарном изображении: вызывается функция HoughLinesP (HoughLines) [16] из библиотеки OpenCV, применяемая для обнаружения прямых линий на растровом изображении. Известно, что соединительные швы здания вертикальные или горизонтальные, поэтому другие линии не учитываются. Пример обнаружения линий на швах приведен на рисунке 2.

В результате выполнения процесса № 5 (рис. 1) находятся предварительные соединительные швы здания на изображении. На данном этапе возникают несколько проблем:

- подсчет площади швов (найденные линии накладываются друг на друга несколько раз или, наоборот, линии не покрывают весь шов);
- обнаружение ложных прямых, не лежащих на шве;
- поиск разрушений на швах (массив найденных прямых конкретно не определяет шов).

Следовательно, необходимо провести отбор линий, обнаруженных алгоритмом Хафа, и преобразовать отобранные группы линий в швы.

В процессе № 6 осуществляется группировка линий по следующим критериям:

- тип линии (горизонтальная или вертикальная);
- положение линии на изображении (для вертикальной – координата  $X$ , для горизонтальной –  $Y$ ).

На рисунке 3 приведена блок-схема процесса № 6 со следующими обозначениями: С1 – условие «Является ли линия вертикальной?», С2 – условие «Является ли линия горизонтальной?», С3 – условие «Входит ли текущая вертикальная линия в окрестность существующей группы вертикальных линий?», С4 – условие «Входит ли текущая горизонтальная линия в

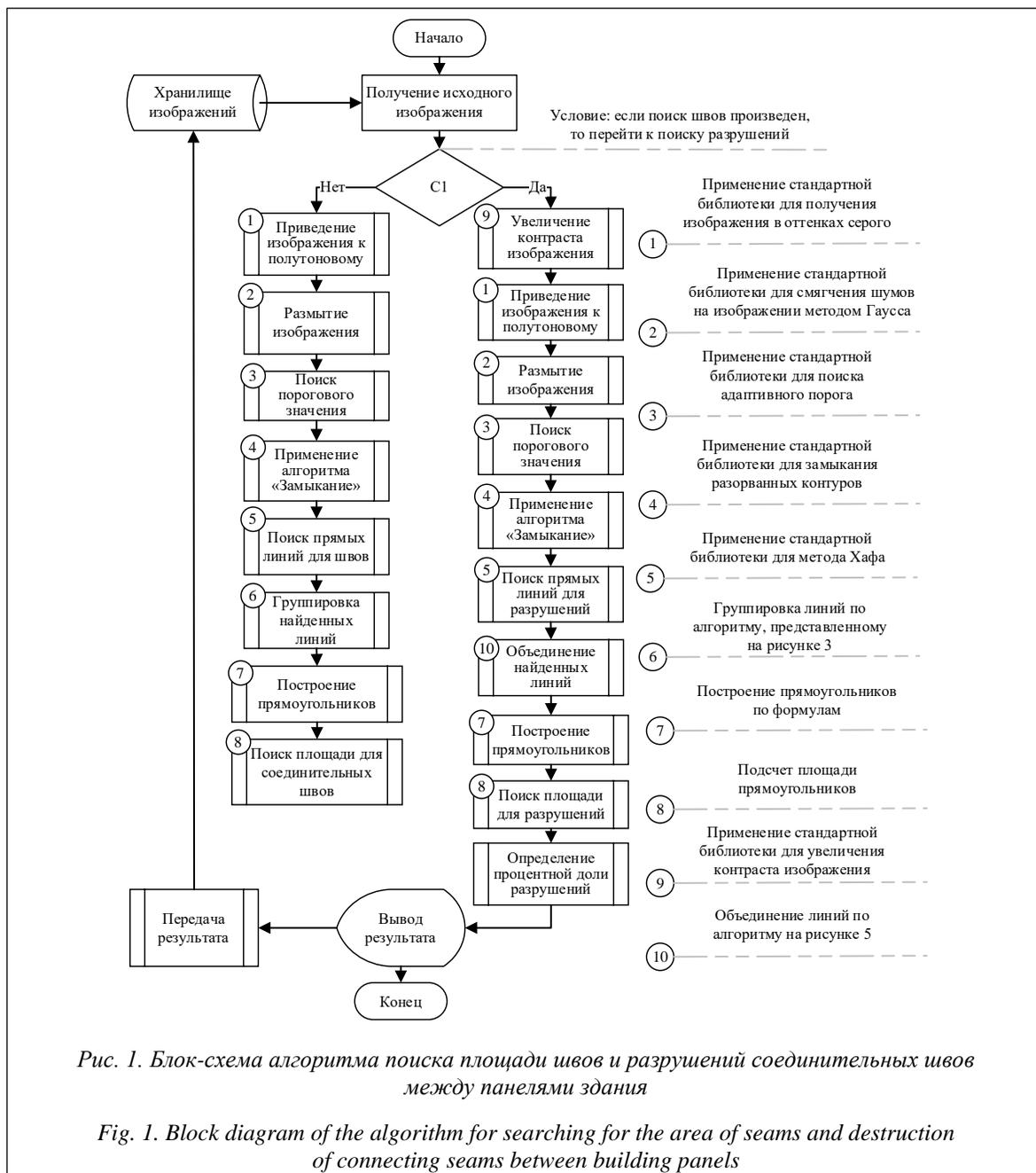


Рис. 1. Блок-схема алгоритма поиска площади швов и разрушений соединительных швов между панелями здания

Fig. 1. Block diagram of the algorithm for searching for the area of seams and destruction of connecting seams between building panels

окрестность существующей группы горизонтальных линий?». Линии числятся в одной группе, если находятся в одной окрестности.

Далее происходит перебор сформированных групп линий (процесс № 7 на рисунке 1). Если численность группы меньше минимально требуемого количества линий, входящих в шов, то такая группа исключается.

Если группа линий удовлетворяет условию, то для нее строится ограничивающий прямоугольник в визуальной системе координат. Определяются верхний левый угол и правый нижний угол прямоугольника по формулам:

$$x_{left} = \min(X), y_{top} = \min(Y), x_{right} = \max(X), y_{bottom} = \max(Y),$$

где  $X = \{x_1, \dots, x_n\}$  – кортеж значений координат начальных и конечных точек линий по оси абсцисс;  $Y = \{y_1, \dots, y_n\}$  – кортеж значений координат начальных и конечных точек линий по оси ординат.

Сформированные прямоугольники хранятся в векторе. Далее прямоугольники перебираются и в случае их пересечения объединяются. На рисунке 4 представлен пример изображения панелей здания с найденными швами после применения метода Хафа и нескольких алгоритмов обработки линий, найденных этим методом.



Рис. 2. Пример обнаружения линий на копии исходного изображения алгоритмом Хафа

Fig. 2. An example of detecting lines on a copy of the original image by the Hough algorithm

После распознавания швов найденные прямоугольники разделяются на фрагменты в соответствии с исключенными областями (окна, двери, лестницы). Особенности алгоритма не

предусматривают изначальное деление прямоугольников на фрагменты, поскольку при построении ограничивающего прямоугольника учитываются минимальные и максимальные значения координат группы линий, находящихся на одной координате и ее окрестности.

Далее отобранные и разделенные прямоугольники строятся на копии исходного изображения и подсчитывается площадь всех найденных швов с учетом вычитания площади пересечения вертикальных и горизонтальных швов.

После нахождения соединительных швов здания осуществляется поиск разрушений на них по алгоритму (другая ветка, рис. 1).

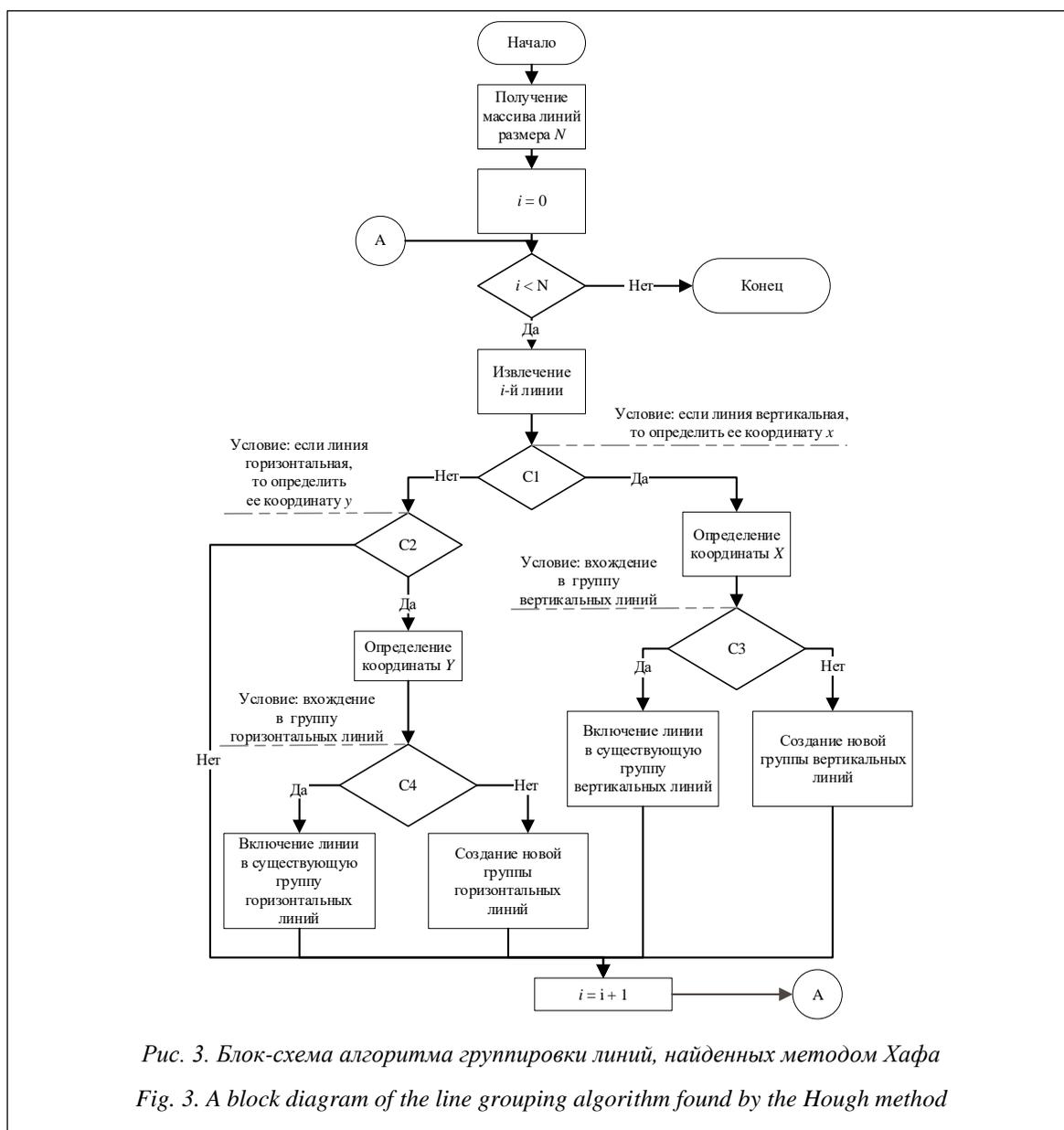


Рис. 3. Блок-схема алгоритма группировки линий, найденных методом Хафа

Fig. 3. A block diagram of the line grouping algorithm found by the Hough method

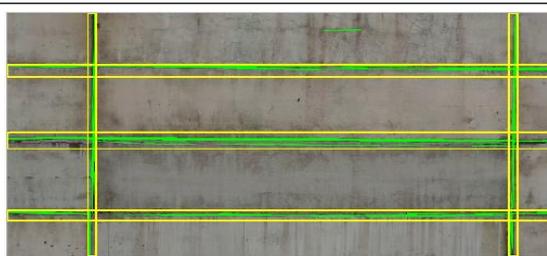


Рис. 4. Пример группировки швов и их распознавания

Fig. 4. An example of grouping seams and their recognition

К копии исходного изображения применяются описанные выше этапы предварительной обработки. Если при процедуре поиска всех

швов на изображении контраст копии исходного изображения по умолчанию не изменяется, то при поиске разрушений контраст копии исходного изображения обязательно увеличивается для выделения темных участков с текущими и прогнозируемыми разрушениями.

В процессе № 5 также вызывается функция HoughLinesP для поиска линий на бинарном изображении. Далее происходит перебор найденных линий. Линии, принадлежащие исключенным областям, не рассматриваются. Если линия вертикальная или горизонтальная, то проверяется ее принадлежность к определенному ранее найденному шву (прямоугольнику), координаты линии записываются в конкретную группу в словаре GroupLinesDamage.

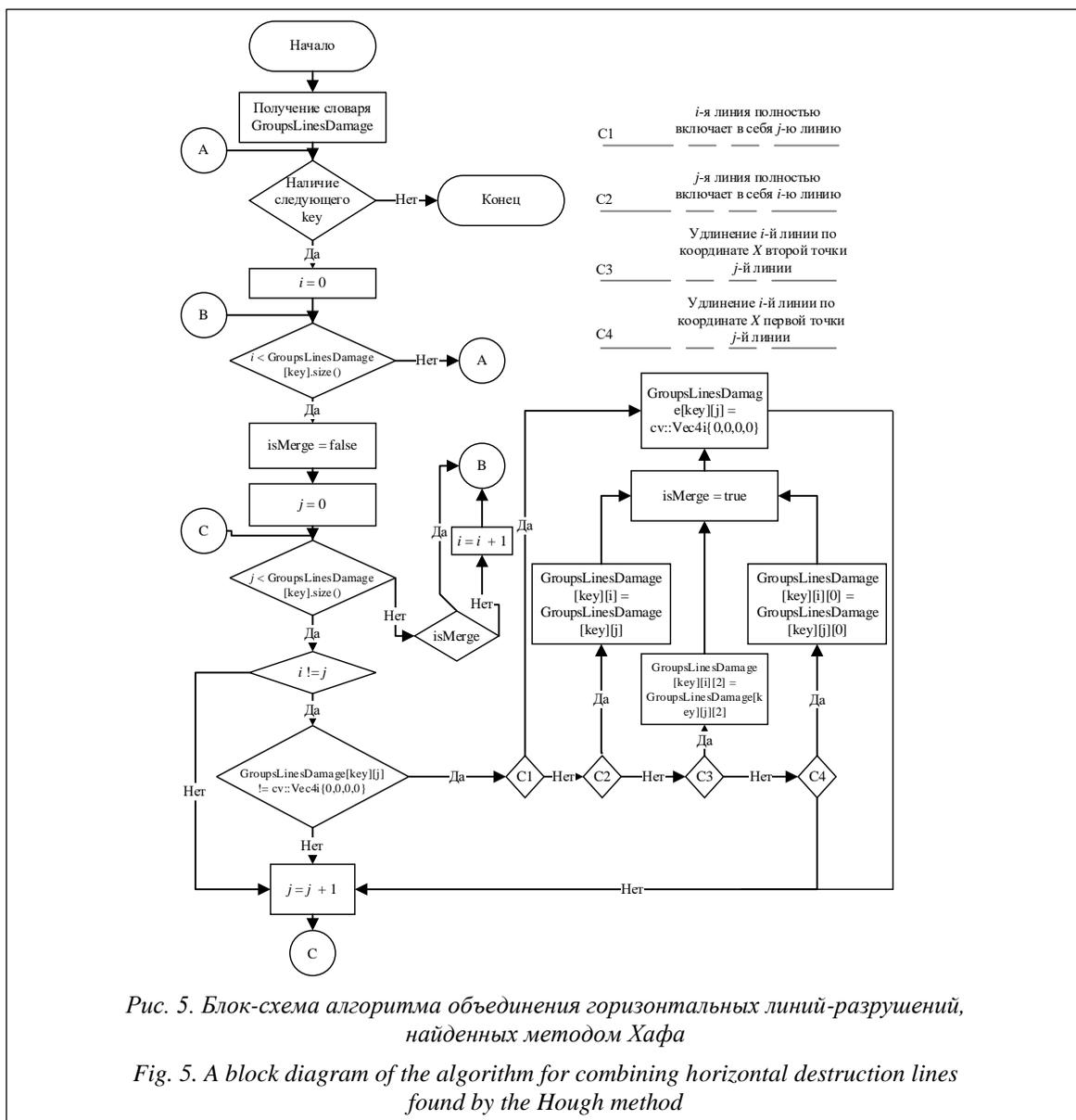


Рис. 5. Блок-схема алгоритма объединения горизонтальных линий-разрушений, найденных методом Хафа

Fig. 5. A block diagram of the algorithm for combining horizontal destruction lines found by the Hough method

После перебора всех линий невозможно подсчитать площадь разрушений, так как линии могут накладываться друг на друга. В процессе № 10 (рис. 1) найденные линии объединяются с учетом их пересечения и наложения (алгоритм для горизонтальных линий изображен на рисунке 5). Алгоритм для вертикальных линий аналогичный, только удлинение линии происходит по оси OY.

Затем повторно перебираются группы швов-разрушений, и для каждой линии из группы швов строится индивидуальный ограничивающий прямоугольник (рис. 6), подсчитывается площадь разрушений.



Рис. 6. Пример обнаруженных разрушений швов

Fig. 6. An example of detected seam failures

## Результаты

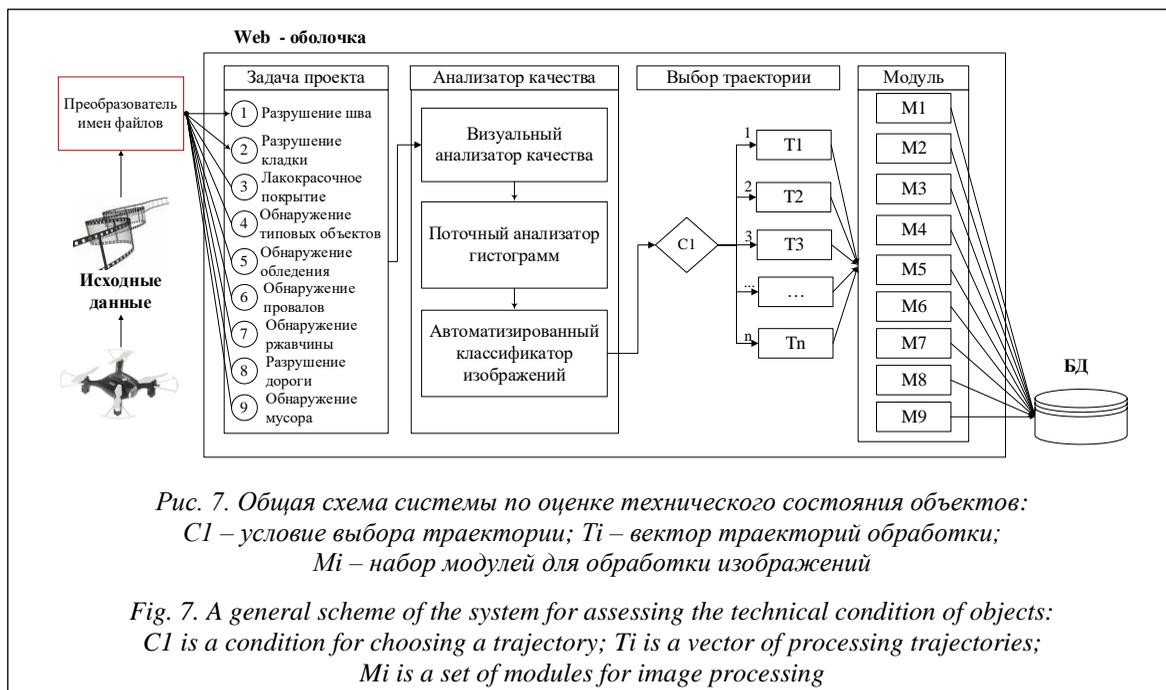
**Общая схема системы по оценке технического состояния объектов.** Программный модуль, описанный в данной работе, предназначен для решения одной из задач проекта построения системы по оценке технического состояния объектов на металлургическом предприятии (рис. 7). Прежде всего осуществляется видеосъемка объектов исследования на территории предприятия с помощью БПЛА. Далее извлекаются кадры из видеопотоков, каждый из кадров получает название в зависимости от задачи по определенному принципу. На следующем этапе полученные кадры проходят визуальный анализ качества и для каждого изображения строится гистограмма яркости. Построенные гистограммы сортируются, определяется пригодность изображения к обработке разработанными программными модулями. На основе вывода выбирается траектория обработки изображения в зависимости от находящегося на изображении объекта и качества кадра.

**Программная реализация модуля для внедрения в систему экспертной оценки при**

**принятии решений о техническом состоянии панельных зданий.** Программный модуль реализован на языке C++ и является динамически подключаемой библиотекой \*.dll для Windows, \*.so для Linux. Модуль разрабатывался с использованием кроссплатформенной среды разработки Qt Creator и фреймворка Qt. Модуль использует библиотеку OpenCV для обработки изображений, которая подключается динамически во время выполнения приложения, а также модуль QtCore для сборки. В программном коде (в заголовочном файле) определен класс BuildingDamageFinder.

Так как модуль предназначен для работы на Linux-сервере, разработано консольное приложение с парсером командной строки, чтобы при запуске исполняемого файла через терминал или функции языка программирования, которые работают с командной строкой, передавать опции и аргументы. Для считывания опций и аргументов из командной строки используются классы QApplication, QCommandLineParser, для установки опций и аргументов – QCommandLineOption. Для автоматического запуска исполняемого файла, к которому подключен программный модуль, сгенерирован bash-скрипт, настраивающий необходимые для работы exe-файла переменные окружения (пути к библиотекам и т.д.). Для развертывания C++ Qt приложения использовалась кроссплатформенная утилита CQtDeployer. Кроме двух изображений здания с выделенными швами и разрушениями, результатом работы модуля также является JSON-файл, поэтому в программном коде используются классы QJsonValue, QJsonObject, QJsonDocument.

**Исследование результатов работы алгоритмов по обнаружению разрушения межпанельных швов здания.** Наличие групп параметров позволяет градуировать уровень обнаруживаемых разрушений соединительных швов здания. В таблице 1 представлен пример градации уровня разрушений соединительных швов здания на изображении для некоторых значений. Выбор определенного уровня разрушений зависит от задач исследователя. По умолчанию разрушениями считаются затемнения на швах и отсутствие материала в шве. Затемнения на швах – их возможные будущие серьезные разрушения. Учет позволяет спрогнозировать долю разрушений соединительных швов здания через некоторое время. При увеличении значения параметра (изменение контраста) потенциальные будущие разрушения



постепенно перестают учитываться в процентной доле, и в расчетах участвуют фрагменты швов без материала, поэтому процент разрушений уменьшается. Изменение параметра постепенно увеличивает контраст копии исходного изображения для обработки.

На рисунке 8 представлены результаты градации уровня обнаруживаемых разрушений с помощью последовательного изменения параметра. Наблюдаются резкие изменения процентной доли разрушений при установке значения параметра с 4-го по 9-е. При значении параметра с 10-го по 15-е замечена стабилизация итогового результата. Таким образом, при начальном и близких к нему значениях параметра происходило отсекание затемнений на швах в быстром темпе. Поскольку упор в созданном алгоритме сделан именно на обнаружение серьезных разрушений швов, процентная доля разрушений также резко изменялась. Дальнейшее изменение значения параметра на итоговый результат не влияет, так как слабые разрушения уже не учитываются в расчетах, а оставшиеся дефекты не могут скрыться из-за серьезной степени разрушения шва.

Окна, двери, лестницы на фасаде и окружающая среда, попадающая в кадры, не представляют интереса для исследователя и мешают процессу обнаружения разрушений швов здания. Возможны два способа выбора областей исследования:

– пользователь выбирает только полезные области и самостоятельно не включает в иссле-

дование лишние объекты, получает общий и отдельный результаты для каждой области;

– пользователь сначала исключает некоторые области, а затем обрабатывает одну полезную область (с исключенными областями).

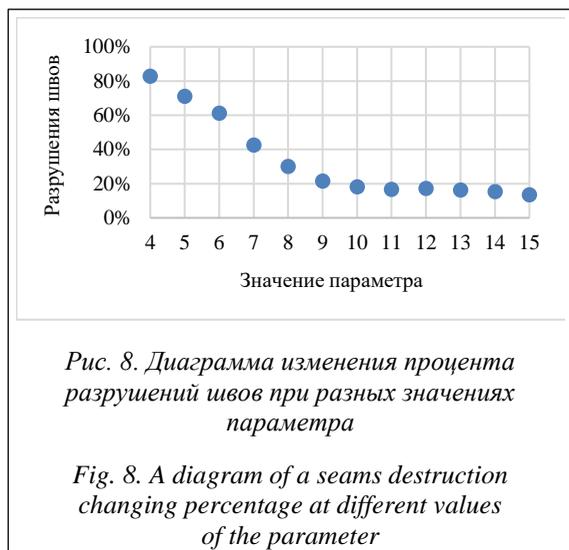
Таблица 1

**Градация уровня обнаруживаемых разрушений соединительных швов здания**

Table 1

**A gradation of the level of detected destruction of the building connecting seams**

Значение параметра	Выходное изображение с выделенными разрушениями	Доля разрушений швов, %
4		82,67
6		61,19
7		42,42
9		21,37
13		16,21
15		13,34



Проведено сравнение этих способов (табл. 2). Перед сравнением изображений с фрагментом или фасадом здания отобраны образцы для исследования из существующей коллекции изображений.

Очевидно, что параметры обнаружения швов и разрушений одинаковы для обоих способов. По умолчанию разрушением считаются полное отсутствие материала в шве, а также затемнения на швах, поэтому доля разрушений на некоторых изображениях превышает 50 %.

Следует отметить, что при обработке образца № 5 двумя способами осуществлена корректировка параметров для обнаружения всех швов фрагмента здания и разрушений швов. Это связано с низкой средней интенсивностью пикселей изображения, если, например, сравнивать со средней интенсивностью пикселей образца № 3 и других образцов.

Иногда соединительные швы обнаруживаются неверно. В большинстве случаев эксперт может исправить ошибки обнаружения с помощью корректировки имеющихся параметров.

**Время выполнения обработки изображения.** Описанный в работе программный модуль создан для сокращения времени первичного осмотра здания на наличие разрушений. Время обработки зависит от архитектуры компьютера, операционной системы и характеристик, а также от операций, выполняемых в фоновом режиме во время процесса обработки изображения.

Визуальный осмотр исходных и выходных изображений показал, что разрушения выделяются верно и процентная доля разрушений сопоставима с количеством выделенных соединительных швов и разрушений на них.

Таблица 2  
Сравнение двух способов работы с исходным изображением

Table 2  
A comparison of two ways of working with an original image

Номер изображения	Исходное изображение фрагмента фасада здания	Результат оценки доли разрушений швов, %	
		Способ 1	Способ 2
1		16,48	18,15
2		93,63	91,47
3		65,32	64,11
4		31,73	31,89
5		5,8	6,18

Изучение алгоритмов и исследование результатов работы программного модуля позволяют выделить следующие отличительные особенности разработки:

- возможность обработки фасадов зданий, снятых БПЛА;
- исключение фрагментов изображения путем передачи координат в программный модуль для построения черного прямоугольника на изображении;
- обработка изображений с учетом исключенных фрагментов изображений;
- обязательное соблюдение параллельности камеры и объекта исследования для верного обнаружения швов;

- подсчет площади швов, текущих и потенциальных разрушений межпанельных швов здания;
- поиск разрушений на швах зданий с разной облицовкой панелей за счет наличия группы параметров;
- работа с изображениями, имеющими особенности: косметический ремонт, низкая интенсивность пикселей, ремонт на панелях;
- получение изображений с выделенными швами и разрушениями;
- получение процентной доли разрушений швов.

### Заключение

Результаты проведенного исследования позволили сделать следующие выводы.

Представленная разработка может заменить работу экспертов по визуальному обследованию зданий, уменьшить срок получения результата, а также сэкономить средства за счет

отказа от услуг экспертной группы по первичному осмотру зданий.

Предложенным методом можно обрабатывать изображения разных размеров. Это условие важно для предприятий с большим количеством зданий в материальном активе, так как метод дает возможность обрабатывать фасады и склейки здания.

Комбинация нескольких параметров сильно влияет на результат обнаружения швов. Установка полей для изменения параметров на пользовательском интерфейсе позволит эксперту улучшить результат обнаружения разрушений и уменьшить зависимость результата от размера изображения, яркости пикселей и общего состояния блоков здания.

Внедрение такого рода программных модулей на крупных предприятиях позволит своевременно обнаруживать и оценивать разрушения соединительных швов панельных зданий.

### Список литературы

1. Леденев В.В., Ярцев В.П. Обследование и мониторинг строительных конструкций зданий и сооружений. Тамбов: изд-во ТГТУ, 2017. 252 с.
2. Qu Z., Wang L. Gating attention convolutional networks with dense connection for pixel-level crack detection. *Multimedia Systems*, 2022, pp. 1–12. doi: 10.1007/s00530-022-01008-3.
3. Zhang Y., Li X., Qiu J., Zhai X., Wei M. GFU-Net: A deep learning approach for automatic metal crack detection. *Proc. NCAAA*, 2021, pp. 375–388. doi: 10.1007/978-981-16-5188-5\_27.
4. Munawar H.S., Hammad A.W.A., Waller S.T., Islam Md.R. Modern crack detection for bridge infrastructure maintenance using machine learning. *Human-Centric Intelligent Systems*, 2022, vol. 2, no. 3-4, pp. 95–112. doi: 10.1007/s44230-022-00009-9.
5. Tian L., Wang Z., Liu W. et al. An improved generative adversarial network with modified loss function for crack detection in electromagnetic nondestructive testing. *Complex Intell. Syst.*, 2022, vol. 8, no. 1, pp. 467–476. doi: 10.1007/s40747-021-00477-9.
6. Jiang F., Wang G., He P., Zheng C., Xiao Z., Wu Y. Application of canny operator thresh-old adaptive segmentation algorithm combined with digital image processing in tunnel face crevice extraction. *The J. of Supercomput.*, 2022, vol. 78, no. 9, pp. 11601–11620. doi: 10.1007/s11227-022-04330-9.
7. Pansare D., Panchwadkar A. Crack detection using image processing. *AIP Conf. Proc.*, 2022, vol. 2653, art. 030007. doi: 10.1063/5.0116894.
8. Attachie J., Owusu G. Utilization of image processing strategy to detect crack on walls. *Proc. ICENCO*, 2020, pp. 43–46. doi: 10.1109/ICENCO49778.2020.9357395.
9. Li J., Li X., Liu K., Yao Z. Crack identification for bridge structures using an unmanned aerial vehicle (UAV) incorporating image geometric correction. *Buildings*, 2022, vol. 12, no. 11, art. 1869. doi: 10.3390/buildings12111869.
10. Cheng C., Seo H., Jun C., Zhao Y. A potential crack region method to detect crack using image processing of multiple thresholding. *Signal, Image and Video Processing*, 2022, vol. 16, no. 6, pp. 1673–1681. doi: 10.1007/s11760-021-02123-w.
11. Liu Z., Li X., Li J., Teng S. New approach to automatically calibrate and detect building cracks. *Buildings*, 2022, vol. 12, art. 1081. doi: 10.3390/buildings12081081.
12. Наркевич М.Ю., Логунова О.С., Аркулис М.Б. и др. Прикладная цифровая платформа для оценки динамики качества опасных производственных объектов на металлургическом предприятии: структура и алгоритмы // *Вестн. ЧГУ*. 2022. № 5. С. 29–48. doi: 10.23859/1994-0637-2022-5-110-3.
13. Гарбар Е.А., Логунова О.С. Экспресс детектор дефектов листового проката: Свид. о регистр. ПрЭВМ № 2022611184. Рос. Федерация, 2022.
14. Qi Z., Liu D., Zhang J., Chen J. Micro-concrete crack detection of underwater structures based on convolutional neural network. *Machine Vision and Applications*, 2022, vol. 33, art. 74. doi: 10.1007/s00138-022-01327-5.
15. Наркевич М.Ю., Логунова О.С., Корниенко В.Д. и др. Мониторинг состояния зданий и сооружений с помощью беспилотных летательных аппаратов: результаты пилотного эксперимента // Программное обеспечение для цифровизации предприятий и организаций: сб. тр. конф. 2021. С. 33–37.
16. Biao C., Bangfeng D., Jiangtao W. Application of an improved hough transform and image correction algorithm in ACC. *JPCS*, 2020, vol. 1621, art. 012044. doi: 10.1088/1742-6596/1621/1/012044.

**Software for automated detection and assessment of building seam failures**

**Veronika V. Kabanova**  
**Oksana S. Logunova**  
**Mikhail Yu. Narkevich**  
**Vladimir D. Kornienko**

**For citation**

Kabanova, V.V., Logunova, O.S., Narkevich, M.Yu., Kornienko, V.D. (2023) 'Software for automated detection and assessment of building seam failures', *Software & Systems*, 36(3), pp. 474–485 (in Russ.). doi: 10.15827/0236-235X.142.474-485

**Article info**

Received: 20.01.23

After revision: 14.04.23

Accepted: 17.05.2023

**Abstract.** The aim of this study is to develop a software module for an automated detection and assessment of building seam failures in an industrial plant. A research involved implementing a software module and a graphical user interface for working with it. The paper proposes a method for detecting seams and assessing building seam failures based on image contrast variation, adaptive threshold, Hough algorithm and several data processing algorithms to complement the Hough algorithm. In order to investigate the condition of building facades of an industrial enterprise, a drone photography and video survey was carried out on the territory of PJSC Magnitogorsk Iron & Steel Works. The resulting collection of images was used to debug the work of the created software module and the method proposed in the article for detecting seams and damage at the building seams. The proposed method is implemented in a software module and allows processing images of different size and pixel brightness. The authors envisage the application of this method with excluded areas in the image. The paper considers current methods of visual inspection of buildings and gives examples of existing software for monitoring the condition of objects. It also describes the procedure of survey performance and presents schemes of joint and fracture detection procedures implemented in the software module. There are the features of the method proposed by the authors. The paper gives an example of gradation of the current and predicted damages of the building seams. The survey included selecting a group of parameters to use in the method; changing in the parameter values enables an expert to adjust joint detection. This development allows estimating the condition of buildings and constructions located at PJSC Magnitogorsk Iron & Steel Works. There are examples of images with separated seams and damages, as well as results of percentage estimations of building connection seams on the image. The results of the study confirm the software module efficiency and the need to implement software modules to monitor the condition of the enterprise territories and buildings.

**Keywords:** damage detection, pattern recognition, software module, building condition monitoring, damage assessment, Hough algorithm

**Acknowledgements.** The work is carried out under contract No 247715 dated July 5, 2021 between PJSC Magnitogorsk Iron & Steel Works and the Nosov Magnitogorsk State Technical University called "Development and application of methods for monitoring territories, buildings and structures of PJSC Magnitogorsk Iron & Steel Works using unmanned aerial vehicles (UAV)"

**Reference List**

1. Ledenev, V.V., Yartsev, V.P. (2017) Survey and Monitoring of Building Structures of Buildings and Structures. Tambov, 252 p. (in Russ.).
2. Qu, Z., Wang, L. (2022) 'Gating attention convolutional networks with dense connection for pixel-level crack detection', *Multimedia Systems*, pp. 1–12. doi: 10.1007/s00530-022-01008-3.
3. Zhang, Y., Li, X., Qiu, J., Zhai, X., Wei, M. (2021) 'GFU-Net: A deep learning approach for automatic metal crack detection', *Proc. NCA*, pp. 375–388. doi: 10.1007/978-981-16-5188-5\_27.
4. Munawar, H.S., Hammad, A.W.A., Waller, S.T., Islam, Md.R. (2022) 'Modern crack detection for bridge infrastructure maintenance using machine learning', *Human-Centric Intelligent Systems*, 2(3-4), pp. 95–112. doi: 10.1007/s44230-022-00009-9.
5. Tian, L., Wang, Z., Liu, W. et al. (2022) 'An improved generative adversarial network with modified loss function for crack detection in electromagnetic nondestructive testing', *Complex Intell. Syst.*, 8(1), pp. 467–476. doi: 10.1007/s40747-021-00477-9.
6. Jiang, F., Wang, G., He, P., Zheng, C., Xiao, Z., Wu, Y. (2022) 'Application of canny operator thresh-old adaptive segmentation algorithm combined with digital image processing in tunnel face crevice extraction', *The J. of Supercomput.*, 78(9), pp. 11601–11620. doi: 10.1007/s11227-022-04330-9.
7. Pansare, D., Panchwadkar, A. (2022) 'Crack detection using image processing', *AIP Conf. Proc.*, 2653, art. 030007. doi: 10.1063/5.0116894.
8. Attachie, J., Owusu, G. (2020) 'Utilization of image processing strategy to detect crack on walls', *Proc. ICENCO*, pp. 43–46. doi: 10.1109/ICENCO49778.2020.93573956.
9. Li, J., Li, X., Liu, K., Yao, Z. (2022) 'Crack identification for bridge structures using an unmanned aerial vehicle (UAV) incorporating image geometric correction', *Buildings*, 12(11), art. 1869. doi: 10.3390/buildings12111869.
10. Cheng, C., Seo, H., Jun, C., Zhao, Y. (2022) 'A potential crack region method to detect crack using image processing of multiple thresholding', *Signal, Image and Video Processing*, 16(6), pp. 1673–1681. doi: 10.1007/s11760-021-02123-w.

11. Liu, Z., Li, X., Li, J., Teng, S. (2022) 'New approach to automatically calibrate and detect building cracks', *Buildings*, 12, art. 1081. doi: 10.3390/buildings12081081.
12. Narkevich, M.Yu., Logunova, O.S., Arkulis, M.B. et al. (2022) 'Applied digital platform for assessing the dynamics of the quality of hazardous industrial facilities at a metallurgical enterprise: structure and algorithms', *ChSU Bull.*, (5), pp. 29–48 (in Russ.). doi: 10.23859/1994-0637-2022-5-110-3.
13. Garbar, E.A., Logunova, O.S. (2022) *Express Detector of Sheet Metal Defects*, Pat. RF, № 2022611184.
14. Qi, Z., Liu, D., Zhang, J., Chen, J. (2022) 'Micro-concrete crack detection of underwater structures based on convolutional neural network', *Machine Vision and Applications*, 33, art. 74. doi: 10.1007/s00138-022-01327-5.
15. Narkevich, M.Yu., Logunova, O.S., Kornienko, V.D. et al. (2021) 'Monitoring the condition of buildings and structures using unmanned aerial vehicles: results of a pilot experiment', *Proc. Software for Digitalization of Enterprises and Organizations*, pp. 33–37 (in Russ.).
16. Biao, C., Bangfeng, D., Jiangtao, W. (2020) 'Application of an improved hough transform and image correction algorithm in ACC', *JPCS*, 1621, art. 012044. doi: 10.1088/1742-6596/1621/1/012044.

**Авторы**

**Кабанова Вероника Валерьевна**<sup>1</sup>,  
студент, kabanova.v.v.nmstu@gmail.com  
**Логунова Оксана Сергеевна**<sup>1</sup>, д.т.н., профессор,  
logunova66@mail.ru  
**Наркевич Михаил Юрьевич**<sup>1</sup>, к.т.н., доцент,  
narkevich\_mu@mail.ru  
**Корниенко Владимир Дмитриевич**<sup>1</sup>, эксперт,  
ведущий инженер,  
volodya.kornienko2319@yandex.ru

<sup>1</sup> Магнитогорский государственный  
технический университет им. Г.И. Носова,  
г. Магнитогорск, 455000, Россия

**Authors**

**Veronika V. Kabanova**<sup>1</sup>, Student,  
kabanova.v.v.nmstu@gmail.com  
**Oksana S. Logunova**<sup>1</sup>, Dr.Sc. (Engineering),  
Professor, logunova66@mail.ru  
**Mikhail Yu. Narkevich**<sup>1</sup>, Ph.D. (Engineering),  
Associate Professor, narkevich\_mu@mail.ru  
**Vladimir D. Kornienko**<sup>1</sup>, Expert,  
Leading Engineer,  
volodya.kornienko2319@yandex.ru

<sup>1</sup> Nosov Magnitogorsk State Technical University  
Magnitogorsk, 455000,  
Russian Federation

## Система назначения персонифицированного лечения по аналогии на основе гибридного способа извлечения прецедентов

В.В. Грибова  
Р.И. Ковалев  
Д.Б. Окунь

### Ссылка для цитирования

Грибова В.В., Ковалев Р.И., Окунь Д.Б. Система назначения персонифицированного лечения по аналогии на основе гибридного способа извлечения прецедентов // Программные продукты и системы. 2023. Т. 36. № 3. С. 486–492. doi: 10.15827/0236-235X.142.486-492

### Информация о статье

Поступила в редакцию: 06.02.23

После доработки: 22.02.23

Принята к публикации: 12.04.2023

**Аннотация.** В работе описывается система назначения персонифицированного лечения на основе прецедентов. Ее уникальной особенностью является извлечение прецедентов на основе гибридного метода, сочетающего извлечение прецедента на основе знаний с классическим способом К-ближайших соседей. Новизна предлагаемого подхода заключается в обеспечении максимальной гибкости и корректности в оценке схожести прецедентов. В работе описаны информационные и программные компоненты системы. Используемая база знаний, как и все информационные ресурсы, строится по своим онтологиям, четко задающим их структуру и семантику. Это позволяет оперативно вносить изменения без привлечения программистов и переработки всей системы. Система реализована на основе мультиагентного подхода. На первом этапе с помощью базы знаний производится предварительный расчет и приведение всех признаков к единой метрике, на втором – непосредственный расчет схожести методом К-ближайших соседей. Схожесть историй болезни определяется совокупно по каждому признаку. На практике система позволяет максимально гибко и точно оценивать схожесть историй болезни, содержащих разнородные по типу признаки. Предлагаемое решение особенно эффективно в условиях дефицита медицинских знаний и данных, когда системы иного типа, в частности, основанные на знаниях, не могут предложить корректное решение.

**Ключевые слова:** система поддержки принятия решений, прецедент, интеллектуальная система, рассуждения по аналогии

**Благодарности.** Работа выполнена при частичной поддержке в рамках госзаданий FZNS-2023-0010 (разработка методов и моделей), а также 0202-2021-0004 (разработка и реализация системы)

Назначение адекватного лечения по поставленному диагнозу, варианту течения, жалобам пациента, результатам его лабораторных и инструментальных исследований, индивидуальным особенностям является для врача сложной задачей. Именно ошибки в назначении лечения особенно часто встречаются и имеют наибольшие негативные последствия.

Системы поддержки принятия врачебных решений разрабатываются для помощи врачам в постановке диагноза и назначении персонифицированного лечения. В литературе приводятся различные классификации методов и подходов к созданию таких систем [1]. Для улучшения клинической практики в некоторых случаях используются гибридные подходы к их реализации [2].

В условиях дефицита медицинских знаний о лечении какого-либо заболевания (например, в начале пандемии COVID-19, при появлении его новых штаммов) или атипичного проявления заболевания и реакции на лечение наиболее целесообразны методы создания систем поддержки принятия решений на основе прецедентов [2].

Эти методы позволяют решать новую, неизвестную задачу, используя накопленный опыт

решения подобных задач. Актуальность создания таких методов обусловлена тем, что они позволяют работать с малыми наборами данных, в то время как использование других методов может привести к неточным и даже ошибочным выводам.

Наиболее известными методами извлечения прецедентов [3] являются методы ближайшего соседа, извлечения прецедентов на основе деревьев решений и извлечения прецедентов на основе знаний.

Метод ближайшего соседа наиболее простой и универсальный. С его помощью разрабатываются такие системы поиска похожих случаев при лечении, как CBR-DENT [4], DePicT [5], «Спутник Врача», разработанная в ИСП РАН [6], а также система по планированию лучевой терапии [7]. К недостаткам метода относятся сложность выбора метрики для определения степени сходства, неэффективность при работе с неполными данными, зависимость от вычислительных ресурсов, а также от размера *базы прецедентов* (БП) [3].

Метод извлечения прецедентов на основе деревьев решений обычно используется для больших БП, так как основная часть работы по

извлечению прецедентов выполняется заранее на этапе построения дерева решений, что значительно сокращает время поиска решения. Данный подход в сочетании с методом ближайшего соседа использует система ReCall от ISoft [8]. Существенным недостатком систем на основе деревьев решений является то, что деревья дают полезные результаты только в случае независимых признаков. Имеет место и проблема значимости [9]: дерево дробит данные на большое количество частных случаев. Чем их больше, тем меньше обучающих прецедентов попадает в каждый такой частный случай.

Метод извлечения прецедентов на основе знаний позволяет учесть знания экспертов и может успешно применяться совместно с другими методами извлечения прецедентов, особенно когда БП имеют большие размеры и предметная область является открытой и динамичной.

Одна из главных проблем прецедентного подхода – выбор соответствующей метрики для индексации и сравнения прецедентов. В каждом конкретном случае этот выбор производится поразному в зависимости от целей лица, принимающего решения, а также самих данных. Серьезным вопросом является интерпретация дискретных и качественных значений.

Анализ существующих систем на основе прецедентов позволил сделать вывод, что либо рассмотренные программные решения ориентированы на узкую задачу, либо описание прецедентов и получаемое решение выполнены в математических терминах, непонятных предметным специалистам. Каждый метод в отдельности имеет существенные недостатки и не позволяет в полной мере использовать такие системы в медицине. Авторы данного исследования считают наиболее перспективным применение гибридных способов, сочетающих различные методы поиска прецедентов.

Целью исследования является разработка оболочки системы поддержки принятия решений по назначению лечения на основе прецедентов, применимой для поиска лечения различных заболеваний (независимо от раздела медицины, класса заболевания, его формы) с возможностью ее интеграции в различные интеллектуальные системы.

### Постановка задачи

Данная работа ориентирована на поиск прецедентов при назначении лечения. На вход системе подается история болезни пациента

с поставленным диагнозом. Требуется найти похожие истории из БП, ранжировать их по степени сходства, а также обеспечить пользователю демонстрацию сходных элементов из текущей истории болезни и ранжированного списка.

История болезни представляет собой формальное описание в медицинских терминах всей информации о больном, включающей его индивидуальные характеристики, анамнез жизни, блоки жалоб, объективные, лабораторные и инструментальные исследования, а также поставленный диагноз. Истории болезни из БП также включают блок с описанием назначенного лечения, возможно, дневник, отображающий динамику лечения и его результат.

История болезни формируется на основе специальной онтологии, которая позволяет четко задать структуру и семантические связи внутри истории болезни [10]. Значения признаков (жалоб, лабораторных и инструментальных исследований и др.) выбираются из базы знаний медицинских терминов и признаков. Она также формируется на основе онтологии, содержит описания признаков и наблюдений для формирования истории болезни, в ней определены типы их шкал и возможные значения.

База знаний медицинских терминов и признаков постоянно пополняется новыми описаниями (данная работа проводится различными медицинскими коллективами, которые формируют терминологические справочники для «своих» разделов медицины). Учитывая этот факт, принципиальным требованием к системе является ее устойчивость: появление новых знаний в базе знаний не должно приводить к модификации разработанной системы [11].

Для реализации всех поставленных задач предлагается гибридный подход к извлечению прецедента. Степени сходимости признаков будут определяться на основе базы знаний медицинских терминов и признаков в сочетании с классическим методом К-ближайших соседей.

### Разработанное решение

Для достижения поставленной цели были разработаны информационные (онтологии и порожденные на их основе БД) и программные (средства редактирования знаний и данных, а также решатель задач) компоненты [12, 13].

Онтология базы знаний медицинских терминов, признаков и наблюдений описывает структуру используемых на практике медицинских

терминов и наблюдений, а также структуру вспомогательной терминологии, необходимую для целостности описания медицинских знаний. В структуре онтологии использована синонимия. Структура описания признаков включает не только описание их имен, характеристик и возможных значений (качественных, количественных и интервальных), но и значения референсных диапазонов с учетом пола, возраста и состояний (например, беременность), а также различных факторов, таких как эпидемиологические (контакт с больным, контакт с животным, характер работы), аллергические, факторы риска (нездоровый образ жизни, вредность производства, сезонность).

По данной онтологии формируется база знаний медицинских терминов и признаков (рис. 1). Все признаки или наблюдения подразделяются на простые, такие как «Пол», «Возраст», и составные, имеющие некоторый набор характеристик, например, признак «Боль в суставе» обладает характеристиками «локализация», «интенсивность», «периодичность», «характер». Возможные значения признаков (характеристик) разделяются на числовые и качественные.

Все качественные значения имеют порядковую шкалу или же шкалу наименований [14]. Порядковая шкала предполагает ранжирование (упорядочение) значений признака в зависимо-

сти от масштабирования. Значения в порядковой шкале обычно располагаются по возрастанию или убыванию, например, характеристика «интенсивность» признака «Головная боль» имеет следующие значения в порядковой шкале: очень слабая, слабая, умеренная, сильная, резкая, резчайшая. Шкала наименований содержит набор возможных значений признака, максимально отличающихся друг от друга.

Онтология фармакологического справочника описывает элементы, позволяющие формально представить лекарственное средство в виде международного непатентованного наименования или фиксированной комбинации в его полном фармакологическом описании.

Онтология истории болезни задает структуру описания здоровья человека на всем его жизненном пути. Данная структура позволяет описать сам прецедент, включающий все случаи оказания медицинской помощи, профилактических мероприятий, оперативно предоставить доступ к информации о проведенных исследованиях, случаях амбулаторно-поликлинического, стационарного и санаторно-курортного лечения пациента, вызовах скорой помощи, содержит раздел о противопоказаниях к применению тех или иных видов лечения для конкретного пациента и перечень непереносимых препаратов.

Программные компоненты разрабатываемой системы включают в себя решатель задач и средства редактирования данных.

Средства редактирования знаний и данных представлены рядом редакторов. Редактор знаний и данных, автоматически генерируемый по онтологии (генератор редакторов входит в состав платформы), обеспечивает порождение целевых ресурсов в соответствии с заданными правилами и контролирует выполнение онтологических соглашений. К тому же автоматически проверяется полнота сформированных данных.

Решатель задач включает два компонента и разработан с использованием мультиагентного подхода. Первый компонент реализован на облачной платформе IACPaaS [15], здесь реализованы интерфейсная подсистема и первичная подготовка данных. Вторая часть выполнена как внешний агент, где происходит непосредственно расчет сходимости. Решатель представляет собой онтолого-ориентированный алгоритм, который для поиска похожих клинических случаев производит обход БП и попарно сопоставляет информацию из истории болезни пациента, а также приводит все наблюдения (признаки) к единой метрике.

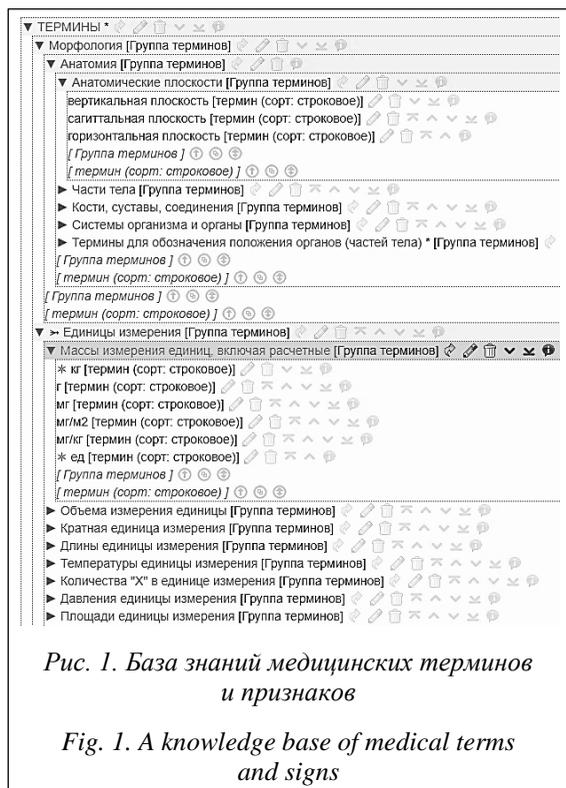


Рис. 1. База знаний медицинских терминов и признаков

Fig. 1. A knowledge base of medical terms and signs

Первый компонент отвечает за выбор истории болезни (рис. 2), после чего она в формате JSON передается во второй компонент, где форматируется и для удобства расчета переносится в СУБД. Для этого все признаки и наблюдения сопоставляются с каждой своей характеристикой (при наличии) и записываются в качестве названия столбцов, в строках записываются их значения. Например, признак «Общая слабость» с характеристиками «Выраженность» и «Периодичность» в СУБД станет двумя признаками – «Общая слабость\_Выраженность» и «Общая слабость\_Периодич-

ность» с соответствующими значениями. Затем происходит обход БП, где попарно сравниваются новый случай из истории болезни и прецедент, для чего каждому признаку (наблюдению) в истории болезни и прецеденте вместо значения присваивается некоторый числовой эквивалент от 0 до 10 в соответствии с базой медицинских терминов и знаний о наблюдениях.

Если качественный признак имеет порядковую шкалу, то ему сопоставляется соответствующее число, где 0 соответствует минимальному значению, а 10 максимальному. Так, напри-

- 38088/16 Острый Холецистит (История болезни)
  - Исследования лабораторные
    - Клинический анализ крови (Группа признаков (вид исследования))
  - Биохимическое исследование крови (Группа признаков (вид исследования))
    - 1 (Номер записи)
      - 20.11.2016-13:13:00.000 (дата)
      - Белок общий в сыворотке (Признак)
      - Креатинин крови (Признак)
        - мкмоль/л (ед.изм.)
        - 75 (значение)
      - Мочевина в крови (Признак)
      - Билирубин общий (Признак)
        - мкмоль/л (ед.изм.)
        - 9.8 (значение)
      - Билирубин прямой (Признак)
      - Билирубин непрямой (Признак)
      - Глюкоза крови (Признак)
      - Аланинаминотрансфераза крови (Признак)
        - ед/л (ед.изм.)
        - 30 (значение)
      - Аспартатаминотрансфераза крови (Признак)
      - Альфа-амилаза крови (Признак)
        - Ед/л (ед. изм.)
        - 47 (значение)
    - Исследование мочи (Группа признаков (вид исследования))
  - Исследования инструментальные
  - Диагноз
  - Объективное состояние больного
  - Результаты компьютерной постановки диагноза
  - Паспортная часть
  - 20.11.2016-09:00:00.000 (дата обращения)
  - Жалобы при обращении
    - Боль в животе (Признак)
      - Присутствие (Характеристика)
        - имеется (значение)
      - Локализация (Характеристика)
    - Тошнота (Признак)
    - Рвота (Признак)
    - Сухость во рту (Признак)
    - Общая слабость (Признак)
  - История настоящего заболевания
  - Результаты компьютерного назначения лечения

Рис. 2. Исходный случай

Fig. 2. An initial case

мер, признак «Головная боль» с характеристикой «Интенсивность», имея значение «умеренная», получит эквивалент 2,5 (рис. 3).

Если используется шкала наименований, то в случае совпадения значений или же попадания в одно и то же интегрирующее значение признакам в истории болезни и прецеденте присваивается 0, то есть полное совпадение; если значения признаков различны, то одному присваивается 0, а другому 10, то есть максимальное различие.

При расчете сходимости числовых значений эквивалент рассчитывается следующим образом: 0 сопоставляется диапазону нормы, а далее с шагом 0,6\* диапазона этой нормы идем до нужного нам отрезка. Так, например, «Гемоглобин» со значением 103 из нового случая получает эквивалент -1, а значение 190 из прецедента – эквивалент 2 (рис. 4). Затем полученные значения необходимо сместить на начало диапазона 0–10, сохранив расстояние между ними, где 0 получает признак из нового случая, а 3 – признак из прецедента. Также следует учитывать, что максимальное расстояние не может быть более 10. Предполагается, что любые значения, расстояние между которыми превышает 10, для нас одинаково максимально непохожи.

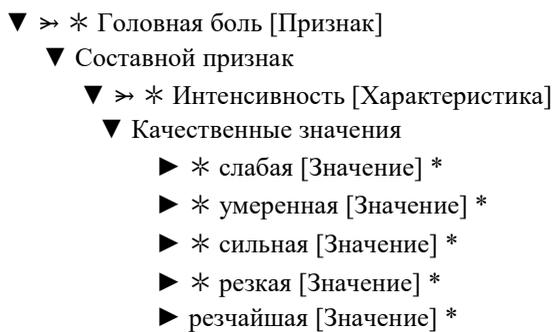


Рис. 3. Порядковая шкала признака «Головная боль»

Fig. 3. An ordinal scale of the "Headache" feature

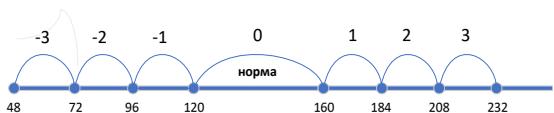


Рис. 4. Шкала значений признака «Гемоглобин»

Fig. 4. A value scale of the "Hemoglobin" feature

Также алгоритм работы решателя подразумевает обработку пропущенных признаков. Если признак отсутствует в прецеденте, то по этому признаку считается максимальное различие, если же в истории болезни – определяется частичная сходимость.

Непосредственный расчет схожести случая из истории болезни и прецедента происходит через евклидово расстояние, а затем переводится в проценты. Такой подход позволяет определить сходимость значений признаков на основе его обобщенного, а не абсолютного значения с учетом различных факторов (пола, возраста, состояний и др.). Первые пять прецедентов, набравшие свыше 60 %, возвращаются в подсистему на платформе IASaaS и предлагаются пользователю в качестве результатов работы (<http://www.swsys.ru/uploaded/image/2023-3/2023-3-dop/1.jpg>). После этого пользователь может провести уточненный поиск прецедента, выбрав интересующие его признаки.

### Заключение

В работе описана реализация системы, помогающей врачу назначать медикаментозное лечение с учетом персональных данных пациента (анамнеза жизни, сопутствующих заболеваний, возраста, предъявляемых жалоб и др.) на основе рассуждения по аналогии (по прецедентам).

Особенностями разработанной системы, принципиально отличающими ее от других медицинских систем, являются ориентация на широкий спектр заболеваний, а также использование гибридного метода извлечения прецедентов: с использованием знаний (базы знаний медицинских терминов и признаков) и метода К-ближайших соседей, позволяющего максимально точно рассчитать схожесть разнотипных признаков. Следует отметить, что истории болезни описываются в соответствии с терминологией, определенной медицинским сообществом, а изменение/дополнение базы медицинских терминов и знаний о наблюдениях не влияет на программные компоненты системы (не требуется модификация программных компонентов). Система реализована как облачный сервис, что позволяет увеличить аудиторию пользователей. Результатом работы системы является не только набор похожих историй болезни, но и демонстрация врачу похожих элементов текущей истории болезни с элементами историй болезни из ранжированного списка.

## Список литературы

1. Гаврилова А.С. Классификация интеллектуальных информационных систем // Информационные системы и технологии в моделировании и управлении: сб. тр. V Междунар. конф. 2020. С. 11–15.
2. Мехоношина Н.В. О системе поддержки и принятия решений в медицине на основе прецедентного подхода // Научная перспектива. 2016. № 11. С. 113–115.
3. Крылов А.В. Проблема извлечения знаний с использованием рассуждений на основе прецедентов // Изв. вузов. Приборостроение. 2018. Т. 61. № 11. С. 956–962. doi: 10.17586/0021-3454-2018-61-11-956-962.
4. Choudhury N., Begum S.A. A survey on case-based reasoning in medicine. IJACSA, 2016, vol. 7, no. 8, pp. 136–144. doi: 10.14569/IJACSA.2016.070820.
5. Nasiri S., Zenkert J., Fathi M. A medical case-based reasoning approach using image classification and text information for recommendation. In: LNTCS. Proc. IWANN, 2015, vol. 9095, pp. 43–55. doi: 10.1007/978-3-319-19222-2\_4.
6. Юдин В.Н., Карпов Л.Е. Неполностью описанные объекты в системах поддержки принятия решений // Программирование. 2017. № 5. С. 24–31.
7. Khussainova G., Petrovic S., Jagannathan R. Retrieval with clustering in a case-based reasoning system for radiotherapy treatment planning. JPCS, 2015, vol. 616, art. 012013. doi: 10.1088/1742-6596/616/1/012013.
8. Hadj-Mabrouk H. Application of case-based reasoning to the safety assessment of critical software used in rail transport. Safety Sci., 2020, vol. 131, art. 104928. doi: 10.1016/j.ssci.2020.104928.
9. Полин Я.А., Зудилова Т.В., Ананченко И.В., Войтюк Т.Е. Деревья решений в задачах классификации: особенности применения и методы повышения качества классификации // Современные наукоемкие технологии. 2020. № 9. С. 59–63. doi: 10.17513/snt.38215.
10. Гаврилова Т.А., Страхович Э.В. Визуально-аналитическое мышление и интеллект-карты в онтологическом инжиниринге // Онтология проектирования. 2020. Т. 10. № 1. С. 87–99. doi: 10.18287/2223-9537-2020-10-1-87-99.
11. Ефименко И.В., Хорошевский В.Ф. Интеллектуальные системы поддержки принятия решений в медицине: ретроспективный обзор состояния исследований и разработок и перспективы // OSTIS. 2017. № 7. С. 251–260.
12. Грибова В.В., Окунь Д.Б. Онтологии для формирования баз знаний и реализации лечебных мероприятий в медицинских интеллектуальных системах // Информатика и системы управления. 2018. № 3. С. 71–80. doi: 10.22250/isu.2018.57.71-80.
13. Gribova V.V., Petryaeva M.V., Okun D.B., Tarasov A.V. Software toolkit for creating intelligent systems in practical and educational medicine. Proc. RPC, 2018, pp. 1–5. doi: 10.1109/RPC.2018.8482130.
14. Цветов В.П. О вложении измерительных шкал // Перспективные информационные технологии: тр. Междунар. науч.-технич. конф. 2018. С. 341–344.
15. Грибова В.В., Клещев А.С., Москаленко Ф.М., Тимченко В.А., Федорищев Л.А., Шалфеева Е.А. Облачная платформа IASaaS для разработки оболочек интеллектуальных сервисов: состояние и перспективы развития // Программные продукты и системы. 2018. Т. 31. № 3. С. 527–536. doi: 10.15827/0236-235X.123.527-536.

Software &amp; Systems

doi: 10.15827/0236-235X.142.486-492

2023, vol. 36, no. 3, pp. 486–492

**The system for prescribing personalized treatment by case-based reasoning using a hybrid precedent extraction method****Valeriya V. Gribova  
Roman I. Kovalev  
Dmitry B. Okun****For citation**

Gribova, V.V., Kovalev, R.I., Okun, D.B. (2023) 'The system for prescribing personalized treatment by case-based reasoning using a hybrid precedent extraction method', *Software & Systems*, 36(3), pp. 486–492 (in Russ.). doi: 10.15827/0236-235X.142.486-492

**Article info**

Received: 06.02.23

After revision: 22.02.23

Accepted: 12.04.2023

**Abstract.** The paper describes a system for prescribing personalized treatment based on precedents. A unique feature of the system is the extraction of precedents based on a hybrid method combining the extraction of a precedent based on knowledge with a classical method of K-nearest neighbors. The novelty of the proposed approach is in ensuring maximum flexibility and correctness in assessing the convergence of precedents. The paper describes the information and software components of the system. The knowledge base used, as well as all information resources, is built according to its ontologies, which clearly define its structure and semantics. This allows making changes to them quickly without involving programmers and processing the entire system. The system is based on a multi-agent approach. The first stage involves a preliminary calculation and bringing all features to a single metric using the knowledge base. The second stage includes

actual calculating the convergence by the K-nearest neighbors method. The convergence of case histories is determined collectively for each attribute. Thus, in practice, the system allows the most flexible and accurate assessment of the similarity of medical histories containing heterogeneous signs. The proposed solution is particularly effective under conditions where there is a shortage of medical knowledge and data, when other types of systems often cannot offer a correct solution, in particular those based on knowledge.

**Keywords:** decision support system, case based reasoning, intelligent system

**Acknowledgements.** This work was partially supported by the state tasks AYATA-2023-0010 (development of methods and models), as well as 0202-2021-0004 (development and implementation of the system)

### Reference List

1. Gavrilova, A.S. (2020) 'Classification of intelligent information systems', *Information Systems and Technologies in Management Modeling*, pp. 11–15 (in Russ.).
2. Mekhonoshina, N.V. (2016) 'On the system of support and decision-making in medicine based on a case-based approach', *Sci. Perspective*, (11), pp. 113–115 (in Russ.).
3. Krylov, A.V. (2018) 'The problem of knowledge retrieving with the use of precedent-based reasoning', *J. of Instrument Engineering*, 61(11), pp. 956–962 (in Russ.) doi: 10.17586/0021-3454-2018-61-11-956-962.
4. Choudhury, N., Begum, S.A. (2016) 'A survey on case-based reasoning in medicine', *IJACSA*, 7(8), pp. 136–144. doi: 10.14569/IJACSA.2016.070820.
5. Nasiri, S., Zenkert, J., Fathi, M. (2015) 'A medical case-based reasoning approach using image classification and text information for recommendation', in: *LNTCS. Proc. IWANN*, 9095, pp. 43–55. doi: 10.1007/978-3-319-19222-2\_4.
6. Yudin, V.N., Karpov, L.E. (2017) 'Dealing with not fully described objects in decision support systems: Alternative approaches', *Programming and Comput. Software*, (5), pp. 24–31 (in Russ.).
7. Khussainova, G., Petrovic, S., Jagannathan, R. (2015) 'Retrieval with clustering in a case-based reasoning system for radiotherapy treatment planning', *JPCS*, 616, art. 012013. doi: 10.1088/1742-6596/616/1/012013.
8. Hadj-Mabrouk, H. (2020) 'Application of case-based reasoning to the safety assessment of critical software used in rail transport', *Safety Sci.*, 131, art. 104928. doi: 10.1016/j.ssci.2020.104928.
9. Polin, Ya.A., Zudilova, T.V., Ananchenko, I.V., Voytiuk, T.E. (2020) 'Decision trees in classification problems: application features and methods for improving the quality of classification', *Modern High Technologies*, (9), pp. 59–63 (in Russ.). doi: 10.17513/snt.38215.
10. Gavrilova, T.A., Strakhovich, E.V. (2020) 'Visual analytical thinking and mind maps for ontology engineering', *Ontology of Designing*, 10(1), pp. 87–99 (in Russ.). doi: 10.18287/2223-9537-2020-10-1-87-99.
11. Efimenko, I.V., Khoroshevsky, V.F. (2017) 'Intelligent decision support systems in medicine: state of the art and beyond', *OSTIS*, (7), pp. 251–260 (in Russ.).
12. Gribova, V.V., Okun, D.B. (2018) 'Ontologies for the formation of knowledge bases about disease treatment in medical intelligent systems', *Informatika i Sistemy Upravleniya*, (3), pp. 71–80 (in Russ.). doi: 10.22250/isu.2018.57.71-80.
13. Gribova, V.V., Petryaeva, M.V., Okun, D.B., Tarasov, A.V. (2018) 'Software toolkit for creating intelligent systems in practical and educational medicine', *Proc. RPC*, pp. 1–5. doi: 10.1109/RPC.2018.8482130.
14. Tsvetov, V.P. (2018) 'On attaching measuring scales', *Advanced Information Technologies: Proc. Intern. Sci. and Tech. Conf.*, pp. 341–344 (in Russ.).
15. Gribova, V.V., Kleshev, A.S., Moskalenko, F.M., Timchenko, V.A., Fedorischev, L.A., Shalfeeva, E.A. (2018) 'IACPaaS cloud platform for the development of intelligent service shells: current state and future evolution', *Software & Systems*, 31(3), pp. 527–536 (in Russ.). doi: 10.15827/0236-235X.123.527-536.

### Авторы

**Грибова Валерия Викторовна**<sup>1,2</sup>,  
д.т.н., чл.-корр. РАН,  
gribova@iacp.dvo.ru

**Ковалев Роман Игоревич**<sup>1,2</sup>, аспирант,  
научный сотрудник, koval-995@mail.ru

**Окунь Дмитрий Борисович**<sup>1</sup>, к.м.н.,  
научный сотрудник, okdm@iacp.dvo.ru

### Authors

**Valeriya V. Gribova**<sup>1,2</sup>, Dr.Sc. (Engineering),  
Corresponding Member of RAS,  
gribova@iacp.dvo.ru

**Roman I. Kovalev**<sup>1,2</sup>, Postgraduate Student,  
Research Associate, koval-995@mail.ru

**Dmitry B. Okun**<sup>1</sup>, Ph.D. (Medics),  
Research Associate, okdm@iacp.dvo.ru

<sup>1</sup> Институт автоматизации и процессов управления  
Дальневосточного отделения РАН,  
г. Владивосток, 690041, Россия

<sup>2</sup> Дальневосточный федеральный университет,  
г. Владивосток, 690922, Россия

<sup>1</sup> Institute of Automation and Control Processes  
Far Eastern Branch of the RAS,  
Vladivostok, 690041, Russian Federation

<sup>2</sup> Far Eastern Federal University, Vladivostok,  
690922, Russian Federation

## Разработка протокола передачи данных на основе комбинированного алгоритма их шифрования

О.А. Бакаева  
Д.А. Барабошкин

### Ссылка для цитирования

Бакаева О.А., Барабошкин Д.А. Разработка протокола передачи данных на основе комбинированного алгоритма их шифрования // Программные продукты и системы. 2023. Т. 36. № 3. С. 493–502. doi: 10.15827/0236-235X.142.493-502

### Информация о статье

Поступила в редакцию: 21.02.23

После доработки: 31.03.23

Принята к публикации: 12.04.2023

**Аннотация.** Передача информации с технической точки зрения невозможна без используемых в них протоколов передачи данных. Одним из основных требований, предъявляемых к протоколам такого рода, является защита данных. Самый надежный метод, обеспечивающий защиту передаваемой информации по различным каналам связи, – шифрование данных. В статье проанализированы стандартные алгоритмы шифрования: AES, RSA, протокол Диффи–Хеллмана и функция хеширования данных SHA256. Выявлены их некоторые особенности, не позволяющие в полном объеме обеспечить максимальную защиту данных при передаче. Поэтому комбинированный алгоритм шифрования данных, суть которого в применении существующих алгоритмов на разных этапах шифрования, позволит избежать проблем, возникающих при использовании одного алгоритма. Предметом данного исследования является функционирование стандартных алгоритмов шифрования: AES, RSA, протокола Диффи–Хеллмана и функции хеширования данных SHA256. Основной результат работы – протокол передачи данных, созданный на основе комбинированного алгоритма шифрования данных. Протокол включает в себя разработку структуры пакета, реализацию процессов ClientResolving и Handshake, а также различные типы структур Payload. В конце осуществляется выбор параметров алгоритмов шифрования Диффи–Хеллмана и AES. Такая последовательность разработки позволила сделать протокол передачи данных универсальным и эффективным. В статье продемонстрирована работа протокола, включающая два этапа: установка соединения и непосредственно передача данных. Практическая значимость исследования заключается в том, что созданный протокол поможет обеспечить полноту, конфиденциальность и безопасность передачи данных любого типа – текст, графика, аудиофайл.

**Ключевые слова:** протокол передачи данных, комбинированный алгоритм шифрования данных, алгоритм AES, алгоритм RSA, протокол Диффи–Хеллмана, функция хеширования данных SHA256, процесс Handshake

Современный человек постоянно использует различные средства для обмена сообщениями, файлами, новостями, решает важные задачи, связанные с профессиональной деятельностью. Огромное количество самых разных конфиденциальных данных пропускается через эти системы. Можно осуществлять аудио- и видеозвонки, отправлять графические файлы, делать покупки, получать справочную информацию, заказывать различные товары и продукты питания, оплачивать услуги и многое другое. Все это стало неотъемлемой частью повседневной жизни.

При этом необходимо защищать передаваемую клиентом информацию и данные, потеря, изменение или модификация которых может нанести ущерб как крупным компаниям, так и физическому лицу.

Целью данной работы является исследование алгоритмов шифрования и протоколов передачи данных.

Поставлены следующие задачи:

– провести анализ применения существующих протоколов передачи данных;

– разработать комбинированный алгоритм шифрования данных;

– разработать протокол передачи данных на основе комбинированного алгоритма шифрования;

– написать программную часть для реализации процессов, организующих работу протокола.

В нашей стране с конца XX века применялись мощные криптографические алгоритмы «Магма» и позже «Кузнечик». Последний достаточно часто упоминают как аналогию криптоалгоритма AES – национального алгоритма шифрования США.

В работе [1] проведен сравнительный анализ российского и американского стандартов шифрования, который доказывает их неуязвимость и надежность в случае атак злоумышленников. В качестве основного критерия сравнения использована криптографическая стойкость алгоритма.

Однако, если рассматривать симметричный алгоритм AES с точки зрения шифрования данных, можно выделить ряд недостатков [2]:

– для шифрования и дешифрования используется один и тот же ключ;

– симметричные алгоритмы часто считаются менее безопасными, так как асимметричный алгоритм опирается на закрытый ключ для дешифрования и отдельный открытый ключ для шифрования файлов;

– более длинные ключи обеспечивают более высокую криптостойкость, поэтому для шифрования может потребоваться больше времени;

– симметричные алгоритмы требуют применения безопасного способа передачи ключа желаемому получателю.

В [3] отмечаются преимущества другого криптоалгоритма – RSA: возможность передачи приватной информации по незащищенным каналам связи без предварительной передачи секретных ключей и обеспечение цифровой подписи финансовых документов. В основе надежности RSA лежит задача разложения чисел на множители (задача факторизации). Так как в настоящее время не существует эффективного метода решения данной проблемы, алгоритм считается надежным. На этой особенности и базируется безопасность его использования. Однако во многих публикациях [4, 5] также отмечается, что неверный выбор параметров шифра RSA может привести к уменьшению его криптостойкости. В некоторых случаях открытый и закрытый ключи могут полностью совпасть, и тогда абонент случайно опубликует секретный ключ, что сделает криптосистему уязвимой.

До недавнего времени алгоритм, предложенный Диффи и Хеллманом (Diffie–Hellman, DH), считался самым безопасным и понятным. Он позволял двум сторонам создавать общий секретный ключ на основе открытых параметров друг друга без необходимости обмена секретными параметрами. Степень защиты зависит от сложности нахождения заданных целых чисел по определенному модулю. Сейчас алгоритм DH имеет функцию аутентификации (для борьбы с такими угрозами, как атака «человек посередине»). Впоследствии алгоритм DH стал основой других, более совершенных алгоритмов, таких как протоколы Нидхема–Шредера и MQV. Развитие протокола DH привело к созданию средств для включения трех сторон в генерацию сеансового ключа. С течением времени появилась угроза со стороны квантового компьютера. В связи с этим остро встал вопрос разработки более устойчивых алгоритмов генерации ключей шифрования. Так, например, в [6] приводится альтернативный протокол DH, ко-

торый заключается в использовании десятичной части трансцендентных чисел, а не больших конечных целых чисел.

Многие специалисты по защите данных особо выделяют функцию хеширования данных SHA256 [7]. Ее основные достоинства состоят в однонаправленности в отличие от симметричных алгоритмов шифрования. У алгоритма SHA-256 также существует  $2^{256}$  возможных вариантов хеш-значений, поэтому в случае какой-либо атаки, если производить перебор всех комбинаций для поиска нужной, на взлом потребуется большое количество операций и времени.

Эти алгоритмы зарекомендовали себя как эффективные и справляющиеся со своим классом задач. Но при этом каждый из них имеет ряд недостатков с точки зрения безопасности передачи и хранения данных пользователя. Таким образом, не существует универсального лучшего криптоалгоритма. Следовательно, необходимы дальнейшие научные изыскания в этой области.

Ценность алгоритмов заключается в том, что они являются основой создания протоколов передачи данных. А непосредственно сами протоколы используются практически во всех сферах жизни человека, где передаются данные (политика, экономика, банковское дело и биржа, образование и др.).

Во многих работах [8, 9] подчеркивается, что в настоящий момент остаются вопросы к надежности любого протокола (даже с несколькими видами ключей для шифрования передаваемой информации и разнообразием методов аутентификации), подтвержденные большим количеством уязвимостей [10]. Это проявляется в следующем:

– UDP: пакеты данных могут прийти не в полном объеме;

– FTP: используется только для передачи файлов;

– AMF: подвержен атакам;

– RTMP и RTSP: ограниченный функционал (передача видео);

– XMPP: зависимость от сервера и версии клиента;

– MTproto: логин и пароль при подключении к прокси передается в открытом виде.

Перечисленными недостатками алгоритмов и протоколов обусловлены следующие основные проблемы безопасности передачи данных:

– надежность аутентификации;

– создание и хранение public и private ключей;

- защищенность данных для авторизации пользователей и переписки;
- неудовлетворительная защищенность данных вследствие использования одного алгоритма шифрования данных;
- уязвимость алгоритма при атаках злоумышленников;
- утечка данных;
- передача информации в неполном объеме из-за обрыва канала.

Идеального протокола с точки зрения обеспечения абсолютной безопасности и конфиденциальности обмена сообщениями не существует, следовательно, разработка собственного протокола передачи данных с сервером с открытым исходным кодом может решить некоторые проблемы, связанные с безопасностью и хранением информации [11].

Практическая значимость работы заключается в разработке собственного протокола передачи данных. Это поможет обеспечить полноту, конфиденциальность и безопасность процесса передачи.

### Анализ алгоритмов шифрования данных

В основе протоколов передачи данных лежат известные стандартные алгоритмы шифрования: AES, RSA, протокол DH и функция хеширования данных SHA256 [12].

Приведем краткую характеристику этих алгоритмов.

AES (Advanced Encryption Standards) – симметричный алгоритм блочного шифрования, который принят в качестве стандарта шифрования правительством США. Может обрабатывать данные блоками размером 128 бит, используя ключи шифрования длиной 128, 192 и 256 бит. Блоки определяют ввод открытого текста и вывод зашифрованного текста [2, 13].

RSA – криптографический алгоритм с открытым ключом, основанный на вычислительной сложности задачи факторизации больших целых чисел. Является самым распространенным асимметричным алгоритмом шифрования в мире. В обобщенном варианте содержит четыре шага: генерация ключей, распределение ключей, шифрование, расшифровка.

Криптосистема RSA используется для передачи ключей к симметричным криптосистемам, формирования цифровой подписи важных документов, создания почтовых клиентов [14].

Протокол DH – криптографический протокол, позволяющий двум и более сторонам по-

лучать общий секретный ключ, используя незащищенный от прослушивания канал связи. Полученный ключ используется для генерации ключей симметричного шифрования и дальнейшего обмена данными [12, 15].

Алгоритм SHA-256 (Secure Hash Algorithm 256-bit) – однонаправленная функция для создания цифровых отпечатков фиксированной длины (256 бит) из входных данных размером до 2,31 эксабайт ( $2^{64}$  бит), являющаяся частным случаем алгоритма из семейства криптографических алгоритмов SHA-2. С учетом уровня развития современных вычислительных мощностей алгоритм SHA-256 – одна из самых безопасных функций хеширования.

### Разработка комбинированного алгоритма передачи пакета данных

Как известно, существуют различные пакеты передачи данных. Каждый из них отличается структурой и содержанием.

Прежде всего для установки соединения между клиентами, желающими обменяться информацией, осуществляется процесс Server Initialization – клиент отправляет пакет на сервер, тем самым показывая, что он находится в сети.

Условимся, что initiator и responder – это два отдельных клиента, которые будут общаться между собой.

После того как клиент прошел инициализацию с сервером, начинается процесс Client Resolving – первоначальное соединение клиентов, с помощью которого проверяется возможность установки связи. Он состоит из следующих действий:

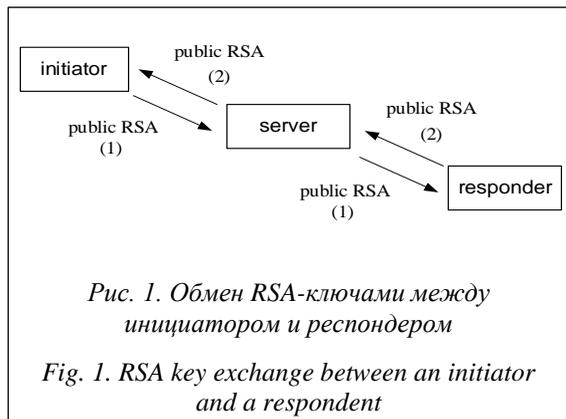
- initiator отправляет пакет INIT\_RESOLVE\_REQUEST;
- responder отправляет пакет INIT\_RESOLVE\_RESPONSE;
- соединение установлено.

После успешного завершения Client Resolving начинается процесс handshake. Он состоит из трех шагов.

**Шаг 1.** Обмен public RSA-ключами между initiator и responder (рис. 1). Идентичность проверяется сравнением хеша от присланного public RSA собеседника с sender\_id. Если они не совпадают, handshake прерывается [16].

Получение хеша и проверка целостности получаемого пакета осуществляются с помощью алгоритма SHA-256.

В процессе handshake происходят обмен public RSA-ключами между initiator и responder,



генерация пар public и private DH keys на сторонах клиентов. Генерируется sharedkey с помощью пары private DH key и public DH key, создаются три криптоматериала. Они хешируются с помощью SHA-256, и от результатов хеша получаются пары encryption\_key (AES key для initiator), decryption\_key для responder и encryption\_key для responder, decryption\_key для initiator [17].

В процессе шифрования при отправке сообщения данные шифруются с помощью алгоритма AES.

**Шаг 2.** Обмен DH-ключами.

Данный алгоритм представляет собой последовательность действий [17]:

- на стороне инициатора генерируется пара public и private DH-ключей;
- public DH-ключ шифруется полученным public RSA собеседника;
- зашифрованный public DH-ключ отправляется собеседнику;
- сервер пересылает public DH-ключ собеседнику;
- на стороне респондера генерируется пара public и private DH-ключей;
- public DH-ключ шифруется полученным public RSA собеседника;
- зашифрованный public DH-ключ отправляется собеседнику, затем генерируется sharedkey с помощью private DH-ключа от респондера и public DH-ключа от инициатора;
- сервер пересылает public DH-ключ собеседнику.

Далее инициатор принимает сообщение сервера и генерирует sharedkey.

После предпоследней операции responder создает криптоматериал. К sharedkey добавляются байты 0, 1, 2:

- 1 – shared key | 0
- 2 – shared key | 1
- 3 – shared key | 2

Полученные криптоматериалы хешируются с помощью SHA-256, и от результатов хеша получают:

- 1 – session\_id;
- 2 – AES key (encryption\_key для initiator, decryption\_key для responder);
- 3 – AES key (encryption\_key для responder, decryption\_key для initiator).

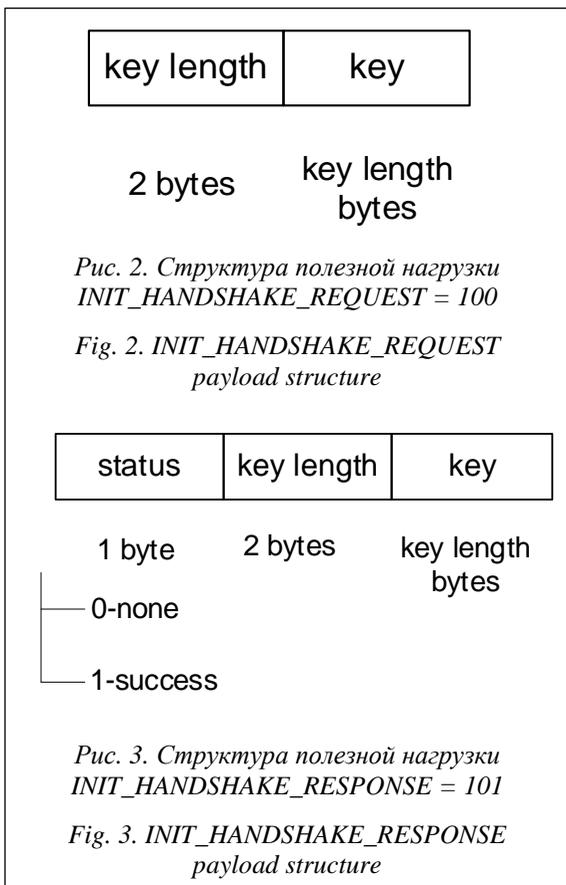
После завершения шага 2 устанавливается защищенное соединение и начинается шаг 3 процесса Handshake, на котором между инициатором и респондером происходит обмен полученными session\_id.

**Шаг 3.** Полученный session\_id отправляется собеседнику. Собеседник отправляет свой полученный session\_id. Происходит сравнение session\_id собеседников. Если они одинаковые, соединение будет установлено. Если различные, соединение обрывается.

Для установки защищенного соединения были разработаны два пакета, представленные на рисунках 2 и 3 соответственно.

При отправке сообщения передаваемые данные шифруются с помощью алгоритма AES.

При делении содержимого пакета у всех его частей будет один и тот же message\_id, различаться будет только part\_id.



### Алгоритм деления контента на части и отправки данных

Процесс составления пакетов.

1. Контент делится на части, то есть блоки (part\_1, part\_2, ..., part\_N-1, rest\_part) размером 1 000 байт; если размер блока меньше, оставшаяся часть заполняется нулями (пример rest\_part).

2. В начало блока добавляется поле block\_size – его длина (длина полезной нагрузки).

3. Все блоки последовательно шифруются с помощью AES (encrypted\_block).

4. Зашифрованная часть заполняется в PART\_MESSAGE\_PAYLOAD.

5. Пакет заполняется.

6. Полученный пакет отправляется на сервер, потом пересылается клиенту.

Проиллюстрируем процесс деления на примере отправляемого контента размером 13 337 байт, Part (batchsize) = 1 000 байт, Message\_id в пределах одного содержимого будет один и тот же (рис. 4).

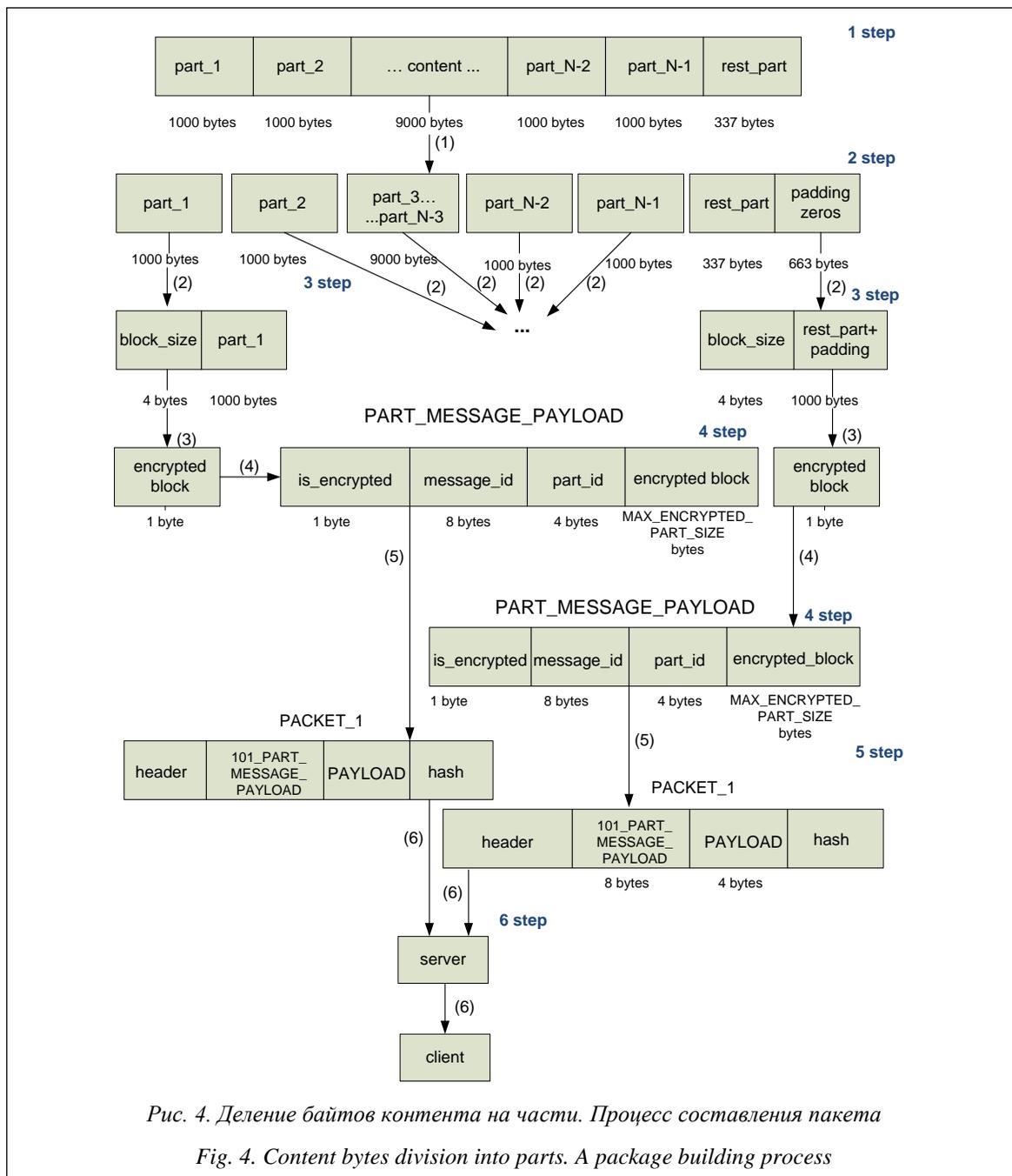


Рис. 4. Деление байтов контента на части. Процесс составления пакета

Fig. 4. Content bytes division into parts. A package building process

### Алгоритм получения исходного контента

На стороне получателя по мере поступления зашифрованных пакетов происходит обратный процесс (рис. 5).

Алгоритм получения исходного контента.

1. Из пакета (PACKET\_1, PACKET\_2, ..., PACKET\_N) берется PAYLOAD.
2. Из блока PAYLOAD берется зашифрованная часть (encrypted\_block) и расшифровывается.
3. Определяется количество байт полезной нагрузки, padding отделяется от полезной нагрузки, если она есть.
4. Из всех полученных блоков собирается исходное содержимое контента (порядок работы с блоками строго последователен).

Для осуществления процесса передачи сообщения были разработаны необходимые пакеты:

**kFullMessage = 100**

is\_encrypted – **1 байт** (зашифровано ли сообщение): 0 – нет, 1 – зашифровано, остальные – discard

message\_id – **8 байт** (id на отправку)

encrypted\_block – **MAX\_ENCRYPTED\_PART\_SIZE байт**

**kPartMessage = 101**

is\_encrypted – **1 байт** (зашифровано ли сообщение): 0 – нет, 1 – зашифровано, остальные – discard

message\_id – **8 байт** (id на отправку)

part\_id – **4 байта** (при делении контента на части каждой части по порядку присваивается part\_id)

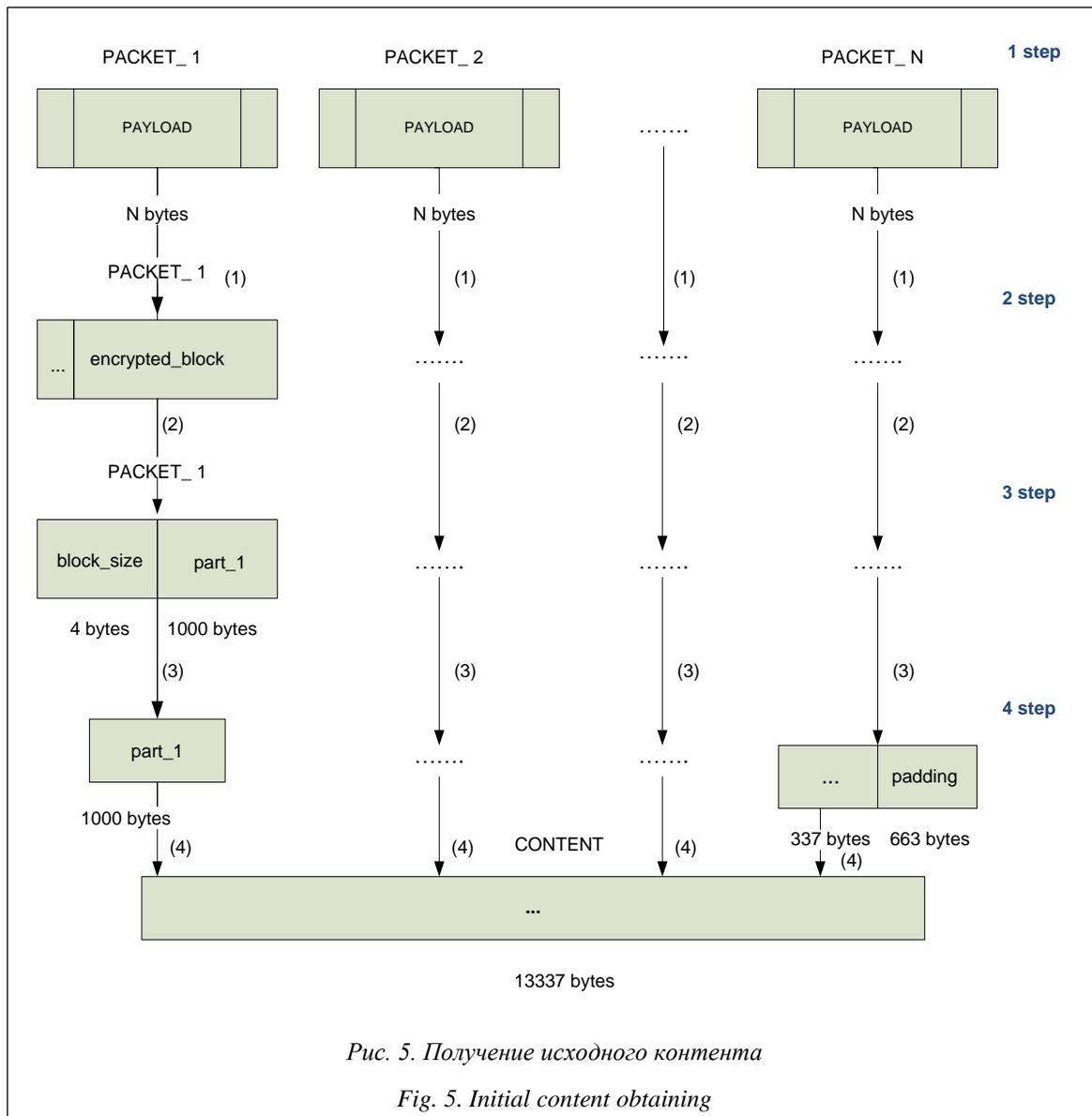


Рис. 5. Получение исходного контента

Fig. 5. Initial content obtaining

encrypted\_block – MAX\_ENCRYPTED\_PART\_SIZE байт

kPartMessageEnd = 102

message\_id – 8 байт

Комбинированный алгоритм основан на иницировании передачи RSA-ключей, а также DH-ключей, необходимых для осуществления процесса Handshake. Получение хеша и проверка целостности получаемого пакета осуществляются с помощью алгоритма SHA-256. В процессе Handshake используются асимметричный алгоритм шифрования RSA и криптографический протокол DH [17]. При отправке сообщения данные шифруются с помощью симметричного алгоритма блочного шифрования AES [18].

### Выводы

Представленный протокол передачи данных имеет определенные преимущества:

- надежная авторизация благодаря созданию RSA-ключей при установке соединения (процесс Handshake);

- защищенность протокола, обусловленная тем, что сообщения шифруются с помощью алгоритма шифрования AES-256 в режиме CTR; изменение данных сообщения в процессе его отправки невозможно, так как в таком случае сообщение нельзя расшифровать на принимающей стороне из-за режима шифрования и проверки на хеш-сумму;

- повышение показателей безопасности передачи информации вследствие применения при создании комбинированного алгоритма шифрования на разных этапах алгоритмов DH, AES, RSA и функции хеширования SHA-256;

- невозможность попадания ключей в руки злоумышленников при атаках или попытках взлома сервера, поскольку ключи не хранятся на сервере (там находятся только хеши публичных ключей), private ключи находятся только у пользователей и никуда не передаются, а публичные ключи – это условно идентификационный номер пользователя.

В аналогичных протоколах данные для авторизации пользователя можно восстановить, следовательно, они хранятся на серверах. У разработанного протокола нет такого изъяна. Данные хранятся на сторонах пользователей. Владелец сервера не сможет получить данные авторизации пользователей и переписки. Сравнительный анализ протоколов передачи данных представлен в таблице.

Таким образом, разработанный алгоритм, включающий использование вышеупомянутых алгоритмов в указанной последовательности, позволяет обеспечить высокий уровень безопасности передачи данных и тем самым сохранить приватность пользователя.

Для разработки протокола были выбраны среда программирования VisualStudio и универсальный язык программирования C++.

Для реализации алгоритмов шифрования и создания комбинированного алгоритма использовалась бесплатная библиотека Crypto++ с открытым исходным кодом и криптографическими алгоритмами с возможностью гибкой настройки.

### Демонстрация работы протокола передачи данных

После компиляции исходного кода запускаются exe-файлы – клиент и сервер протокола. При входе нужно или создать пару RSA-ключей, необходимых для процесса Handshake и генерации user\_id, или использовать уже имеющиеся. Если вход осуществляется впервые, следует создать новую пару ключей. Соединение установлено. Созданы новые user\_id пользователей (<http://www.swsys.ru/uploaded/image/2023-3/2023-3-dop/2.jpg>).

После успешной установки соединения у пользователей появятся папки self и other (<http://www.swsys.ru/uploaded/image/2023-3/2023-3-dop/3.jpg>). В них будут лежать RSA-ключи, необходимые для процесса Handshake (<http://www.swsys.ru/uploaded/image/2023-3/2023-3-dop/4.jpg>).

Из команд выбирается 1. Send text (t). Вводится сообщение, далее на клиентской и серверной сторонах отображается выполнение команды, осуществляется отправка зашифрованного текстового сообщения (<http://www.swsys.ru/uploaded/image/2023-3/2023-3-dop/5.jpg>).

Из команд выбирается 2. Sendfile (f). Отправляемый графический файл находится в папке files, далее на клиентской и серверной сторонах отображается выполнение команды, осуществляется отправка графического файла (<http://www.swsys.ru/uploaded/image/2023-3/2023-3-dop/6.jpg>). Полученный файл появится в корне папки, где находятся exe-файлы клиента и сервера.

Файл отправился. Полученный файл появился в корне папки, где находятся exe-файлы клиента и сервера (<http://www.swsys.ru/uploaded/image/2023-3/2023-3-dop/30.jpg>).

## Сравнительный анализ протоколов передачи данных

## A comparative analysis of data transfer protocols

Критерий	Стандартный протокол	Разрабатываемый протокол
Безопасность – защищенность данных (в целом)	Данные аутентификации и информация пользователей хранятся на сервере, что небезопасно	Сервер не хранит критически важные данные, что повышает безопасность
Безопасность – защищенность данных (на разных этапах)	Индивидуальна для каждого протокола. Уровень безопасности высокий, но протоколы не гарантируют 100-процентную защищенность данных (при атаках)	Структура генерации private и public ключей обеспечивает безопасность данных на этапе создания. Организация процесса Handshake обеспечивает безопасность данных на этапе соединения пользователей
Использование собственного сервера	Частично используется в некоторых протоколах	Реализовано. Повышает безопасность хранения данных и сохраняет приватность пользователя
Использование алгоритмов шифрования внутри протокола	Использование одного алгоритма шифрования ненадежно и часто приводит к нарушению безопасности	Используется комбинированный алгоритм шифрования (DH, RSA, AES, SHA-256), который защищает от разных атак, в том числе и man in the middle
Случаи утечки информации	Были выявлены	При функциональном тестировании протокола не были выявлены (масштабное тестирование не проводилось)
Хранение данных для авторизации	Данные для авторизации хранятся на серверах	Данные для авторизации хранятся на сторонах пользователей
Типы передаваемых данных	Некоторые протоколы работают только с одним типом данных	Любые типы данных
Целостность	Имеется (через хеш)	Имеется (через хеш)

Как видно из демонстрации работы протокола передачи данных, соединение устанавливается успешно. Текстовый и графический файлы отправляются и получаются без потери информации, что свидетельствует о работоспособности полученного протокола.

### Заключение

Многим пользователям важно знать, что их данные конфиденциальны и не передаются третьим лицам, а компаниям важна корпоративная защита внутренней информации. Не важно, какая информация передается – личная или служебная, она не должна поте-

ряться, а тем более попасть в руки мошенников.

В основе созданного протокола лежит комбинированный алгоритм шифрования данных, суть которого в последовательном применении стандартных алгоритмов шифрования на разных этапах передачи. Представленный алгоритм позволяет обеспечить максимальный уровень безопасности передачи информации и тем самым сохранить приватность пользователя.

Собственный протокол передачи данных поможет сохранить конфиденциальную информацию и при этом быть уверенным в том, что она не передается третьим лицам и доставляется в полном объеме и исходном качестве.

### Список литературы

1. Соболев М.А. Сравнительный анализ российского стандарта шифрования по ГОСТ Р 34.12–2015 и американского стандарта шифрования AES // Политехнический молодежный журнал. 2022. № 04. С. 1–13. doi: 10.18698/2541-8009-2022-4-785.

2. Архипова И.С. Криптографический алгоритм AES как средство защиты информации // Аллея науки. 2018. Т. 6. № 4. С. 83–87.
3. Алексеев А.П. Уязвимости алгоритма вычисления секретного ключа в криптосистеме RSA // Системы управления, связи и безопасности. 2015. № 3. С. 83–91. URL: [https://www.elibrary.ru/download/elibrary\\_24252379\\_56555611.pdf](https://www.elibrary.ru/download/elibrary_24252379_56555611.pdf) (дата обращения: 10.02.2023).
4. Чиченин А.Д., Портнов Е.М. Недостатки использования алгоритма шифрования RSA // Интернаука. 2020. № 25-1. С. 11–12.
5. Квист Т.Д. Криптографический алгоритм RSA и его уязвимости при неправильном использовании // Научно-исследовательская работа обучающихся и молодых ученых: матер. 74-й Всерос. науч. конф. 2022. С. 530–533.
6. Коршев М.А. Альтернативный протокол Диффи–Хеллмана // Современные научные исследования и инновации. 2021. № 2. URL: <http://web.snauka.ru/issues/2021/02/94553> (дата обращения: 28.02.2023).
7. Дремов И.С., Гирина А.Н. Использование алгоритма SHA-256 для хеширования данных // Тенденции развития науки и образования. 2022. Т. 86. № 1. С. 57–61. doi: 10.18411/trnio-06-2022-19.
8. Биджиева С.Х., Шебзухова К.В. Сетевые протоколы передачи данных: преимущества и недостатки // Тенденции развития науки и образования. 2022. Т. 86. № 1. С. 43–45. doi: 10.18411/trnio-06-2022-14.
9. Дементьев В.Е., Чулков А.А. Кибервоздействия на протоколы сетей передачи данных // Изв. ТулГУ. Технические науки. 2020. № 10. С. 245–254.
10. Лившиц И.И., Лонщик П.А. Формирование метрик для измерения результативности систем менеджмента информационной безопасности // Вестн. ИГТУ. 2016. Т. 112. № 5. С. 65–72. doi: 10.21285/1814-3520-2016-5-65-72.
11. Бакланова А.Р., Газизов А.Р. Применение защитных протоколов в системе передачи данных // Студенческий вестн. 2019. № 48-6. С. 29–31.
12. Барабошкин Д.А., Бакаева О.А. Анализ алгоритмов шифрования данных // За нами будущее: взгляд молодых ученых на инновационное развитие общества: сб. ст. науч. конф. 2022. Т. 2. С. 449–452.
13. Баймухамедов М.Ф., Жикеев А.А. Алгоритм шифрования AES как средство обеспечения информационной безопасности // Актуальные научные исследования в современном мире. 2019. № 9-1. С. 24–29.
14. Ибрагимов Б.Э. Алгоритм шифрования RSA // TNS. Сб. избр. ст. по матер. науч. конф. ГНИИ «Нацразвитие»: матер. конф. 2019. С. 230–232.
15. Давтян А.В. Система открытого распределения ключей Диффи–Хеллмана // Лучшая исследовательская работа: сб. статей. 2021. С. 76–81.
16. Абелян В.З. Криптографический алгоритм RSA // IJASCSE. 2020. № 1. С. 4–10.
17. Барабошкин Д.А., Бакаева О.А. Разработка комбинированного алгоритма шифрования мультимедийных данных в процессе их передачи // Математическое моделирование, численные методы и комплексы программ: сб. тр. X Междунар. науч. молодежн. школы-семинара им. Е.В. Воскресенского. 2022. С. 27–31. URL: <https://conf.svtmo.ru/files/2022/papers/paper05.pdf> (дата обращения: 28.02.2023).
18. Тищенко А.А., Лысов Д.А. Алгоритм шифрования AES, вопросы реализации и безопасности // Экономическая безопасность: правовые, экономические, экологические аспекты: сб. тр. 2016. С. 116–118.

## Developing a data transfer protocol based on a combined data encryption algorithm

Olga A. Bakaeva  
Dmitry A. Baraboshkin

### For citation

Bakaeva, O.A., Baraboshkin, D.A. (2023) 'Developing a data transfer protocol based on a combined data encryption algorithm', *Software & Systems*, 36(3), pp. 493–502 (in Russ.). doi: 10.15827/0236-235X.142.493-502

### Article info

Received: 21.02.23

After revision: 31.03.23

Accepted: 12.04.2023

**Abstract.** From a technical point of view, information transfer is impossible without using data transfer protocols. One of the main requirements for such protocols is data protection. The most reliable method that ensures the protection of information transmitted over various communication channels is data encryption. The article analyzes standard encryption algorithms: AES, RSA, Diffie–Hellman protocol and SHA256 data hash function. It identifies some of their features that do not allow ensuring maximum data protection during their transfer completely. Therefore, the development of a combined data encryption algorithm, the essence of which is the use of existing algorithms at different stages of encryption, will help to avoid the problems that arise when using a single protocol. The research subject is functioning of standard encryption algorithms: AES, RSA, Diffie–Hellman protocol and SHA256 data hash function. The main result of the work is creating a data transfer protocol based on a combined data encryption algorithm. The protocol includes the development of the packet structure, the implementation of the Client Resolving and Handshake processes, as well as various types of Payload

structures. At the end, the parameters of the DH (Diffie-Hellman) and AES encryption algorithms are selected. This sequence of development made it possible to make this data transfer protocol universal and efficient. The article demonstrates the protocol operation that consists of two stages: establishing a connection and data transfer. The practical significance of the work is in the fact that the developed data transfer protocol will help to ensure the completeness, confidentiality and security of any type data transfer (text, graphics, audio file).

**Keywords:** data transfer protocol, combined data encryption algorithm, AES algorithm, RSA algorithm, Diffie-Hellman protocol, SHA256 data hashing function, Handshake process

### Reference List

1. Sobolev, M.A. (2022) 'Comparative analysis of Russian GOST R 34.12-2015 encryption standard and American encryption standard AES', *Politechnical Student J.*, (04), pp. 1–13 (in Russ.). doi: 10.18698/2541-8009-2022-4-785.
2. Arkhipova, I.S. (2018) 'AES cryptographic algorithm as a means of information protection', *Alley of Science*, 6(4), pp. 83–87 (in Russ.).
3. Alekseev, A.P. (2015) 'Vulnerabilities algorithm for computing the secret key in the RSA cryptosystem', *Systems of Control, Communication and Security*, (3), pp. 83–91, available at: [https://www.elibrary.ru/download/elibrary\\_24252379\\_56555611.pdf](https://www.elibrary.ru/download/elibrary_24252379_56555611.pdf) (accessed February 10, 2023) (in Russ.).
4. Chichenin, A.D., Portnov, E.M. (2020) 'Disadvantages of using the RSA encryption algorithm', *Interscience*, (25-1), pp. 11–12 (in Russ.).
5. Kvist, T.D. (2022) 'RSA cryptographic algorithm and its vulnerabilities when misusing', *Proc. Proc. 74th All-Russ. Sci. Conf. "Research Work of Students and Young Scientists"*, pp. 530–533 (in Russ.).
6. Korshev, M.A. (2021) 'Alternative Diffie–Hellman protocol', *Modern Sci. Researches and Innovations*, (2), available at: <http://web.snauka.ru/issues/2021/02/94553> (accessed February 28, 2023) (in Russ.).
7. Dremov, I.S., Girina, A.N. (2022) 'Using the SHA-256 algorithm for hashing data', *Trends in the Development of Science and Education*, 86(1), pp. 57–61 (in Russ.). doi: 10.18411/trnio-06-2022-19.
8. Bidzhieva, S.Kh., Shebzukhova, K.V. (2022) 'Network data transfer protocols: Advantages and disadvantages', *Trends in the Development of Science and Education*, 86(1), pp. 43–45 (in Russ.). doi: 10.18411/trnio-06-2022-14.
9. Dementev, V.E., Chulkov, A.A. (2020) 'Cyber attacks on data network protocols', *Proc. of the TSU. Tech. Sci.*, (10), pp. 245–254 (in Russ.).
10. Livshits, I.I., Lontsikh, P.A. (2016) 'Formation of metrics to measure information security management system efficiency', *Proc. of ISTU*, 112(5), pp. 65–72 (in Russ.). doi: 10.21285/1814-3520-2016-5-65-72.
11. Baklanova, A.R., Gazizov, A.R. (2019) 'Application of security protocols in the data transmission system', *Student Bull.*, (48-6), pp. 29–31 (in Russ.).
12. Baraboshkin, D.A., Bakaeva, O.A. (2022) 'Analysis of data encryption algorithms', *Proc. Sci. Conf. The Future is Behind us: A View of Young Scientists on the Innovative Development of Society*, 2, pp. 449–452 (in Russ.).
13. Baimukhamedov, M.F., Zhikeev, A.A. (2019) 'AES encryption algorithm as a means of ensuring information security', *Actual Sci. Research in the Modern World*, (9-1), pp. 24–29 (in Russ.).
14. Ibraimov, B.E. (2019) 'RSA encryption algorithm', *Proc. TNS. Collection of Selected Articles Based on the Materials of Sci. Conf. of the GNII "National Development"*, pp. 230–232 (in Russ.).
15. Davtyan, A.V. (2021) 'Diffie–Hellman public key distribution system', *Proc. Best Research Paper*, pp. 76–81 (in Russ.).
16. Abelyan, V.Z. (2020) 'RSA cryptographic algorithm', *IJASCSE*, (1), pp. 4–10 (in Russ.).
17. Baraboshkin, D.A., Bakaeva, O.A. (2022) 'The development of a combined algorithm for encrypting multimedia data during transmission', *Proc. X Int. Sci. Youth School-seminar Math. Modeling, Numerical Methods and Software Packages*, pp. 27–31, available at: <https://conf.svmu.ru/files/2022/papers/paper05.pdf> (accessed February 28, 2023) (in Russ.).
18. Tishchenko, A.A., Lysov, D.A. (2016) 'AES encryption algorithm, implementation and security issues', *Proc. Int. Conf. Economic Security: Legal, Economic, Environmental Aspects*, pp. 116–118 (in Russ.).

### Авторы

**Бакаева Ольга Александровна**<sup>1</sup>, к.т.н.,  
доцент кафедры систем автоматизированного  
проектирования, helga\_rm@rambler.ru  
**Барабошкин Дмитрий Александрович**<sup>1</sup>,  
магистр, torcktaer@yandex.ru

### Authors

**Olga A. Bakaeva**<sup>1</sup>, Ph.D. (Engineering),  
Associate Professor,  
helga\_rm@rambler.ru  
**Dmitry A. Baraboshkin**<sup>1</sup>, Master of Science,  
torcktaer@yandex.ru

<sup>1</sup> Национальный исследовательский Мордовский  
государственный университет им Н.П. Огарева,  
г. Саранск, 430005, Россия

<sup>1</sup> National Research N.P. Ogarev Mordovian  
State University, Saransk,  
430005, Russian Federation

## СО Д Е Р Ж А Н И Е

<b>Оленев А.А., Калмыков И.А., Киричек К.А.</b> Построение совершенных нормальных форм булевых функций для схемотехнических реализаций протоколов аутентификации с использованием Maple .....	349
<b>Ulyanov S.V., Ulyanov V.S.</b> Fast quantum search algorithm modelling on conventional computers: Information analysis of the halting problem.....	361
<b>Мунтян Е.Р.</b> Разработка алгоритма пропорционального разделения ГН-графа для формирования зон влияния объектов в сложных технических системах .....	378
<b>Саяпин О.В., Тиханычев О.В., Безвесильная А.А., Чискидов С.В.</b> Об одной тенденции развития алгоритмов, реализуемых в системах поддержки принятия решений .....	388
<b>Грачев В.Г., Ушаков А.С.</b> Исследование свойств конкурентного двоичного дерева.....	398
<b>Вяткин С.И., Долговесов Б.С.</b> Метод деформации функционально заданных объектов с применением графических процессоров .....	414
<b>Игнатьев В.В.</b> Компенсация разных типов неопределенностей при управлении техническим объектом с помощью интеллектуальных регуляторов .....	423
<b>Баранов А.В., Корепанов П.М., Лепешев И.А.</b> Контроль подключений USB-устройств в локальной вычислительной сети компьютеров под управлением Astra Linux SE .....	432
<b>Федулов А.С., Лазарев А.И.</b> Организация адаптивной маршрутизации данных в электроэнергетических комплексах с использованием онтологических нечетких классификаторов.....	442
<b>Емельянов В.А., Черный С.Г., Емельянова Н.Ю.</b> Архитектура системы поддержки принятия решений в процессе мониторинга технического состояния критического оборудования .....	451
<b>Соломаха Г.М., Тулуева В.А., Хижняк С.В.</b> Программный комплекс планирования производства и управления запасами.....	459
<b>Сазанов В.А., Хлобыстова А.О., Абрамов М.В.</b> Разработка чат-бота для отслеживания расписания учебных занятий в университете .....	466
<b>Кабанова В.В., Логунова О.С., Наркевич М.Ю., Корниенко В.Д.</b> Программное обеспечение для автоматизированного обнаружения и оценки разрушений соединительных швов зданий.....	474
<b>Грибова В.В., Ковалев Р.И., Окунь Д.Б.</b> Система назначения персонифицированного лечения по аналогии на основе гибридного способа извлечения прецедентов.....	486
<b>Бакаева О.А., Барабошкин Д.А.</b> Разработка протокола передачи данных на основе комбинированного алгоритма их шифрования .....	493

## C O N T E N T

<b><i>Olenev, A.A., Kalmykov, I.A., Kirichek, K.A.</i></b>	
Constructing perfect normal forms of Boolean functions for circuit implementations of authentication protocols using Maple.....	349
<b><i>Ulyanov, S.V., Ulyanov, V.S.</i></b>	
Fast quantum search algorithm modelling on conventional computers: Information analysis of the halting problem.....	361
<b><i>Muntyan, E.R.</i></b>	
Developing a GH-graph proportional separation algorithm to form of object influence zones in complex technical systems .....	378
<b><i>Sayapin, O.V., Tikhanychev, O.V., Bezvesilnaya, A.A., Chiskidov, S.V.</i></b>	
On one trend in the development of algorithms implemented in decision support systems .....	388
<b><i>Grachev, V.G., Ushakov, A.S.</i></b>	
A research on concurrent binary tree properties .....	398
<b><i>Vyatkin, S.I., Dolgovesov, B.S.</i></b>	
A method of deformation of functionally defined objects using graphics processing units .....	414
<b><i>Ignatyev, V.V.</i></b>	
Compensation of various types of uncertainties when controlling a technical object using intelligent controllers .....	423
<b><i>Baranov, A.V., Korepanov, P.M., Lepeshev, I.A.</i></b>	
USB connections control in the local network of computers running under Astra Linux SE .....	432
<b><i>Fedulov, A.S., Lazarev, A.I.</i></b>	
Organization of adaptive data routing in electric power complexes using ontological fuzzy classifiers.....	442
<b><i>Yemelyanov, V.A., Chernyi, S.G., Yemelyanova, N.Yu.</i></b>	
An architecture of the decision support system for monitoring a technical condition of critical equipment .....	451
<b><i>Solomakha, G.M., Tulueva, V.A., Khizhnyak, S.V.</i></b>	
Software system for production planning and inventory management .....	459
<b><i>Sazanov, V.A., Khlobystova, A.O., Abramov, M.V.</i></b>	
Developing a chatbot to monitor a university's academic timetable.....	466
<b><i>Kabanova, V.V., Logunova, O.S., Narkevich, M.Yu., Kornienko, V.D.</i></b>	
Software for automated detection and assessment of building seam failures .....	474
<b><i>Gribova, V.V., Kovalev, R.I., Okun, D.B.</i></b>	
The system for prescribing personalized treatment by case-based reasoning using a hybrid precedent extraction method.....	486
<b><i>Bakaeva, O.A., Baraboshkin, D.A.</i></b>	
Developing a data transfer protocol based on a combined data encryption algorithm.....	493

## *Внимание подписчиков*

Научный журнал  
«Программные продукты и системы»  
распространяется по подписке  
через подписное агентство «УП УРАЛ-ПРЕСС»  
Индекс издания  
в электронном каталоге агентства 70799

*Телефон редакции: (4822) 39-91-49*  
*Электронная почта: red@cps.tver.ru*  
*Сайт журнала: www.swsys.ru*