

# Программные продукты и системы

DOI 10.15827/0236-235X

/ О платформе обмена  
научными датасетами

/ Моделирование  
температурных  
полей

/ Автоматизированное  
выявление хайповых  
технологий

/ Оптимизация  
техпроцессов  
раскрытия деталей



№ 1  
ТОМ 38  
2025



## Уважаемые читатели!

Очередной выпуск нашего журнала открывают статьи, посвященные памяти известного ученого Сергея Юрьевича Соловьева, доктора физико-математических наук, профессора Московского государственного университета им. М.В. Ломоносова, автора фундаментальных работ в области инженерии знаний, алгебры и теории формальных грамматик. В феврале 2025 года ему исполнилось бы 70 лет.

В таланте С.Ю. Соловьева как ученого гармонично сочетались интуитивная и логическая составляющие. Его научные результаты отличает безусловная оригинальность мышления. Сергей Юрьевич глубоко понимал и ценил эстетическую составляющую научной теории. Интуиция не раз подсказывала ему оригинальные идеи, позволявшие решать сложные задачи.

Свои талантливые идеи С.Ю. Соловьев подкреплял строгим научным обоснованием. Тексты его публикаций безупречны с научной точки зрения, в них он виртуозно применял математический аппарат на стыке различных направлений. Столь же безупречны архитектура и код разработанных им программных систем.

Педагогический талант Сергея Юрьевича проявился в его уникальных лекционных курсах по программированию, теории алгоритмов, экспертным системам, информационным банковским технологиям и интернет-технологиям, которые он читал в ведущих вузах Москвы и, конечно, в родном Московском университете, где прошел путь от студента до заведующего кафедрой факультета вычислительной математики и кибернетики, воспитал учеников, с которыми щедро делился знаниями, научными идеями, вниманием и душевной теплотой.

Являясь авторитетным ученым, Сергей Юрьевич Соловьев активно участвовал в научно-экспертной деятельности: более пятнадцати лет был членом экспертного совета ВАК при Минобрнауки России по управлению, вычислительной технике и информатике. Награжден Почетной грамотой Президиума ВАК за большие заслуги в работе по аттестации научных и научно-педагогических кадров. Многие годы работал экспертом и членом издательского совета РФФИ, экспертом Российского научного фонда, был членом редколлегий нескольких научных журналов, членом диссертационных советов.

В номере журнала представлены также разработки других авторов, посвященные математическому моделированию, реализации моделей нейро-нечеткой сети, проектированию платформы обмена научными датасетами, автоматизации инженерных расчетов оборудования.

*А.Н. Сотников, д.ф.-м.н., профессор,  
заместитель главного редактора журнала*

---

Научно-исследовательский институт  
«Центрпрограммсистем»

---

# Программные продукты и системы

НАУЧНЫЙ ЖУРНАЛ

2025, том 38, № 1  
(год издания тридцать восьмой)

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

## SOFTWARE & SYSTEMS

Research Journal

2025, vol. 38, no. 1

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

---

Research Institute CENTERPROGRAMSYSTEM

---

## © ПРОГРАММНЫЕ ПРОДУКТЫ И СИСТЕМЫ

Научный журнал

2025. Т. 38. № 1

DOI: 10.15827/0236-235X.149

Главный редактор

**Г.И. САВИН**, академик РАН

Научные редакторы номера:

**Н.А. СЕМЕНОВ**, д.т.н., профессор

**Т.М. ТАТАРНИКОВА**, д.т.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, главный научный сотрудник НИЦ «Курчатовский институт» (г. Москва, Россия)

**Афанасьев А.П.** – д.ф.-м.н., профессор Московского физико-технического института (г. Москва, Россия)

**Баранов А.В.** – к.т.н., доцент, начальник отдела НИЦ «Курчатовский институт» (г. Москва, Россия)

**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)

**Кольчугина Е.А.** – д.т.н., профессор Пензенского государственного университета (г. Пенза, Россия)

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

**Мамросенко К.А.** – к.т.н., доцент, руководитель Центра визуализации и спутниковых информационных технологий НИИСИ (г. Москва, Россия)

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

**Рыбаков А.А.** – к.ф.-м.н., начальник отдела НИЦ «Курчатовский институт» (г. Москва, Россия)

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

**Татарникова Т.М.** – д.т.н., 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 (г. Москва, Россия)

**Шабанов Б.М.** – д.т.н., чл.-корр. РАН, заместитель директора НИЦ «Курчатовский институт» (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

г. Тверь, просп. Николая Корыткова, д. 3а,  
170024, Россия

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 16.03.2025 г.

Отпечатано ИПШ «Фактор и К»

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

Выпускается один раз в квартал

Год издания тридцать восьмой

Формат 60×84 1/8. Объем 176 стр.

Заказ № 2. Тираж 1000 экз. Цена 550,00 руб.

© SOFTWARE & SYSTEMS

Research Journal

2025, vol. 38, no. 1

DOI: 10.15827/0236-235X.149

**Editor-in-chief**

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

**Science editors of the issue:**

**N.A. SEMENOV**, Dr.Sci. (Engineering), Professor

**T.M. TATARNIKOVA**, Dr.Sci. (Engineering), 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 ПИИ № ФС 77-77843

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

**ISSN 2311-2735** (online)

## EDITORIAL BOARD

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

**Sotnikov A.N.** – Deputy Editor-in-Chief, Dr.Sci. (Physics and Mathematics), Professor, Chief Researcher of the National Research Centre “Kurchatov Institute” (Moscow, Russian Federation)

**Afanasyev A.P.** – Dr.Sci. (Physics and Mathematics), Professor of the Moscow Institute of Physics and Technology, (Moscow, Russian Federation)

**Baranov A.V.** – Cand. of Sci. (Engineering), Associate Professor, Head of Department of the National Research Centre “Kurchatov Institute” (Moscow, Russian Federation)

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

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

**Elizarov A.M.** – Dr.Sci. (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.Sci. (Engineering), Professor of the National Research University “Moscow Power Engineering Institute” (Moscow, Russian Federation)

**Kol’chugina E.A.** – Dr.Sci. (Engineering), Professor of the Penza State University (Penza, Russian Federation)

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

**Mamrosenko K.A.** – Cand. of Sci. (Engineering), Associate Professor, Head of the Center of Visualization and Satellite Information Technologies SRISA (Moscow, Russian Federation)

**Rybakov A.A.** – Cand. of Sci. (Physics and Mathematics), Head of Department of the National Research Centre “Kurchatov Institute” (Moscow, Russian Federation)

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

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

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

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

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

**Shabanov B.M.** – Dr.Sci. (Engineering), Corresponding Member of RAS, Deputy Director of National Research Center “Kurchatov Institute” (Moscow, Russian Federation)

**Yazenin A.V.** – Dr.Sci. (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 16.03.2025

Printed in printing-office “Faktor i K”

Krylova St. 26, Tver, 170100, Russian Federation

Published quarterly. 38th year of publication

Format 60x84 1/8. Wordage 176 pages

Prod. order № 2. Circulation 1000 copies. 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 четырех, количество статей одного автора в номере, включая соавторство, – не более двух. Список литературы, наличие которого обязательно, должен включать не менее 10 пунктов.

Необходимы также содержательная структурированная аннотация (не менее 200 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, НИЦ «Курчатовский институт» (г. Москва) и НИИ «Центрпрограммсистем» (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).

## Методы и средства извлечения терминов из текстов для терминологических задач

Е.И. Большакова <sup>1</sup>✉, В.В. Семак <sup>1</sup>

<sup>1</sup> Московский государственный университет им. М.В. Ломоносова,  
г. Москва, 119991, Россия

### Ссылка для цитирования

Большакова Е.И., Семак В.В. Методы и средства извлечения терминов из текстов для терминологических задач // Программные продукты и системы. 2025. Т. 38. № 1. С. 5–16. doi: 10.15827/0236-235X.149.005-016

### Информация о статье

Группа специальностей ВАК: 1.2.3

Поступила в редакцию: 18.08.2024

После доработки: 14.09.2024

Принята к публикации: 29.09.2024

**Аннотация.** В статье рассматривается современное состояние области автоматического извлечения терминов из специализированных текстов на естественном языке, включая научно-технические документы. К числу актуальных практических приложений методов и средств извлечения терминов из текстов относятся создание терминологических словарей, тезаурусов и глоссариев предметных областей, а также выявление ключевых слов и построение предметных указателей для узкоспециализированных документов. Представлен обзор подходов к автоматическому распознаванию и извлечению терминологических слов и словосочетаний, охватывающих традиционные статистические методы, а также методы на основе машинного обучения, включая обучение по признакам терминов и с применением современных нейросетевых языковых моделей. Проводится сравнение подходов, в том числе оценки качества распознавания и извлечения терминов, указываются наиболее известные программные средства для автоматизации извлечения терминов в рамках статистического подхода и обучения по признакам. Описываются исследования, проведенные авторами для распознавания терминов на базе нейросетевых языковых моделей применительно к обработке научных текстов по математике и программированию на русском языке. Кратко характеризуется набор данных с терминологической разметкой, созданный для обучения программных моделей распознавания терминов и охватывающий данные семи близких предметных областей. Программные модели разработаны на основе предобученной нейросетевой модели BERT с ее дообучением двумя способами: как бинарного классификатора терминов-кандидатов (предварительно извлеченных из текстов) и как классификатора для последовательной разметки терминологических слов в обрабатываемых текстах. Для разработанных моделей экспериментально определены оценки качества распознавания терминов, проведено сравнение со статистическим методом. Лучшие результаты продемонстрировали модели бинарной классификации, существенно превосходя остальные рассмотренные подходы. Проведенные эксперименты показывают применимость обученных моделей и для текстов близкой научной области.

**Ключевые слова:** автоматическая обработка текстов специализированных областей, терминологические слова и словосочетания, автоматическое извлечение терминов, машинное обучение для распознавания терминов, извлечение терминов на базе языковой модели BERT

**Введение.** Автоматическое извлечение терминов из текстов (*Automatic Term Extraction*, АТЕ, или *Automatic Term Recognition*, АТР) – одна из традиционных задач в области автоматической обработки текстов [1, 2]. Ее актуальность обусловлена стремительным развитием науки и техники и появлением в связи с этим новых терминов – слов и словосочетаний, обозначающих понятия разных предметных областей. Хотя задача АТЕ изучается более трех десятилетий, ее решения не достигают высокого качества, как во многих других задачах автоматической обработки текстов. Прежде всего это объясняется сложностью и неоднозначностью критерия терминологичности, который относится к области семантики и трудно формализуется и применяется на практике человеком.

В то же время терминология специальных предметных областей представляет собой важ-

ный пласт знаний, необходимых при решении многих прикладных задач. Методы АТЕ применяются для автоматизированного построения – терминологических словарей, тезаурусов и онтологий предметных областей по коллекциям специализированных текстов [3];

– глоссариев – перечней терминов с их определениями для проблемно-ориентированных областей и узкоспециализированных документов [4, 5];

– предметных указателей для научно-технических книг и документов [6].

Задача АТЕ также важна для улучшения методов машинного перевода специализированных текстов [7].

К настоящему моменту известны следующие подходы к автоматическому распознаванию и извлечению терминов:

– ставший традиционным статистический подход, основанный на ранжировании терми-

нов-кандидатов по терминологичности с помощью статистических мер и применяемый в основном для обработки текстовых коллекций [8–10];

– подход на основе машинного обучения бинарного классификатора термин/нетермин (для предварительно выделенных кандидатов) [11, 12], позволяющий выявить наилучшую комбинацию признаков (лингвистических и статистических) для распознавания терминов;

– подход последовательной разметки слов текста, при котором на базе машинного обучения строится модель для распознавания терминов прямо в тексте, путем выявления и разметки входящих в термины слов [13];

– подход с применением современных нейросетевых трансформерных языковых моделей (например, BERT [14]) для машинного обучения, при котором вместо набора признаков для обучения выступает контекстная информация, воплощенная в векторах слов (эмбедингах) применяемой языковой модели [15–18].

Последний подход применяется либо для бинарной классификации термин/нетермин, либо для последовательной разметки терминов в обрабатываемых текстах. В ряде работ он продемонстрировал лучшие результаты решения задачи АТЕ (предположительно, за счет глубокого предобучения применяемых нейросетевых языковых моделей на больших массивах неразмеченных текстов), однако требует дальнейшего исследования, поскольку эксперименты были относительно немногочисленны и проводились лишь для некоторых языков текста (преимущественно английского). Одна из возникающих проблем связана с недостатком открытых текстовых корпусов с эталонной терминологической разметкой, необходимых для проведения экспериментов по обучению моделей АТЕ и их оценки. Особенно острая ситуация сложилась для русского языка, работ по применению указанного подхода практически нет.

Отдельной темой для исследований является применимость моделей АТЕ, обученных для текстов определенной предметной области, для извлечения терминов в текстах из другой области без потери качества, которая обычно бывает при машинном обучении на признаках.

В настоящей работе задача АТЕ рассматривается применительно к русскому языку в рамках подхода на базе нейросетевых языковых моделей. Основная цель – экспериментально оценить и сравнить два способа распознавания терминов на основе нейросетевой модели BERT:

– бинарный классификатор для распознавания терминов с использованием контекста в виде объемлющего предложения;

– машинный классификатор, распознающий термины в тексте путем последовательной разметки входящих в них слов.

Обучение классификаторов, их оценка и сравнение проводились на одном и том же размеченном наборе данных, созданном для данной задачи из русскоязычных учебно-научных текстов. Набор данных включал термины из нескольких областей математики и программирования, что позволило оценить качество моделей при переходе от одной области к другой (от математики к программированию и наоборот). Дополнительно сопоставлялось качество извлечения терминов указанными классификаторами (стандартные метрики точности, полноты, F-меры) и методом статистического подхода. Показано, что подход на базе модели BERT как бинарного классификатора термин/нетермин достигает 73 % F1-меры и по качеству распознавания терминов превосходит модель последовательной разметки, а также статистический метод.

### Статистический подход к извлечению терминов

Традиционные статистические методы АТЕ [8–10] хорошо изучены и основаны на предположении, что термины часто встречаются в текстах в определенных грамматических формах, для распознавания которых используются статистические и лингвистические свойства (признаки) терминов.

К лингвистическим признакам в первую очередь относятся грамматические образцы многословных терминов, например, прилагательное + существительное + существительное (*спектральный коэффициент излучения*) и др. Лингвистические свойства могут учитывать употребление терминов в определенных контекстах (например, «...будем называть сюръекцией такое отображение...»), а также списки стоп-слов – слов и словосочетаний, которые не могут быть терминами или входить в них. К ним относятся некоторые слова общей лексики (*другой, схема* и т.п.) и оценочные слова (*плохой* и т.д.).

Статистические признаки (меры) основаны на частотах употребления слов в текстах и оценивают вероятность того, что те или иные слова и словосочетания на самом деле будут терминами. Одним из наиболее эффективных

критериев этой группы является *C-value* [19] – мера терминологичности для многословных терминов, учитывающая частоту словосочетания, число входящих в него слов, частоту вложенных в него словосочетаний и поощряющая словосочетания, не входящие в состав других, более длинных.

Общая схема всех статистических методов извлечения терминов из текста включает следующие этапы:

- распознавание слов и словосочетаний по заданным грамматическим образцам и контекстам; извлечение из текста распознанных единиц в качестве терминов-кандидатов;
- ранжирование этих кандидатов по значению выбранной статистической меры с целью получения истинных терминов в верхней части ранжированного списка;
- фильтрация (удаление) некоторых терминов-кандидатов с учетом заданного списка стоп-слов;
- извлечение первых  $K$  элементов из ранжированного и отфильтрованного списка, которые и считаются терминами.

Оценка качества извлечения терминов в рамках рассматриваемого подхода обычно осуществляется с использованием метрик точности и средней точности. Точность равна доле истинных терминов среди всех извлеченных элементов (*Precision@K* – точность на  $K$  кандидатах), а средняя точность (*Average Precision*, *AP*) оценивает, насколько истинные термины находятся ближе к началу ранжированного списка:

$$Precision@K = \frac{\sum_{i=1}^k rel(i)}{k},$$

$$AP = \frac{\sum_{k=1}^n Precision@K \times rel(k)}{R},$$

где  $rel(i) = 1$ , если термин-кандидат под номером  $i$  действительно является термином, и  $rel(i) = 0$  в противном случае;  $R$  – число истинных терминов среди извлеченных.

Эффективность статистических методов АТЕ зависит от предметной области обрабатываемых текстов, размера текстовой коллекции и параметров метода (в частности, от набора грамматических образцов), на практике это обычно 30–60 % средней точности. В силу своей статистической природы такие методы лучше работают для объемных текстов и обычно применяются для извлечения терминологии из коллекций проблемно-ориентированных тек-

тов, хотя могут быть применимы и для отдельных текстов [20], например, для построения глоссариев и предметных указателей [21]. Для повышения качества извлечения терминов в рамках статистического подхода применяются сложные эвристические и зависящие от области текстов стратегии фильтрации, например, такая стратегия представлена в работе [6], где средняя точность извлечения терминов достигала 70 %. Преимущество подхода в том, что статистические методы не требуют каких-либо размеченных данных.

В рамках рассмотренного подхода созданы программные инструменты, например [22, 23], отличающиеся языком программирования (в частности, Java в [24], Scala в [25, 26]) и позволяющие задавать параметры реализуемого статистического метода – статистические меры, набор грамматических образцов, список стоп-слов, а также вычислять значения выбранных мер на заданных текстовых коллекциях.

### Машинное обучение на основе набора признаков терминов

При этом подходе для задачи АТЕ применяется традиционное машинное обучение (байесовский метод, логистическая регрессия, метод опорных векторов, градиентный бустинг и др.) и за счет обучения на размеченных данных определяется значимость признаков, по которым выносится решение, является ли слово или словосочетание термином [3, 11, 12].

Общая схема извлечения терминов из текста включает три этапа.

1. Формирование набора терминов-кандидатов. Как и в статистическом подходе, оно обычно выполняется с помощью грамматических образцов, но в качестве кандидатов могут выступать и всевозможные  $N$ -граммы слов текста.

2. Вычисление значений признаков этих кандидатов: орфографических (регистр букв, наличие небуквенных символов и др.), статистических (статистические меры), лингвистических и контекстных (наличие определенных слов в самом кандидате или в его контексте и др.).

3. Обучение (а затем и применение) машинного классификатора для распознавания терминов на основе вычисленных признаков.

К примеру, на этапах 1 и 2 в работе [11] учитывались  $N$ -граммы ( $N$  от 1 до 5), за исключением стоп-слов, и такие их признаки, как частотность и *C-value*, а в работе [3] применялись

грамматические образцы и несколько статистических признаков.

Машинное обучение позволяет выявить оптимальную комбинацию признаков терминов и тем самым повысить качество их извлечения. Качество традиционно оценивается как точность (доля истинных терминов среди найденных), полнота (доля истинных терминов среди всех терминов текста) и F1-мера (среднее значение гармонической полноты и точности).

Для обучения классификатора необходим корпус с терминологической разметкой, что долгое время составляло проблему из-за малочисленности и отсутствия общепринятых корпусов с ручной (эталонной) разметкой. В исследовательских работах последних лет из немногих доступных ныне корпусов с ручной разметкой терминов преимущественно используется АСТЕР [27] – недавно созданный мультязычный корпус, включающий тексты на трех языках (английском, французском, голландском) для четырех предметных областей (ветровая энергетика, сердечная недостаточность, коррупция, выездка лошадей). Кроме размеченных текстов, корпус содержит списки терминов для каждого языка и каждой области.

В работе [12] на основе корпуса АСТЕР проведено сравнение статистического метода TermoStat (грамматические образцы и статистические меры) с моделью машинного обучения HAMLET. При обучении HAMLET был применен метод случайного леса и рассмотрены более 130 различных признаков терминов. Для различных комбинаций языков и предметных областей обучены отдельные классификаторы, усредненное значение F1-меры для них оказалось в пределах 46.7–54.9 %, что значительно выше, чем полученные 28 % F1-меры, продемонстрированной моделью TermoStat.

Хотя в ряде работ показано, что машинное обучение на признаках достигает лучших результатов при извлечении терминов для заданной предметной области, ценность подхода ограничена, поскольку качество работы обученных классификаторов обычно падает (и даже существенно) на текстах из другой области, где термины могут иметь другие значимые признаки. Одна из причин связана с тем, что набор признаков слабо отражает семантику текста, аналогичная причина действует и для статистического подхода. Еще одним слабым местом машинного обучения на признаках является необходимость ручного (экспертного) подбора признаков.

### Распознавание терминов на основе языковых моделей

С появлением в практике обработки текстов нейросетевых языковых моделей архитектуры Transformer, таких как BERT [14], в ряде работ по АТЕ [15–18] было предложено использовать для машинного обучения вместо наборов признаков терминов (лингвистических, статистических, контекстных) векторные представления слов из предобученных языковых моделей – контекстуализированные эмбединги, сохраняющие контекст применения слов.

В работах [15, 16] предобученные модели BERT дообучались как бинарные классификаторы предсказывать для заданной пары из предложения и  $N$ -граммы из него (рассматриваемой как кандидат в термины), является ли  $N$ -грамма термином или нет. Дообучение (*fine-tuning*) проводилось на размеченных данных по аналогии с задачей предсказания следующего предложения: если  $N$ -грамма являлась термином в контексте предложения, этот пример пары являлся положительным, в противном случае – отрицательным. При таком подходе обучающие данные должны содержать набор пар вида <предложение текста + термин-кандидат из него> и термины-кандидаты заранее извлекаются из текста (как и в вышеописанных подходах к АТЕ).

В статье [15] проведено сравнение дообученных как бинарные классификаторы моделей BERT (RoBERTa для английского языка и CamemBERT для французского) с классификатором на основе метода XGBoost, обученным на наборе признаков терминов (лингвистических и статистических). Положительные примеры обучающего набора были построены с использованием данных корпуса АСТЕР, тогда как отрицательные пары ( $N$ -граммы, которые не являются терминами) были сгенерированы случайным образом. В экспериментах обученная модель XGBoost показала высокую точность распознавания терминов, но низкую полноту, в итоге дав около 27 % F1-меры, в то время как модель классификации на основе BERT существенно превзошла этот результат, показав 48 % F1-меры.

Отметим, что, хотя описанный подход к бинарной классификации термин/нетермин на базе моделей BERT не требует ручного подбора признаков, у него есть существенный недостаток: для получения обучающего набора данных необходима генерация пар с участием всех возможных  $N$ -грамм, что вычислительно затратно.

### Машинное обучение для распознавания терминов на основе последовательной разметки

Другой способ распознавания терминов на базе нейросетевых языковых моделей, представленный в работах [13, 17], не требует предварительного извлечения кандидатов в термины, вместо этого термины распознаются прямо в текстах моделью машинного обучения. Модель обучается на тексте с размеченными терминами и затем выполняет последовательную разметку слов-токенов текста, действуя аналогично таким известным задачам разметки последовательностей, как определение части речи слов и распознавание именованных существностей. Точнее, обучается модель классификации, которая для каждого слова-токена текста предсказывает, является ли токен частью какого-либо термина или нет. Для этого используется разметка токенов ВЮ или ЮО (В помечает начальное слово термина, Ю – его внутренние слова, О – слова текста, не являющиеся частью никакого термина). Обученный классификатор проставляет эти метки словам текста, после чего предсказанные метки могут быть использованы для извлечения терминов из размеченного текста и формирования из них списка.

В работе [13] для обучения классификаторов применялись данные из мультязычного корпуса ACTER, сравнивались несколько методов машинного обучения: часто применяемый для последовательной разметки метод CRF (*Conditional Random Field*), выполняющий обучение на признаках, рекуррентные нейронные сети (RNN) с векторными представлениями слов (эмбедингами) и дообучение модели BERT как классификатора токенов. При этом рассматривались мультязычные и одноязычные эмбединги моделей BERT для представленных в корпусе языков. Эксперименты показали, что модель RNN с одноязычными эмбедингами достигает 47–57 % F1-меры для распознавания терминов (в зависимости от предметной области текстов), превосходя такие оценки CRF-модели и дообученных моделей BERT. Мультязычные эмбединги могут даже улучшить F1-меру до 75 %, если дополнительно к обучающим данным для целевой предметной области берутся данные на другом языке, но для той же области (однако на практике наборы данных с терминологической разметкой на нескольких языках встречаются крайне редко).

Исследование подхода к АТЕ на основе последовательной разметки, но для словенского языка описано в работе [17], в ней применялись несколько моделей семейства BERT и недавно созданный размеченный корпус RSDO5 с терминами четырех предметных областей (биомеханика, химия, ветеринария, лингвистика). Были реализованы 12 моделей распознавания терминов с обучением на текстах одной области и тестированием на другой, результаты показали высокое значение F1-меры – 64–71 %, что доказывает возможность переноса обученных моделей с одной области на другую.

В статье [16] проведено сравнение двух подходов на базе нейросетевых языковых моделей: последовательной разметки токенов текста и бинарной классификации для предсказания термин/нетермин по парам <предложение + термин-кандидат>. На данных корпуса ACTER были проведены эксперименты с кросс-языковым обучением мультязычной модели XML-RoBERTa (семейства BERT), то есть с обучением на одном языке и тестированием на другом, при этом рассматривались разные варианты смены области текста для обучения и тестирования. Для разных пар языков и пар областей обученный бинарный классификатор показал невысокие результаты: 40–58 % F1-меры, в то время как классификатор для последовательной разметки продемонстрировал 44–69 %.

Похожее исследование представлено в [18], где также описаны эксперименты в условиях смены предметной области текста для обучения и тестирования для текстов и предметных областей корпуса ACTER. Однако обученные классификаторы BERT как для последовательной разметки, так и для бинарной классификации термин/нетермин показали довольно низкие результаты: в пределах 34–43 % F1-меры в зависимости от конкретной пары областей и рассматриваемого языка текста.

Таким образом, в рассмотренных работах в области АТЕ на основе нейросетевых языковых моделей оценки качества распознавания терминов обученными моделями отличаются, варьируясь в зависимости от языка текстов и применяемых для обучения данных, что требует дальнейшего изучения. Тем не менее качество распознавания превосходит таковое для статистических методов, а ряд обученных моделей показал довольно высокое значение F1-меры распознавания терминов. Для русского языка подобные модели бинарной клас-

сификации и последовательной разметки текстов практически не исследованы из-за отсутствия подходящих размеченных наборов данных, и настоящая работа восполняет этот пробел, включая вопрос о переносе моделей, обученных для одной предметной области, на другую область.

### Данные для обучения нейросетевых моделей извлечения терминов для русского языка

Для разработки и экспериментальной оценки методов распознавания терминов на основе нейросетевых языковых моделей применительно к текстам на русском языке был построен набор размеченных данных.

В качестве исходных специализированных текстов были выбраны научные тексты по математике и программированию, поскольку задача АТЕ особенно актуальна для обработки научно-технических текстов, содержащих много специальных терминов. Коллекция текстов включала свободно доступные тексты семи учебных пособий на русском языке общим объемом 267 тысяч слов-токенов. Важно, что для этих текстов по результатам предшествующих исследований [21] были известны списки входящих в них (извлеченных) терминов. Тексты охватывали такие предметные области, как *математический анализ* (МатАн), *дифференциальные уравнения* (ДифУр), *дискретная математика* (ДисМат), *искусственный интеллект* (ИИ), *формальные грамматики* (ФормГр), *системы программирования* (СисПрог) и *языки программирования* (ЯзПрог). Эти области, хотя имеют некоторое число общих терминов (например, *функция*, *множество* и др.), все же существенно различаются по терминологии. Статистика по объему текстов (число текенов-

слов) и числу уникальных (разных) терминов в них представлена в таблице 1 (вторая и третья строки).

Для обучения бинарного классификатора пар вида <предложение + словосочетание из него>, который определяет, является ли термин данное словосочетание, был разработан (на языке Python, с помощью библиотеки SpaCy) соответствующий набор с более чем 23 тысячами положительных и отрицательных примеров пар. Предложения брались из текстов коллекции, и положительные примеры включали термин из них, а отрицательные – словосочетания из этих предложений, которые не были терминами. В третьей строке таблицы 1 представлено количество примеров-пар по областям. Приведем положительный пример: <для бесконечных множеств говорить о количестве элементов не имеет смысла, но говорить о мощности множества можно + мощности множества> и отрицательный: <задача о нахождении кратчайшего расстояния может быть решена прямым перебором всевозможных расстояний + всевозможных расстояний>.

При построении набора примеров все тексты были сегментированы на предложения. Для формирования положительных примеров найдены все вхождения каждого термина в предложения текстов коллекции, и предложения связаны с терминами, входящими в них. Для составления отрицательных примеров в текстах найдены все  $N$ -граммы длиной менее пяти и состоящие только из существительных и прилагательных, из которых отброшены все термины, а оставшиеся  $N$ -граммы (например, *данное решение*) связаны с исходными предложениями. Количество отрицательных примеров было уравнено с числом положительных для баланса положительного и отрицательного классов.

Таблица 1

Статистика по областям текстовой коллекции

Table 1

Statistics on text collection domains

Область	МатАн	ДифУр	ДисМат	ИИ	ФормГр	СисПрог	ЯзПрог	В целом
# слова	76 093	19 156	31 085	31 452	17 720	52 515	39 015	267 036
# уникальные термины	360	44	163	95	69	294	106	1 131
# обучающие пары	6 056	1 148	2 948	1 868	1 256	5 620	4 738	23 622
# вхождения терминов	657	854	1 181	250	329	2 516	2 040	7 827

Вторая часть рассматриваемого набора данных была создана для обучения модели последовательной разметки терминов в текстах. В текстах исходной коллекции семи предметных областей найдены вхождения терминов из известных для каждой области списков, и каждое найденное вхождение термина размечено на основе ВЮ-разметки (В – начальное слово-токен термина, I – внутренние слова, O – слова, не являющиеся частью термина), например: *...пересечением[V-term] множеств[I-term] называется[O] множество[V-term] которое[O]...*

Всего было размечено 7 827 вхождений (терминопотреблений), их количество по областям представлено в последней строке таблицы 1. Нейросетевая модель обучалась предсказывать одну из трех меток (В, I, O), согласно которым термины могут быть затем извлечены в список.

### Программные модели распознавания терминов русского языка: обучение

Для экспериментов по распознаванию русскоязычных терминов на базе нейросетевых языковых моделей были выбраны две предобученные модели BERT, наиболее часто применяемые в настоящий момент для автоматического анализа русскоязычных текстов:

- ruBert-base-cased от проекта DeepPavlov [28] (далее ruBert-DeepPavlov) – многоязычный BERT [14], доработанный на русской википедии и новостных корпусах;

- ruBert-base от SberDevices (далее ruBert-Sber) [29] – BERT, исходно обученный для русского языка на текстах из русской википедии, новостей, книг, веб-сайтов и субтитров к фильмам.

Для дообучения этих моделей в обоих исследуемых подходах к распознаванию терминов

(бинарная классификация, последовательная разметка) рассматривались три варианта разбиения созданного набора данных на подмножества для обучения, валидации и тестирования. В таблице 2 представлена информация об этих вариантах: составе каждого подмножества и количестве примеров в каждом из них. В первом варианте для обучения берутся данные из математики (условное название варианта – Математика), для валидации используются примеры из областей искусственного интеллекта и формальных грамматик, а для тестирования – данные из программирования. Вторым вариантом (Программирование) строился как противоположный первому (обучение на программировании, тестирование на математике). В третьем варианте области математики и программирования перемешаны. Рассмотрение таких вариантов позволило исследовать зависимость качества распознавания терминов от выбора предметной области для обучения.

При разбиении на подмножества для обучения, валидации и тестирования, кроме близости объединяемых предметных областей, были также учтены стандартные в машинном обучении пропорции данных в этих подмножествах. Отметим, что во всех вариантах разбиения доля общих терминов для обучающего и тестового подмножеств сравнительно мала, менее 4–7 %. Общие термины не исключались, поскольку это неизбежное явление для близких областей, которое следовало учитывать в экспериментах.

В предварительных экспериментах по обучению моделей была выполнена настройка различных гиперпараметров моделей. Для достижения наилучших значений F1-меры для бинарной классификации оказалось достаточно трех эпох обучения с оптимизатором AdamW и скоростью обучения 5e-6, а для по-

Варианты разбиения набора данных

Таблица 2

Table 2

Dataset splitting options

Разбиение набора данных для обучения	Обучение		Валидация		Тестирование	
	Области	#	Области	#	Области	#
1 Математика	МатАн, ДифУр, ДисМат	10 152	ИИ ФормГр	3 134	СисПрог ЯзПрог	10 346
2 Программирование	СисПрог, ЯзПрог, ИИ, ФормГрам	13 470	ДифУр ДисМат	4 096	МатАн	6 056
3 Смешение областей	ДифУр, ДисМат, ИИ, ФормГр, ЯзПрог	11 946	СисПрог	5 620	МатАн	6 056

следовательной разметки понадобилось 10 эпох со скоростью обучения  $5e-5$  (задача последовательной разметки оказалась сложнее для обучения).

В итоге было обучено по шесть BERT-моделей для каждого из двух рассмотренных подходов:

- бинарный классификатор термин/нетермин, входными данными для которого служит пара <предложение + словосочетание из него>;
- классификатор токенов при последовательной разметке текста, входом которого является неразмеченный текст, в котором размечаются токены-слова, входящие в термины.

### Программные модели распознавания терминов русского языка: результаты

Для всех обученных моделей качество классификации (то есть распознавания терминов) оценивалось по показателям точности, полноты и F1-меры, результаты представлены в таблице 3. Все модели демонстрируют приемлемые результаты по F1-мере от 49.1 до 73.3 %, но бинарные классификаторы для всех вариантов обучения (разбиения) и для всех моделей имеют ощутимо лучшие результаты – от 66.9 до 73.3 % F1-меры.

Сравнение по предобученным моделям показывает, что модели *ruBert/Sber* в среднем показывают лучшие результаты по F1-мере, но разрыв с другой моделью не очень большой: 53.5 % против 51.6 % для последовательной разметки и 72.0 % против 68.9 % для бинарной классификации.

Точность распознавания терминов (*P*) у всех моделей оказалась довольно высокой: до 84.5 % для бинарного классификатора и 79.9 % для последовательной разметки, но полнота (*R*) устойчиво ниже точности, хотя для бинарной классификации приемлема: в интервале от 55.6 до 70.5 %.

Сравнение двух подходов показало, что модели последовательной разметки немного проигрывают моделям бинарной классификации в точности (хотя на варианте Программирование модель *ruBert-Sber* оказалась лучше), но существенно проигрывает в полноте, причем на всех разбиениях (в среднем 40.7 % против 63.5 %).

Следует отметить, что в каждом из подходов качество моделей для разных разбиений набора данных по областям отличается незначительно, означая тем самым, что смена области от обучения к тестированию проходит без существенного падения качества – это может быть применено для распознавания терминов в новых (но близких) областях, для которых нет еще размеченных данных для обучения.

Кроме сопоставления эффективности подходов и качества моделей при разных разбиениях набора данных, было проведено сравнение с результатами традиционного статистического подхода на тех же исходных текстах по математике и программированию, по которым строился обучающий набор данных. Сравнимый статистический метод опирался

- на грамматические образцы именных словосочетаний русского языка и шаблоны типичных контекстов употреблений терминов для выделения терминов-кандидатов;

Таблица 3

### Оценки качества распознавания терминов

Table 3

#### Quality assessments of term recognition

Модель	Разбиение набора данных	Подход					
		Последовательная разметка			Бинарная классификация		
		P, %	R, %	F1, %	P, %	R, %	F1, %
<i>ruBert-DeepPavlov</i>	Математика	76.1	40.4	52.8	83.9	55.6	66.9
	Программирование	78.9	37.4	50.8	80.3	58.1	67.4
	Смешение	63.5	42.8	51.1	75.7	69.6	72.5
В среднем	по разбиениям	72.8	40.2	51.6	80.0	61.1	68.9
<i>ruBert-Sber</i>	Математика	75.9	43.3	55.1	<b>84.5</b>	63.9	72.8
	Программирование	<b>79.9</b>	35.4	49.1	77.6	63.2	70.0
	Смешение	74.0	<b>44.8</b>	<b>55.8</b>	76.3	<b>70.5</b>	<b>73.3</b>
В среднем	по разбиениям	76.6	41.2	53.4	79.5	65.9	<b>72.0</b>
Среднее по моделям		74.7	40.7	52.5	79.7	63.5	<b>70.5</b>

– на фильтрацию терминов-кандидатов (исключались стоп-слова и некоторые словосочетания с ними);

– на применение статистической меры терминологичности C-value для упорядочения терминов-кандидатов.

Результирующие оценки подходов сведены в таблицу 4, которая показывает, что наилучшим является подход с бинарной классификацией терминов, а подход с последовательной разметкой имеет примерно такое же качество извлечения терминов, что и статистический метод.

Таблица 4

#### Сравнение эффективности подходов к распознаванию терминов

Table 4

#### Comparing effectiveness of term recognition approaches

Подход	P, %	R, %	F1, %
Последовательная разметка	74.7	40.7	52.5
Бинарная классификация	79.7	63.5	<b>70.5</b>
Статистический метод	52.0	56.0	52.0

Результаты моделей распознавания терминов для русского языка, созданных на базе предобученных нейросетевых языковых моделей, показали, что качество бинарного BERT-классификатора сравнимо или немного лучше аналогов для других языков (английского, французского, словенского) [15, 16, 18]. Что же касается классификатора для последовательной разметки, то обученные модели, представленные авторами, имеют лучшие результаты, чем их аналоги в работе [18], сравнимы с аналогами из [16], но хуже, чем в [17]. Низкие результаты моделей, возможно, объясняются меньшим размером набора данных для обучения.

Проведенный вручную анализ ошибок распознавания терминов, допущенных наилучшими моделями, показал, что не были распознаны как термины некоторые словосочетания (например, *императивная парадигма*), которые в действительности таковыми являются (это свидетельствует об ошибках в разметке набора данных, которые в дальнейшем могут быть устранены). Более сложными оказались ложноотрицательные (например, *уровень списка*) и ложноположительные случаи (например, *набор функций*), когда словосочетания содержали слова общей лексики (*уровень и набор*). Заметим, однако, что использование таких слов в качестве элементов термина затрудняет распознавание терминов даже людьми.

#### Заключение

В работе охарактеризованы известные подходы к задаче автоматического распознавания и извлечения терминов из текстов. Проведено исследование современного подхода на базе нейросетевой языковой модели BERT применительно к русскому языку и с учетом перехода на близкую предметную область. Оценка качества распознавания терминов разработанными программными моделями показала, что бинарная классификация термин/нетермин при наличии контекстного предложения превосходит метод на основе последовательной разметки терминов в тексте, а также статистический метод. Для обучения моделей и проведения исследования был построен репрезентативный открыто доступный набор размеченных данных, охватывающий термины из семи научных областей математики и программирования и применимый для дальнейшего развития рассмотренной задачи.

#### Список литературы

1. Vivaldi J., Rodrigues H. Evaluation of terms and term extraction systems: A practical approach. *Terminology*, 2007, vol. 13, no. 2, pp. 225–248. doi: 10.1075/term.13.2.06viv.
2. Korkontzelos I., Ananiadou S. Term extraction. In: *The oxford handbook of computational linguistics*, 2014, pp. 991–1012. doi: 10.1093/oxfordhb/9780199573691.013.004.
3. Loukachevitch N., Nokel M. An experimental study of term extraction for real information-retrieval thesauri. *Proc. TIA-2013*, 2013, pp. 69–76.
4. Соловьев С.Ю. Схема и формула глоссария // КИИ-2008: тр. конф. 2008. Т. 2. С. 157–164.
5. Мальковский М.Г., Соловьев С.Ю. От терминологических сетей к толковым словарям // OSTIS-2015: мат. Междунар. науч.-технич. конф. 2015. С. 281–284.
6. Bolshakova E.I., Ivanov K.M. Automating hierarchical subject index construction for scientific documents. In: *LNAI. Proc. RCAI-2020*, 2020, vol. 12412, pp. 201–214. doi: 10.1007/978-3-030-59535-7\_14.
7. Arcan M., Turchi M., Tonelli S., Buitelaar P. Leveraging bilingual terminology to improve machine translation in a CAT environment. *Natural Language Eng.*, 2017, vol. 23, no. 5, pp. 763–788. doi: 10.1017/S1351324917000195.
8. Paziienza M.T., Pennacchiotti M., Zanzotto F.M. Terminology extraction: An analysis of linguistic and statistical approaches. In: *STUDFUZZ. Proc. Knowledge Mining*, 2005, vol. 185, pp. 255–279. doi: 10.1007/3-540-32394-5\_20.

9. Zhang Z., Iria J., Brewster C., Ciravegna F. A comparative evaluation of term recognition algorithms. Proc. Int. Conf. LREC'08, 2008, pp. 2108–2111.
10. Zhang Z., Gao J., Ciravegna F. SemRe-Rank: Improving automatic term extraction by incorporating semantic relatedness with personalised PageRank. ACM TKDD, 2018, vol. 12, no. 5, art. 57, pp. 1–41. doi: 10.1145/3201408.
11. Yuan Y., Gao J., Zhang Y. Supervised learning for robust term extraction. Proc. IALP, 2017, pp. 302–305. doi: 10.1109/IALP.2017.8300603.
12. Terryn A.R., Drouin P., Hoste V., Lefever E. Analysing the impact of supervised machine learning on automatic term extraction: HAMLET Vs TermoStat. Proc. Int. Conf. RANLP, 2019, pp. 1013–1022. doi: 10.26615/978-954-452-056-4\_117.
13. Terryn A.R., Hoste V., Lefever E. Tagging terms in text: A supervised sequential labeling approach to automatic term extraction. Terminology, 2022, vol. 28, no. 1, pp. 157–189. doi: 10.1075/term.21010.rig.
14. Devlin J., Chang M., Lee K., Toutanova K. BERT: Pre-training of deep bidirectional transformers for language understanding. Proc. Conf. of the North, 2019, pp. 4171–4186. doi: 10.18653/v1/N19-1423.
15. Hazem A., Bouhandi M., Boudin F., Daille B. TermEval 2020: TALN-LS2N system for automatic term extraction. Proc. Int. Workshop on Computational Terminology, 2020, pp. 95–100.
16. Lang C., Wachowiak L., Heinisch B., Gromann D. Transforming term extraction: Transformer-based approaches to multilingual term extraction across domains. Proc. ACL-IJCNLP, 2021, pp. 3607–3620. doi: 10.18653/v1/2021.findings-acl.316.
17. Tran H.T.H., Martinc M., Repar A., Doucet A., Pollak S. A transformer-based sequence-labeling approach to the slovenian cross-domain automatic term extraction. Proc. Slovenian Conf. on Language Technologies and Digital Humanities, 2022, pp. 196–204.
18. Hazem A., Bouhandi M., Boudin F., Daille B. Cross-lingual and cross-domain transfer learning for automatic term extraction from low resource data. Proc. LREC, 2022, pp. 648–662.
19. Frantzi K., Ananiadou S., Mima H. Automatic recognition of multi-word terms: The C-value/NC-value method. IJDL, 2000, vol. 3, no. 2, pp. 115–130. doi: 10.1007/s007999900023.
20. Šajatović A., Buljan M., Snajder J., Basic B.D. Evaluating automatic term extraction methods on individual documents. Proc. Joint Workshop MWE-WN, 2019, pp. 149–154. doi: 10.18653/v1/W19-5118.
21. Большакова Е.И., Иванов К.М. Извлечение терминов для построения предметного указателя учебно-научного текста // Компьютерная лингвистика и интеллектуальные технологии: мат. конф. «Диалог». 2018. № 17. С. 143–152 (на англ.).
22. Cram D., Daille B. TermSuite: Terminology extraction with term variant detection. Proc. 54th Annual Meeting ACL, 2016, pp. 13–18. doi: 10.18653/v1/P16-4003.
23. Marciniak M., Mykowiecka A., Rychlik P. TermoPL – a flexible tool for terminology extraction. Proc. 10th Int. Conf. LREC'16, 2016, pp. 2278–2284.
24. Zhang Z., Gao J., Ciravegna F. Jate 2.0: Java automatic term extraction with apache solr. Proc. 10th Int. Conf. LREC'16, 2016, pp. 2262–2269.
25. Astrakhantsev N. ATR4S: toolkit with state-of-the-art automatic terms recognition methods in Scala. LRE, 2018, vol. 52, pp. 853–872. doi: 10.1007/s10579-017-9409-4.
26. Машонский И.Д., Большакова Е.И. Инструментальные средства извлечения терминов из текстов: разработка компонентов для русского языка // Программные системы и инструменты: Тематический сборник. 2020. № 20. С. 94–105.
27. Terryn A.R., Hoste V., Drouin P., Lefever E. TermEval 2020: Shared task on automatic term extraction using the annotated corpora for term extraction research (ACTER) dataset. Proc. Int. Workshop COMPUTERM, 2020, pp. 85–94.
28. Куратов Ю., Архипов М. Адаптация глубоких двунаправленных многоязычных моделей на основе архитектуры transformer для русского языка // Компьютерная лингвистика и интеллектуальные технологии: мат. конф. «Диалог». 2019. № 18. С. 310–316 (на англ.).
29. Zmitrovich D., Abramov A., Kalmykov A., Kadulin V. et al. A family of pretrained transformer language models for Russian. Proc. LREC-COLING, 2024, pp. 507–524.

## Methods and means of term extraction from texts for terminological tasks

Elena I. Bolshakova , Vladislav V. Semak <sup>1</sup>

<sup>1</sup> Lomonosov Moscow State University, Moscow, 119991, Russian Federation

### For citation

Bolshakova, E.I., Semak, V.V. (2025) 'Methods and means of term extraction from texts for terminological tasks', *Software & Systems*, 38(1), pp. 5–16 (in Russ.). doi: 10.15827/0236-235X.149.005-016

### Article info

Received: 18.08.2024

After revision: 14.09.2024

Accepted: 29.09.2024

**Abstract.** The paper describes the current state in the field of automatic term extraction from specialized natural language texts, including scientific and technical documents. Practical applications of methods and tools for extracting terms from texts include creation of terminological dictionaries, thesauri, and glossaries of problem oriented domains, as well as extraction of keywords and construction of subject indexes for specialized documents. The paper overviews approaches to automatic recognition and extraction of terminological words and phrases, which cover traditional statistical methods and methods based on machine learning by using term features or with modern neural network transformer-based language models. A comparison of the approaches is presented, including quality assessments for term recognition and term extraction. The most well-known software tools for automating term extraction within the statistical approach and learning by features are indicated. Authors' studies on term recognition based on neural network language models are described, being applied to Russian scientific texts on mathematics and programming. The data set with terminological annotations created for training term recognition models is briefly characterized, the dataset covers the data from seven related domains. The term recognition models were developed on the basis of pre-trained neural network model BERT, with its additional training (fine-tuning) in two ways: as binary classifier of candidate terms (previously extracted from texts) and as classifier for sequential labeling words in texts. For the developed models, the quality of term recognition is experimentally evaluated, and a comparison with the statistical approach was carried out. The best quality is demonstrated by binary classification models, significantly surpassing the other considered approaches. The experiments also show the applicability of the trained models to texts in close scientific domains.

**Keywords:** automatic processing of specialized texts, terminological words and phrases, automatic term extraction, machine learning for term recognition, BERT-based term extraction

### References

1. Vivaldi, J., Rodrigues, H. (2007) 'Evaluation of terms and term extraction systems: A practical approach', *Terminology*, 13(2), pp. 225–248. doi: 10.1075/term.13.2.06viv.
2. Korkontzelos, I., Ananiadou, S. (2014) 'Term extraction', in *The Oxford Handbook of Computational Linguistics*, pp. 991–1012. doi: 10.1093/oxfordhb/9780199573691.013.004.
3. Loukachevitch, N., Nokel, M. (2013), 'An experimental study of term extraction for real information-retrieval thesauri', *Proc. TIA-2013*, pp. 69–76.
4. Solovev, S.Yu. (2008) 'Glossary diagram and formula', *Proc. CAI-2008*, 2, pp. 157–164 (in Russ.).
5. Malkovsky, M.G., Soloviev, S.Yu. (2015) 'From terminological networks to the explanatory dictionaries', *Proc. Int. Sci. Tech. Conf. OSTIS-2015*, pp. 281–284 (in Russ.).
6. Bolshakova, E.I., Ivanov, K.M. (2020) 'Automating hierarchical subject index construction for scientific documents', in *LNAI. Proc. RCI-2020*, 12412, pp. 201–214. doi: 10.1007/978-3-030-59535-7\_14.
7. Arcan, M., Turchi, M., Tonelli, S., Buitelaar, P. (2017) 'Leveraging bilingual terminology to improve machine translation in a CAT environment', *Natural Language Eng.*, 23(5), pp. 763–788. doi: 10.1017/S1351324917000195.
8. Paziienza, M.T., Pennacchiotti, M., Zanzotto, F.M. (2005) 'Terminology extraction: An analysis of linguistic and statistical approaches', In: *STUDFUZZ. Proc. Knowledge Mining*, 185, pp. 255–279. doi: 10.1007/3-540-32394-5\_20.
9. Zhang, Z., Ira, J., Brewster, C., Ciravegna, F. (2008) 'A comparative evaluation of term recognition algorithms', *Proc. Int. Conf. LREC'08*, pp. 2108–2111.
10. Zhang, Z., Gao, J., Ciravegna, F. (2018) 'SemRe-Rank: Improving automatic term extraction by incorporating semantic relatedness with personalised PageRank', *ACM TKDD*, 12(5), art. 57, pp. 1–41. doi: 10.1145/3201408.
11. Yuan, Y., Gao, J., Zhang, Y. (2017) 'Supervised learning for robust term extraction', *Proc. IALP*, pp. 302–305. doi: 10.1109/IALP.2017.8300603.
12. Terryn, A.R., Drouin, P., Hoste, V., Lefever, E. (2019) 'Analysing the impact of supervised machine learning on automatic term extraction: HAMLET Vs TermoStat', *Proc. Int. Conf. RANLP*, pp. 1013–1022. doi: 10.26615/978-954-452-056-4\_117.
13. Terryn, A.R., Hoste, V., Lefever, E. (2022) 'Tagging terms in text: A supervised sequential labelling approach to automatic term extraction', *Terminology*, 28(1), pp. 157–189. doi: 10.1075/term.21010.rig.
14. Devlin, J., Chang, M., Lee, K., Toutanova, K. (2019) 'BERT: Pre-training of deep bidirectional transformers for language understanding', *Proc. Conf. of the North*, pp. 4171–4186. doi: 10.18653/v1/N19-1423.
15. Hazem, A., Bouhandi, M., Boudin, F., Daille, B. (2020) 'TermEval 2020: TALN-LS2N system for automatic term extraction', *Proc. Int. Workshop on Computational Terminology*, pp. 95–100.
16. Lang, C., Wachowiak, L., Heinisch, B., Gromann, D. (2021) 'Transforming term extraction: Transformer-based approaches to multilingual term extraction across domains', *Proc. ACL-IJCNLP*, 2021, pp. 3607–3620. doi: 10.18653/v1/2021.findings-acl.316.
17. Tran, H.T.H., Martinc, M., Repar, A., Doucet, A., Pollak, S. (2022) 'A transformer-based sequence-labeling approach to the slovenian cross-domain automatic term extraction', *Proc. Slovenian Conf. on Language Technologies and Digital Humanities*, pp. 196–204.
18. Hazem, A., Bouhandi, M., Boudin, F., Daille, B. (2022) 'Cross-lingual and cross-domain transfer learning for automatic term extraction from low resource data', *Proc. LREC*, pp. 648–662.
19. Frantzi, K., Ananiadou, S., Mima, H. (2000) 'Automatic recognition of multi-word terms: The C-value/NC-value method', *IJDL*, 3(2), pp. 115–130. doi: 10.1007/s007999900023.
20. Šajatović, A., Buljan, M., Snajder, J., Basic, B.D. (2019) 'Evaluating automatic term extraction methods on individual documents', *Proc. Joint Workshop MWE-WN*, pp. 149–154. doi: 10.18653/v1/W19-5118.

21. Bolshakova, E.I., Ivanov, K.M. (2018) 'Term extraction for constructing subject index of educational scientific text', *Computational Linguistics and Intellectual Technologies: Proc. Conf. "Dialogue"*, pp. 143–152.
22. Cram, D., Daille, B. (2016) 'TermSuite: Terminology extraction with term variant detection', *Proc. 54th Annual Meeting ACL*, pp. 13–18. doi: 10.18653/v1/P16-4003.
23. Marciniak, M., Mykowiecka, A., Rychlik, P. (2016) 'TermoPL – a flexible tool for terminology extraction', *Proc. 10th Int. Conf. LREC'16*, pp. 2278–2284.
24. Zhang, Z., Gao, J., Ciravegna, F. (2016) 'Jate 2.0: Java automatic term extraction with apache solr', *Proc. 10th Int. Conf. LREC'16*, pp. 2262–2269.
25. Astrakhantsev, N. (2018) 'ATR4S: toolkit with state-of-the-art automatic terms recognition methods in Scala', *LRE*, 52, pp. 853–872. doi: 10.1007/s10579-017-9409-4.
26. Mashonsky, I.D., Bolshakova, E.I. (2020) 'Tools for terminology extraction from texts: development of components for russian language', *Software Systems and Tools*, (20), pp. 94–105 (in Russ.).
27. Terryn, A.R., Hoste, V., Drouin, P., Lefever, E. (2020) 'Termeval 2020: Shared task on automatic term extraction using the annotated corpora for term extraction research (ACTER) dataset', *Proc. Int. Workshop COMPUTERM*, pp. 85–94.
28. Kuratov, Yu., Arkhipov, M. (2019) 'Adaptation of deep bidirectional multilingual transformers for Russian language', *Computational Linguistics and Intellectual Technologies: Proc. Conf. "Dialogue"*, (18), pp. 310–316.
29. Zmitrovich, D., Abramov, A., Kalmykov, A., Kadulin, V. et al. (2024) 'A family of pretrained transformer language models for Russian', *Proc. LREC-COLING*, pp. 507–524.

**Авторы**

**Большакова Елена Игоревна**<sup>1</sup>,  
к.ф.-м.н., доцент,  
bolsh@cs.msu.ru

**Семак Владислав Викторович**<sup>1</sup>,  
аспирант, vlad.semakk@gmail.com

**Authors**

**Elena I. Bolshakova**<sup>1</sup>,  
Cand. of Sci. (Physics and Mathematics),  
Associate Professor, bolsh@cs.msu.ru

**Vladislav V. Semak**<sup>1</sup>,  
Postgraduate Student, vlad.semakk@gmail.com

<sup>1</sup> Московский государственный университет  
им. М.В. Ломоносова, г. Москва, 119991, Россия

<sup>1</sup> Lomonosov Moscow State University,  
Moscow, 119991, Russian Federation

## Использование семантического анализа для автоматизированного выявления хайповых технологий

И.В. Логинова<sup>1</sup>✉, А.С. Пиекалнитс<sup>1</sup>, А.В. Соколов<sup>1</sup>

<sup>1</sup> Институт статистических исследований и экономики знаний НИУ «Высшая школа экономики», г. Москва, 101000, Россия

### Ссылка для цитирования

Логинова И.В., Пиекалнитс А.С., Соколов А.В. Использование семантического анализа для автоматизированного выявления хайповых технологий // Программные продукты и системы. 2025. Т. 38. № 1. С. 17–26. doi: 10.15827/0236-235X.149.017-026

### Информация о статье

Группа специальностей ВАК: 1.2.1

Поступила в редакцию: 12.07.2024

После доработки: 23.07.2024

Принята к публикации: 30.07.2024

**Аннотация.** Предметом проведенного исследования являются завышенные общественные ожидания от новых технологий (хайпов). В статье представлены результаты разработки и апробации методики автоматизированного выявления хайпов среди технологических тематик на основе их текстового следа на примере сферы цифровых технологий. Количество новых технологических разработок в мире постоянно растет, однако реальный потенциал их практического применения может сильно различаться. Важно понимание надежных факторов, позволяющих отличать тренды от хайпов. К числу возможных признаков хайпов отраслевые и технологические эксперты предлагают относить, как правило, отсутствие устойчивой бизнес-модели, несформированный или заведомо ограниченный потребительский рынок, большое количество более эффективных альтернатив. Выявление хайпов в технологической повестке было и остается сложной аналитической задачей. Это обусловлено несогласованностью терминологического аппарата, экспертной природой задачи, недостаточно развитыми методологическими подходами, отсутствием конкретных технических инструментов. Описываемая в статье методика включает извлечение методами обработки естественного языка и компьютерной лингвистики из нескольких десятков миллионов текстовых документов различных типов (научные публикации, патенты, рыночная аналитика) терминов, означающих технологии, расчет объективной меры хайповости каждой технологии, а также построение визуальной карты – наглядного представления ландшафта технологий, позволяющего отделить устойчивые тренды от потенциальных хайпов (тематик с высоким значением показателя хайповости). Хайп-карты в комплексе с другими аналитическими результатами могут использовать лица, принимающие решения, в задачах выявления приоритетных направлений развития, анализа текущих и прогнозирования будущих трендов, риск-менеджмента.

**Ключевые слова:** семантический анализ, большие данные, анализ трендов, технологическая аналитика, хайпы, оценка рисков

**Введение.** Потребность в выработке прикладных решений, направленных на автоматизацию задач технологической и стратегической аналитики, обусловлена рядом вызовов. Риски управленческих ошибок при стратегически значимом выборе велики, поэтому аналитика данных как процесс информационного обеспечения принятия стратегических решений должна опираться на полные и достоверные источники сведений, носить объективный и доказательный характер, а обеспечение информацией лица, принимающего решения, должно быть своевременным и релевантным (сфокусированным). Значительное влияние на деятельность технологических компаний, в частности, на формирование их научно-технологических и продуктово-рыночных стратегий, инвестиционные решения, управление ресурсами, оказывают хайпы, что повышает потребность в разработке надежных подходов для их выявления.

Хайп в контексте данной статьи означает повышенный ажиотаж к определенной теме,

продукту, явлению, событию и др. [1]. Нередко в литературе под хайпом понимается избыточный, чрезмерно высокий интерес к чему-либо, за которым так или иначе следует снижение шумихи, а часто – разочарование [2, 3]. Данный феномен особенно ярко выражен в сфере технологий, где инновации проходят через несколько фаз хайпа, характеризующихся первоначальным повышенным вниманием, последующим снижением ажиотажа и окончательной стабилизацией – восстановлением или падением интереса.

Концепция хайп-цикла, или хайп-кривой, разработанная компанией Gartner, иллюстрирует такую динамику, показывая, как технологии проходят через определенные стадии хайпа: триггер инновации, пик завышенных ожиданий, дно разочарований, склон просветления и плато продуктивности (<https://www.gartner.com/en/research/methodologies/gartner-hype-cycle>). В литературе выделяется ряд ограничений хайп-цикла как инструмента оценки техно-

гической зрелости, обусловленных методологическими изъянами [4], однако критики модели Gartner замечают, что сама концепция хайпа остается перспективной для применения в задачах технологического прогнозирования [4, 5]. Кроме того, хайп-цикл как продукт является источником данных для того, чтобы сделать технологическое прогнозирование более контролируемым, повторяемым и проверяемым, что подтверждается экспериментами по обучению модели машинного обучения (модели ML – *Machine Learning*) для решения задачи предсказания потенциально разрушительных технологий [6].

В настоящей статье хайп рассматривается как первая стадия формирования устойчивого во времени тренда, являющего собой накопленный концентрированный общественный запрос на решение какой-то конкретной задачи ([https://lab-w.com/method\\_meetup/trendwatching](https://lab-w.com/method_meetup/trendwatching)). При этом хайп проявляет себя посредством различных сигналов, часть из которых характеризуются изначально высокой популярностью в результате высокого резонанса обсуждений в СМИ.

Возможности автоматизации аналитической задачи по выявлению хайпов часто оцениваются критически из-за необходимости учитывать контекстуальные и психологические факторы данного явления. Частичная автоматизация задачи выявления хайпов подразумевает определение хайпа в терминах фактической высокой упоминаемости явления в СМИ/новостных источниках на фоне умеренной или низкой представленности этого же явления в научно-технической документации. Таким образом, кроме классических методов автоматизации (в первую очередь машинное обучение, предиктивные модели, временные ряды и др.) применительно к выявлению хайпов, необходимо подключать аналитику больших данных из специфичных для этой цели источников текстов (прежде всего социальных сетей, но также новостных сайтов, блогов, поисковых систем и т.п.), sentiment-анализ как инструмент выявления преобладающего общественного мнения. Такой комплексный инструментарий, тем не менее, будет направлен на реализацию аналитического сценария, в рамках которого исследователь работает с некоторым перечнем гипотез относительно хайпов и по результатам объективного анализа больших текстовых данных принимает решение по каждой технологии, к которой она тяготеет, – к хайпу или к обоснованной разработке. Таким образом, автоматический инструментарий, основан-

ный на анализе текстового следа (текстовой статистики) направлений, нацелен на получение объективных оценок, которые могут быть использованы в качестве информационно-аналитической поддержки решения данной задачи. Однако эффективный подход требует сочетания автоматизированных систем с экспертным анализом для получения более точных результатов.

### Обзор существующих подходов

Особое влияние на проблематику выявления и анализа хайпов оказывает стремительное развитие технологий *искусственного интеллекта* (ИИ). Современные генеративные языковые модели могут создавать правдоподобные синтетические тексты, которые способны влиять на общественное мнение, несмотря на свое искусственное происхождение [7]. В некоторых работах отмечается, что чат-боты, использующие передовые ИИ-технологии, могут применяться для распространения дезинформации (например, генерировать фальшивые новостные статьи, выдавая их за написанные авторитетными источниками) с минимальными затратами. Это способствует усилению эффекта хайпа вокруг технологий, оказывая негативное влияние на результаты аналитики [8].

К рискам, связанным с ИИ и использованием хайпов в задачах технологического прогнозирования, относятся появление и распространение фейков и дипфейков, применяющих такие технологии, как, например, генеративно-состязательные сети для создания реалистичного, но сфабрикованного текстового, аудио- и видеоконтента. Исследования показывают, что фейки в обозримом будущем будут оказывать воздействие, подрывающее доверие к информационным источникам, и стимулировать распространение ложных нарративов [9]. При этом современные технологии их распознавания все еще находятся на недостаточном уровне развития и требуют инвестиций в большее число фундаментальных исследований в данной области, хотя некоторые из предлагаемых методологий и достигают относительно высокой точности [10–12]. Таким образом, в условиях бурного развития технологий генеративного ИИ использование хайпов как источников данных для технологического прогнозирования и стратегической аналитики может искажать результаты и приводить к ошибочным выводам из-за наличия в данных отчасти или полностью сфальсифицированных элементов.

Поскольку хайпы зачастую связаны с неоправданно завышенными ожиданиями и подвергаются искажению за счет использования технологий ИИ, актуальна задача выявления потенциальных хайпов при проведении технологической аналитики и исключения таких хайповых тем при принятии стратегических решений. Для ее решения могут применяться как количественные, так и качественные инструменты, наиболее известным из которых является ранее упомянутый хайп-цикл Gartner, позволяющий разграничивать технологии, находящиеся на пике завышенных ожиданий и менее подверженные влиянию хайпа. Однако в силу того, что экспертная методология Gartner имеет свои ограничения, далее рассматриваются перспективные альтернативы, позволяющие более объективно выявлять и исключать хайпы на основе данных. Например, исследователи рассматривают возможности использования комплексной системы метрик, отражающих интерес к новым технологиям в различных источниках, для оценки местоположения технологий на хайп-кривой Gartner [13]. Возможность автоматизированного моделирования хайп-цикла также подтверждается в работе [14], где авторы при помощи методов машинного обучения проводят анализ текстовых данных (научных статей и патентов) для определения динамики местоположения на кривой Gartner такой технологии, как квантовый компьютер.

Наряду с описанными рассматриваются и альтернативные методики, например, структурированный качественный подход, основанный на применении критического реализма, для определения факторов, которые действительно влияют на успешное внедрение технологий. Для выявления и разграничения обоснованных прогнозов и чрезмерно оптимистичных ожиданий предлагается фокусироваться на оценке реальных, измеримых эффектов, которые могут последовать за внедрением определенной технологии: эффектах автоматизации, информационных (улучшение координации и принятия решений), а также трансформационных эффектах (трансформация рынков). Еще один пример использования качественного анализа для выявления и отделения хайпов от научно-обоснованной информации можно найти в методологии *Grounded Theory* (GT). Она применяется для систематического анализа литературы, субъективность которого сводится к минимуму, по блокчейн-технологиям. Обширный корпус литературы (включающий как научные, так и научно-популярные материалы)

кодируется и категоризируется, после чего содержимое полученных категорий анализируется (в том числе на предмет обоснованности) и интерпретируется, позволяя выявлять хайпы в массиве текстовых данных.

Известна и смешанная методология, сочетающая применение модели ARIMA (интегрированной модели авторегрессии – скользящего среднего) и семантических сетей для выявления зарождающегося хайпа в текстовых медиа (блогах и газетах). Успешное применение подобного подхода говорит о возможности идентификации аномальных всплесков интереса в данных, в том числе и в области исследования технологий. Так, в работе [15] исследуются поведение розничных инвесторов на российском фондовом рынке, а также их склонность к иррациональным действиям под влиянием социальных сетей и хайпов вокруг определенных ценных бумаг. Авторы вводят понятие индикатора хайпа, который интегрирует рассчитываемые при помощи методов машинного обучения метрики внимания (частота упоминаний компании) и настроения (позитивная или негативная тональность сообщений в отношении компании) на основе текстовой информации из социальных сетей. По результатам исследования индикатор хайпа продемонстрировал способность выявлять необоснованные ожидания инвесторов и способствовать построению прибыльных инвестиционных стратегий. Таким образом, подобный подход позволил не только идентифицировать компании, вокруг которых формируется необоснованный ажиотаж на фондовом рынке, но и сигнализировать о возможности построения аналогичных индикаторов хайпа для технологической сферы с целью эффективного автоматизированного выявления ажиотажа вокруг технологий.

Таким образом, хайпы, зачастую оказывающиеся продуктом преувеличенных ожиданий (влиять на которые также могут манипуляции общественным сознанием при помощи и технологий ИИ), способны исказить результаты технологической аналитики, создавая ложные представления о возможностях и перспективах развития технологий. Это, в свою очередь, может нести риски для процесса принятия решений. В литературе предлагаются различные инструменты для выявления хайпов в данных с помощью как качественного анализа, так и автоматизированных количественных методологий. Отделение хайпов от обоснованной информации в условиях роста объемов данных является нелинейной в решении задачей и тре-

бует комплексных подходов к выявлению чрезмерно завышенных ожиданий. Сочетание методов анализа больших данных, реализуемых в системе iFORA, и экспертного знания помогает выявлять подозрительные явления по временным всплескам ажиотажа, разграничивать их от устойчивых трендов, а также подтверждать или опровергать соответствующие гипотезы в задачах стратегического планирования.

### **Методика анализа больших текстовых данных для автоматизированного выявления хайповых технологий**

Подход к автоматизированному выявлению хайпов на основе данных реализован в Институте статистических исследований и экономики знаний НИУ ВШЭ с использованием собственной разработки – системы интеллектуального анализа больших данных iFORA. Это уникальная коллекция, включающая более 800 млн документов из различных источников: научные публикации, патенты, документы рыночной аналитики и профессиональных СМИ, гранты и заявки на них, аналитические и стратегические документы, вакансии, образовательные программы и др. Ключевые типы документов представлены на русском, английском и китайском языках. Семантический анализ данных основан на технологиях глубокой обработки естественного языка, интегрирующих передовые методы машинного обучения и инструменты ИИ. Система основана на модульном подходе и позволяет комбинировать специализированные блоки в зависимости от решаемых аналитических задач для поддержки принятия управленческих решений в сфере науки, технологий и инноваций. Одним из таких блоков является хайп-анализ, который включает извлечение из больших массивов текстов тематик, обозначающих в большинстве случаев технологии или технологические направления, а также выявление среди них технологий, подверженных рискам завышенных ожиданий, то есть хайпов.

#### *Извлечение технологических тематик.*

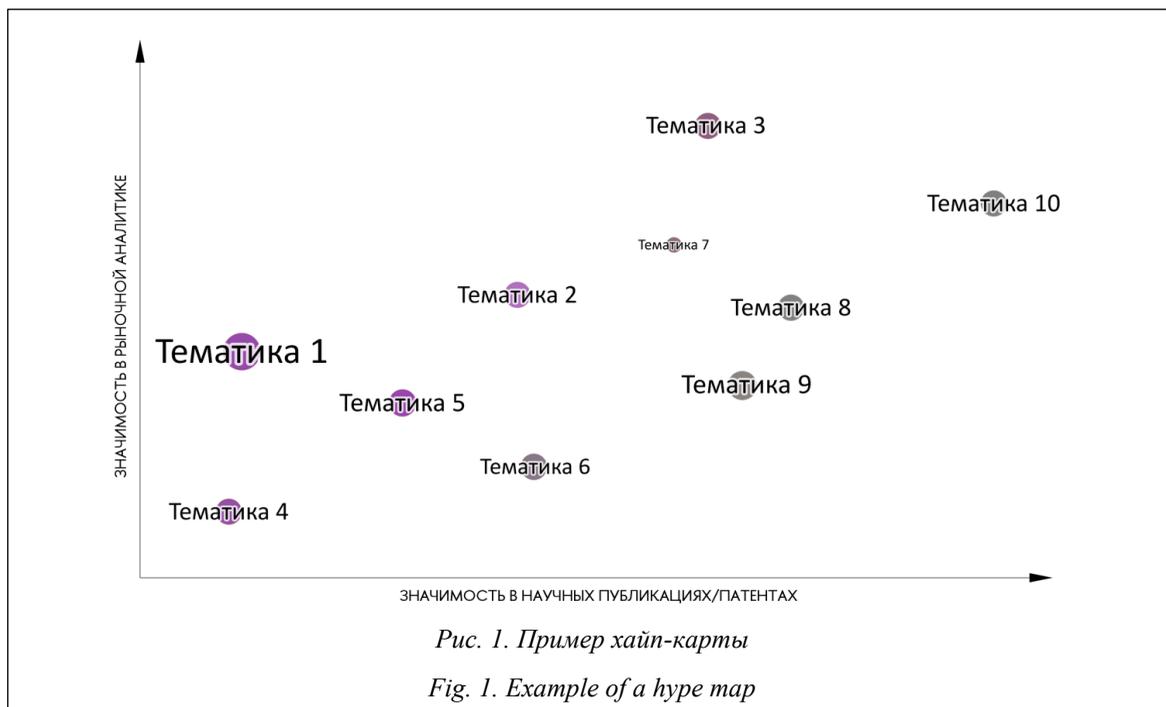
Надежное извлечение из больших массивов неструктурированных текстов терминов, обозначающих технологии и технологические направления, является результатом разработки специальных языковых (эмбединговых) моделей, пригодных для выделения из терминологических полей высокоспециализированных терминов (обозначающих не только технологии, но в общем случае и произвольные заданные

артефакты). Извлечение терминов предполагает работу с предварительно структурированными текстовыми документами, для чего каждый документ разделяется на отдельные предложения, слова и словосочетания с различными лингвистическими характеристиками. Сами документы отбираются из БД iFORA с использованием специально заданного поискового условия (набора понятий, в достаточной мере описывающих анализируемую тематическую область) и алгоритмов «умного» семантического поиска, учитывающего контекст употребления релевантной лексики в документах. Синтаксический анализ связей между словами в предложениях таких документов позволяет выявить зависимости слов (где одно слово является управляющим, а другое зависимым) и тип их связи, основанный на контексте. Далее для отдельных слов и словосочетаний вычисляются универсальные числовые характеристики в целях перехода от лингвистического аппарата к статистическому и наоборот, что позволяет отделить термины, содержащие информацию об объекте изучения, от лингвистического шума. Наиболее часто используемыми и эффективными показателями для определения высокоинформативных терминов являются частота слов, C-value, T-score и т.д. Также о высоком качестве результатов свидетельствует такой показатель, как поточечная взаимная информация. Это относительно простая логарифмическая линейная функция, типичная для использования в нейронных сетях с целью создания высококачественных векторных представлений терминов на больших корпусах данных.

#### *Выявление хайпов.*

К извлеченным из текстов технологическим тематикам далее применяется хайп-анализ, который позволяет разделять исследуемые тематики на две категории: обоснованные тренды/технологии и хайпы, не имеющие под собой значимой научно-технической базы исследований и разработок. Другими словами, хайп-анализ служит инструментом для разграничения устойчивых трендов и кратковременных всплесков необоснованного общественного интереса к определенным темам, которые становятся излишне популярными в медиапространстве. Результаты хайп-анализа визуализируются в виде хайп-карты (рис. 1).

Хайп-карта представляет собой диаграмму рассеяния тематик, расположенных в пространстве двух осей по рассчитанным для них значениям семантических показателей значимости в двух источниках – научных публика-



циях/патентах и документах рыночной аналитики и профессиональных СМИ. Значимость ( $freq_i$ ) отражает интенсивность упоминания тематики в документах источника за выбранный период с учетом размера анализируемого корпуса документов:

$$freq_i = \frac{n_i}{N_i}, \tag{1}$$

где  $n_i$  – встречаемость термина в  $i$ -й год;  $N_i$  – размер корпуса документов в  $i$ -й год.

Отдельные тематики отражаются на хайп-карте круговыми значками. Высокий показатель значимости тематики указывает на ее сильную представленность в массиве документов. По оси абсцисс на хайп-карте представляются относительные показатели значимости тематик в научной или технологической повестке (рассчитываются на основе анализа научных публикаций или патентов соответственно, в отдельных случаях – по совокупному значению научных публикаций и патентов), по оси ординат – в медиаповестке (на основе анализа документов рыночной аналитики и профессиональных СМИ). За счет такого сравнения одни тематики могут быть определены как устойчивые тренды/технологии, а другие – как потенциальные хайпы.

При интерпретации данной визуализации к хайпам могут быть отнесены тематики с относительно высокой значимостью в медиапространстве и низкой в научных публикациях/патентах.

Такие тематики характеризуются высоким показателем хайповости ( $H$ ), рассчитываемым как отношение значимости в двух анализируемых источниках (как правило, рассматривается отношение  $freq_m$  – значимости тематики в документах рыночной аналитики и профессиональных СМИ к  $freq_s$  – значимости в научных публикациях):

$$H = \frac{freq_m}{freq_s}. \tag{2}$$

Чем выше данный показатель, тем с большей вероятностью ту или иную тематику можно отнести к категории хайпов. В пространстве двух осей такие тематики обычно расположены в левой верхней части хайп-карты. Размер и цвет фигурных значков могут быть одинаковыми, а могут задаваться в системе экспертно и отражать различные семантические показатели для углубленной аналитики. Таким образом, хайп-карты наглядно визуализируют отличия в уровне значимости тематик в разных источниках данных. Показатель хайповости и расположение тематик на визуализации предоставляют исследователям информацию об их относительно большей популярности в медиаповестке по сравнению с научной/технологической. Такое сравнение по типам источников позволяет выявлять гипотезы о тематиках-хайпах, вызывающих необоснованно высокий интерес. При этом целесообразно проведение дополнительного анализа хайпов для формирования более надежных вы-



**Обозначение направлений исследований и разработок в сфере цифровых технологий, представленных на хайп-карте**

**Noting research and development areas in the field of digital technologies on the hype map**

Лейбл кругового значка на хайп-карте – Соответствующее направление исследований и разработок	Показатель хайповости	Лейбл кругового значка на хайп-карте – Соответствующее направление исследований и разработок	Показатель хайповости
30 – Приложения метавселенных	888,615	43 – Биометрические технологии	3,704
42 – Цифровые аватары	367,054	22 – Облачные технологии	3,340
33 – Бескодовая разработка программного обеспечения (low-code / no-code)	283,994	14 – Сенсорные системы	2,995
32 – Цифровые финансовые активы	94,551	11 – Интернет вещей	2,968
06 – Речевая аналитика	70,846	15 – Промышленная робототехника	2,965
23 – Децентрализованные финансовые сервисы (DeFi)	28,375	12 – Умный город	2,841
01 – Интеллектуальный анализ данных	25,276	04 – Системы автоматического принятия решений	2,638
21 – Кибербезопасность	22,544	49 – Аддитивные технологии	2,110
18 – Беспилотная агротехника	18,718	35 – Технологии представления и доставки образовательного материала	1,982
16 – Беспилотный наземный транспорт (автомобили и общественный транспорт)	17,604	26 – Системы хранения разнородных больших данных	1,919
40 – Спутниковые системы связи	16,577	36 – Квантовые вычисления	1,839
44 – Интеллектуальные голосовые помощники	11,925	41 – Системы позиционирования и навигации	1,188
19 – Высокоскоростные технологии беспроводной передачи данных на малых дистанциях (WiFi, Bluetooth и др.)	11,382	05 – Анализ и обработка естественного языка	1,012
24 – Технологии распределенного реестра	9,120	08 – Автоматизированные системы медицинской диагностики	0,998
34 – Технологии управления жизненным циклом продукта	9,026	25 – Распределенные вычисления	0,884
17 – Технологии беспроводной передачи данных (5G, 6G и др.)	8,719	29 – Преобразование и кодирование данных	0,781
13 – Коллаборативные технологии	7,249	37 – Квантовые коммуникации	0,752
45 – Автоматическая идентификация и сбор данных	6,491	03 – Компьютерное зрение	0,734
21 – Цифровые двойники физических процессов	5,317	47 – Технологии управления образованием	0,589
10 – Когнитивные вычисления	5,110	07 – Системы идентификации и анализа звукового контента	0,465
28 – Системы аутентификации пользователей	4,396	46 – Технологии человеко-машинного взаимодействия	0,444
09 – Телемедицина	4,314	38 – Фотоника	0,354
50 – Адаптивное производство	4,204	02 – Обучение нейросетей	0,326
48 – Цифровые фабрики	3,914	39 – Геоинформационные системы	0,255
31 – Технологии виртуальной и дополненной реальности	3,858	20 – Имитационное моделирование	0,115

доступные для массового использования. В условиях высокой хайповости технологий компании могут делать значительные инвестиции в их развитие, опираясь на прогнозы о быстром возврате вложений. Тем не менее отсутствие реальных прикладных решений и их практиче-

ской ценности может привести к неэффективности таких инвестиций и финансовым потерям. Кроме того, дефицит знаний у пользователей о возможностях и функциональности новых технологий является значительным препятствием для их распространения. Так, например,

технологии бескодовой разработки могут позволить специалистам без профессиональных технических навыков создавать IT-продукты, но недостаток осведомленности о возможностях таких инструментов может препятствовать их активному внедрению в бизнес-процессы. В целях повышения конкурентоспособности компании могут интегрировать данные технологии в свои стратегии развития. Однако, если технологии окажутся временным хайпом, а не устойчивым трендом, возникнет необходимость пересмотра стратегий, что приведет к значительным финансовым потерям. Также следует отметить отсутствие четких регуляторных рамок в отношении хайповых технологий, что создает дополнительные барьеры для формирования полноценного рынка. Например, децентрализованные финансовые сервисы, открывающие доступ к широкому спектру финансовых инструментов, ограничены в использовании из-за неопределенности в отношении правовых аспектов их внедрения. Наряду с децентрализованными финансами цифровые финансовые активы также характеризуются невысокой практической применимостью, что может быть обусловлено недостаточной интеграцией в существующие финансовые системы и неопределенностью правовых аспектов их использования. Таким образом, ограниченность прикладных решений, дефицит знаний у пользователей и отсутствие четких регуляторных рамок могут способствовать высокой хайповости технологий. Данные факторы указывают на отсутствие или низкую зрелость массовых рынков, а активное медиаосвещение технологий может способствовать инвестициям в технологии, которые не будут реализованы в виде коммерческих продуктов.

### Заключение

В настоящей работе рассмотрены возможности автоматизированного выявления хайпов среди трендов и технологий на основе анализа больших текстовых данных. Апробация предложенного подхода реализована в рамках семантического картирования направлений исследований и разработок на примере сферы циф-

ровых технологий. С использованием системы iFORA проанализированы англоязычные научные публикации и документы отраслевой рыночной аналитики и профессиональных СМИ за 2017–2022 гг.

Результаты проведенного анализа позволили выделить как крупные системообразующие направления цифровых технологий, широко представленные в зарубежных научных публикациях, так и гипотезы о технологиях-хайпах, характеризующихся всплесками интереса в медиа. Так, к числу наиболее хайповых отнесены направления исследований и разработок, связанные с развитием метавселенных, цифровых аватаров и бескодовой разработки программного обеспечения. Направления, связанные с метавселенными и цифровыми аватарами, несмотря на значительное внимание со стороны общественности, к концу 2024 г. не демонстрируют устойчивость и масштабируемость, необходимые для трансформации в долгосрочные тренды. В то же время определение технологий бескодовой разработки программного обеспечения как хайпа в 2022 г. сегодня может быть интерпретировано как отражение обоснованно растущего общественного интереса, связанного с закономерным этапом развития технологии. Активная медийная поддержка данного направления подчеркивала перспективность и способствовала его становлению в качестве устойчивого тренда.

Таким образом, анализ больших текстовых данных можно рассматривать в качестве эффективного инструмента для частично автоматизированного выявления технологических хайпов. Для минимизации рисков, связанных с определением приоритетных направлений развития на основе их текстового следа, рекомендуется проводить углубленный анализ выявляемых хайповых трендов и технологий, учитывая их развитие в динамике. Такой подход позволит более точно различать краткосрочные всплески интереса от устойчивых тенденций, своевременно реагировать на изменения технологических рынков и принимать обоснованные решения относительно инвестиций и стратегического планирования.

### Список литературы

1. Logue D., Grimes M. Living up to the hype: How new ventures manage the resource and liability of future-oriented visions within the nascent market of impact investing. *AMJ*, 2022, vol. 65, no. 3, pp. 1055–1082. doi: 10.5465/amj.2020.1583.
2. Intemann K. Understanding the problem of “hype”: Exaggeration, values, and trust in science. *Canadian J. of Philosophy*, 2022, vol. 52, no. 3, pp. 279–294. doi: 10.1017/can.2020.45.
3. Yap J., McLellan B. A historical analysis of hydrogen economy research, development, and expectations, 1972 to 2020. *Environments*, 2023, vol. 10, no. 1, art. 11. doi: 10.3390/environments10010011.

4. White G.R.T., Samuel A. Programmatic advertising: Forewarning and avoiding hype-cycle failure. TFSC, 2019, vol. 144, pp. 157–168. doi: 10.1016/j.techfore.2019.03.020.
5. Shi Y., Herniman J. The role of expectation in innovation evolution: Exploring hype cycles. *Technovation*, 2023, vol. 119, no. 3, art. 102459. doi: 10.1016/j.technovation.2022.102459.
6. Chen X., Han T. Disruptive technology forecasting based on gartner hype cycle. Proc. TEMSCON, 2019, pp. 1–6. doi: 10.1109/TEMSCON.2019.8813649.
7. Kreps S., McCain R.M., Brundage M. All the news that's fit to fabricate: AI-generated text as a tool of media misinformation. *SSRN Electronic J.*, 2022, vol. 9, no. 1, pp. 104–117. URL: <https://ssrn.com/abstract=3525002> (дата обращения: 23.06.2024). doi: 10.2139/ssrn.3525002.
8. Shoaib M.R., Wang Z., Ahvanooy M.T., Zhao J. Deepfakes, misinformation, and disinformation in the era of frontier AI, generative AI, and large AI models. Proc. ICCA, 2023, pp. 1–7. doi: 10.1109/ICCA59364.2023.10401723.
9. Nnamdi N., Oniyinde O.A., Abegunde B. An appraisal of the implications of deep fakes: The need for urgent international legislations. *AJLG*, 2023, vol. 8, no. 1, pp. 43–70. doi: 10.47672/ajlg.1540.
10. Korshunov P., Marcel S. Vulnerability assessment and detection of deepfake videos. Proc. ICB, 2019, pp. 1–6. doi: 10.1109/ICB45273.2019.8987375.
11. Burnaev E., Mironov E., Shpilman A., Mironenko M., Katalevsky D. Practical AI cases for solving ESG challenges. *Sustainability*, 2023, vol. 15, no. 17, art. 12731. doi: 10.3390/su151712731.
12. Ikram S.T., Priya V., Chambial S., Sood D., Arulkumar V. A performance enhancement of deepfake video detection through the use of a hybrid CNN Deep learning model. *IJECE*, 2023, vol. 14, no. 2, pp. 169–178. doi: 10.32985/ijece.14.2.6.
13. Carbonell J., Sánchez-Esguevillas A., Carro B. Easing the assessment of emerging technologies in technology observatories. Findings about patterns of dissemination of emerging technologies on the internet. *Technology Analysis & Strategic Management*, 2018, vol. 30, no. 1, pp. 113–129. doi: 10.1080/09537325.2017.1337886.
14. Kondo Y., Asatani K., Sakata I. Extending hype cycle prediction by applying graph network analysis. Proc. PICMET, 2022, pp. 1–9. doi: 10.23919/PICMET53225.2022.9882551.
15. Teplova T., Tomtosov A., Sokolova T. A retail investor in a cobweb of social networks. *PLoS ONE*, 2022, vol. 17, no. 12, art. e0276924. doi: 10.1371/journal.pone.0276924.

Software &amp; Systems

doi: 10.15827/0236-235X.149.017-026

2025, 38(1), pp. 17–26

### Automated identification of hype technologies: Semantic analysis

Irina V. Loginova <sup>1</sup>✉, Anna S. Piekalnits <sup>1</sup>, Aleksandr V. Sokolov <sup>1</sup><sup>1</sup> National Research University Higher School of Economics,  
Moscow, 101000, Russian Federation

#### For citation

Loginova, I.V., Piekalnits, A.S., Sokolov, A.V. (2025) 'Automated identification of hype technologies: Semantic analysis', *Software & Systems*, 38(1), pp. 17–26 (in Russ.). doi: 10.15827/0236-235X.149.017-026

#### Article info

Received: 12.07.2024

After revision: 23.07.2024

Accepted: 30.07.2024

**Abstract.** The research focuses on inflated public expectations of new technologies, or hypes. The paper presents the results of the development and testing an automated methodology for the automated identification of hypes among technological topics based on their textual trace in the digital technology field. The amount of new technological developments in the world is constantly growing, however their real potential for practical application varies greatly. Therefore, it is important to understand reliable factors to distinguish trends from hypes. Typically, industry and technology experts suggest that the possible signs of hypes include the following ones: absence of a stable business model, an unformed or obviously limited consumer market, and a large number of more effective alternatives. Identifying hypes in the technology agenda remains a difficult analytical task. This is due to the terminological inconsistency, the expert nature of the task, insufficiently developed methodological approaches, and the lack of specific technical tools. The method described in this paper involves the extraction of terms referring to technologies using natural language processing and computational linguistics techniques. These terms are extracted from several dozens of millions of different types of text documents, such as scientific publications, patents, and market analytics. The method also includes calculating an objective measure of each technology's "hype"

and constructing a visual map that illustrates the technology landscape that allows separating sustainable trends from potential hypes. Decision makers can use such hype maps in conjunction with other analytical results to identify priority development areas, analyze current and forecast future trends, and in risk management.

**Keywords:** semantic analysis, big data, trend analysis, technology analytics, hypes, risk assessment

### References

1. Logue, D., Grimes, M. (2022) 'Living up to the hype: How new ventures manage the resource and liability of future-oriented visions within the nascent market of impact investing', *AMJ*, 65(3), pp. 1055–1082. doi: 10.5465/amj.2020.1583.
2. Intemann, K. (2022) 'Understanding the problem of "hype": Exaggeration, values, and trust in science', *Canadian J. of Philosophy*, 52(3), pp. 279–294. doi: 10.1017/can.2020.45.
3. Yap, J., McLellan, B. (2023) 'A historical analysis of hydrogen economy research, development, and expectations, 1972 to 2020', *Environments*, 10(1), art. 11. doi: 10.3390/environments10010011.
4. White, G.R.T., Samuel, A. (2019) 'Programmatic advertising: Forewarning and avoiding hype-cycle failure', *TFSC*, 144, pp. 157–168. doi: 10.1016/j.techfore.2019.03.020.
5. Shi, Y., Herniman, J. (2023) 'The role of expectation in innovation evolution: Exploring hype cycles', *Technovation*, 119(3), art. 102459. doi: 10.1016/j.technovation.2022.102459.
6. Chen, X., Han, T. (2019) 'Disruptive technology forecasting based on gartner hype cycle', *Proc. TEMSCON*, pp. 1–6. doi: 10.1109/TEMSCON.2019.8813649.
7. Kreps, S., McCain, R.M., Brundage, M. (2022) 'All the news that's fit to fabricate: AI-generated text as a tool of media misinformation', *SSRN Electronic J.*, 9(1), pp. 104–117, available at: <https://ssrn.com/abstract=3525002> (accessed June 23, 2024). doi: 10.2139/ssrn.3525002.
8. Shoaib, M.R., Wang, Z., Ahvanooy, M.T., Zhao, J. (2023) 'Deepfakes, misinformation, and disinformation in the era of frontier AI, generative AI, and large AI models', *Proc. ICCA*, pp. 1–7. doi: 10.1109/ICCA59364.2023.10401723.
9. Nnamdi, N., Oniyinde, O.A., Abegunde, B. (2023) 'An appraisal of the implications of deep fakes: The need for urgent international legislations', *AJLG*, 8(1), pp. 43–70. doi: 10.47672/ajlg.1540.
10. Korshunov, P., Marcel, S. (2019) 'Vulnerability assessment and detection of deepfake videos', *Proc. ICB*, pp. 1–6. doi: 10.1109/ICB45273.2019.8987375.
11. Burnaev, E., Mironov, E., Shpilman, A., Mironenko, M., Katalevsky, D. (2023) 'Practical AI cases for solving ESG challenges', *Sustainability*, 15(17), art. 12731. doi: 10.3390/su151712731.
12. Ikram, S.T., Priya, V., Chambial, S., Sood, D., Arulkumar, V. (2023) 'A performance enhancement of deepfake video detection through the use of a hybrid CNN Deep learning model', *IJECEs*, 14(2), pp. 169–178. doi: 10.32985/ijece.14.2.6.
13. Carbonell, J., Sánchez-Esguevillas, A., Carro, B. (2018) 'Easing the assessment of emerging technologies in technology observatories. Findings about patterns of dissemination of emerging technologies on the internet', *Technology Analysis & Strategic Management*, 30(1), pp. 113–129. doi: 10.1080/09537325.2017.1337886.
14. Kondo, Y., Asatani, K., Sakata, I. (2022) 'Extending hype cycle prediction by applying graph network analysis', *Proc. PICMET*, pp. 1–9. doi: 10.23919/PICMET53225.2022.9882551.
15. Teplova, T., Tomtosov, A., Sokolova, T. (2022) 'A retail investor in a cobweb of social networks', *PLoS ONE*, 17(12), art. e0276924. doi: 10.1371/journal.pone.0276924.

### Авторы

**Логинава Ирина Владимировна**<sup>1</sup>,  
заведующий отделом, [iloginova@hse.ru](mailto:iloginova@hse.ru)

**Пиекалните Анна Сергеевна**<sup>1</sup>,  
ведущий эксперт, [apiekalnits@hse.ru](mailto:apiekalnits@hse.ru)

**Соколов Александр Васильевич**<sup>1</sup>, профессор,  
заместитель директора, [sokolov@hse.ru](mailto:sokolov@hse.ru)

### Authors

**Irina V. Loginova**<sup>1</sup>,  
Head of Department, [iloginova@hse.ru](mailto:iloginova@hse.ru)

**Anna S. Piekalnits**<sup>1</sup>, Leading Expert,  
[apiekalnits@hse.ru](mailto:apiekalnits@hse.ru)

**Aleksandr V. Sokolov**<sup>1</sup>, Professor,  
Deputy Director, [sokolov@hse.ru](mailto:sokolov@hse.ru)

<sup>1</sup> Институт статистических исследований  
и экономики знаний НИУ  
«Высшая школа экономики»,  
г. Москва, 101000, Россия

<sup>1</sup> National Research University  
Higher School of Economics,  
Moscow, 101000,  
Russian Federation

## Генетический алгоритм размещения требований в задаче планирования производственных процессов потокового типа

А.И. Кибзун<sup>1</sup>, В.А. Рассказова<sup>1</sup> ✉

<sup>1</sup> Институт компьютерных наук и прикладной математики,  
Московский авиационный институт (национальный исследовательский университет)  
(МАИ), г. Москва, 125993, Россия

### Ссылка для цитирования

Кибзун А.И., Рассказова В.А. Генетический алгоритм размещения требований в задаче планирования производственных процессов потокового типа // Программные продукты и системы. 2025. Т. 38. № 1. С. 27–38. doi: 10.15827/0236-235X.149.027-038

### Информация о статье

Группа специальностей ВАК: 2.3.5

Поступила в редакцию: 18.07.2024

После доработки: 14.08.2024

Принята к публикации: 23.08.2024

**Аннотация.** В статье рассматривается задача планирования производственных процессов потокового типа. В рамках каскадной схемы комплексное решение охватывает этап назначения подготовительных агрегатов и последующий этап формирования детализированных технологических маршрутов для исполнения заданного множества требований точно в срок и с учетом ограничений на допустимые длительности обработки на каждом переделе. Данная схема реализуется в составе проблемно-ориентированного вычислительного комплекса, однако по ряду естественных причин задача может оказаться несовместной уже на этапе назначения подготовительных агрегатов. Один из путей преодоления обозначенных трудностей – разработка и реализация алгоритмов штрафных функций для поиска максимальных совместных подсистем в противоречивых задачах оптимизации. В настоящей работе для этих целей предлагается идеологически другой подход, основанный на рассмотрении предварительного этапа размещения требований таким образом, чтобы последующие этапы решения комплексной задачи были гарантированно разрешимы. Размещение требований формализуется как задача поиска отображения установленного вида, оптимального по эвристическому критерию потенциальной нагрузки на подготовительные агрегаты в рассматриваемом периоде планирования. Для решения этой задачи авторы статьи разработали генетический алгоритм, что обусловило существенное преимущество по быстродействию в сравнении с фундаментальными подходами математического программирования (например, в сравнении с моделями целочисленного линейного программирования). В целях снижения рисков вымирания популяции на каждой итерации генетического алгоритма применяется правило безусловной миграции представителя с наименьшим значением критерия. Такой подход обеспечивает также эффективные показатели сходимости алгоритма по числу итераций без существенного улучшения целевого функционала. Разработанный генетический алгоритм реализуется как автономный модуль вычислительного комплекса для решения задач планирования процессного производства. Вычислительный эксперимент проводится с использованием данного модуля в разрезе сравнительного анализа качества решения исходной комплексной задачи.

**Ключевые слова:** генетический алгоритм, вычислительный комплекс, производственное планирование, процессное производство, теория расписаний, комбинаторная оптимизация

**Введение.** Производство потокового типа (процессное производство) представляет собой широкую прикладную область. Его отличительными особенностями являются непрерывность отдельных этапов и ритмичность операций в строго установленной последовательности. В настоящее время многие промышленные предприятия переживают стадию цифровой трансформации, когда задачи повышения эффективности производства решаются путем внедрения цифровых сервисов планирования, слежения, мониторинга и предиктивной аналитики. Такая тенденция обусловлена не только значительными оптимизационными возможностями, но и ограничениями инфраструктурных преобразований, поскольку вмешательство в топологию цехов, включая расширение парка оборудования, часто оказывается крайне трудо-

емким или даже невозможным. В связи с этим особую актуальность приобретают задачи разработки и внедрения полнофункциональных проблемно-ориентированных вычислительных комплексов для решения упомянутых прикладных задач.

Традиционно выделяют три уровня производственного планирования: стратегическое (верхний уровень, долгосрочное планирование), формирование комплексного графика производства (средний уровень) и оперативное планирование и управление (нижний уровень). В работе [1] обсуждаются особенности разработки и внедрения систем планирования. В настоящей статье рассматривается задача среднесрочного планирования производственных процессов потокового типа.

К современным инструментам решения задач среднесрочного планирования производственных процессов относят системы класса APS (*Advanced Planning and Scheduling*). Среди популярных отечественных систем данного класса можно выделить такие, как AVM.APS, BFG APS, БИА.APS и Аскон Гольфстрим. Из зарубежных разработок класса APS широкое распространение получила Tecnomatix Plant Simulation. Основными методами решения задач оптимизации в данных системах выступают алгоритмы имитационного моделирования и машинного обучения [2–4]. В настоящей статье для решения рассматриваемой комплексной задачи планирования используется система, в основе которой лежат подходы математического программирования [5]. Рассматриваемая комплексная задача планирования производственных процессов потокового типа включает два этапа. На первом этапе для фиксированного по времени и машинам множества требований (работ) формируется график их обработки на основных подготовительных агрегатах. На следующем этапе для фиксированных по времени, машинам и подготовительным агрегатам требований формируется детализированный технологический маршрут. Такой декомпозированный подход обусловлен практическими аспектами рассматриваемого типа производства и, кроме того, позволяет существенно снизить размерности оптимизационных задач, возникающих на каждом этапе. Однако часто уже на первом этапе решения комплексной задачи (назначение подготовительных агрегатов) возникающие оптимизационные модели оказываются несовместными. Природа такой несовместности во многом обусловлена исходным размещением требований по машинам и фиксированным временем начала и длительности их обработки. В этой связи в настоящей статье предлагается рассмотрение вспомогательного этапа размещения требований в целях обеспечения последующей гарантированной разрешимости этапов назначения подготовительных агрегатов и формирования детализированных технологических маршрутов. Для решения задачи размещения требований предлагается генетический алгоритм, реализуемый в качестве автономного модуля системы планирования [5].

Генетические алгоритмы определяют одно из фундаментальных направлений современных исследований в области случайно направленного поиска [6, 7]. Эффективность их применения для решения различных прикладных

задач подтверждается многочисленными работами отечественных и зарубежных авторов. Так, например, авторы работы [8] применяют генетические алгоритмы для решения задач размещения, возникающих при проектировании интегральных схем. В [9, 10] генетические алгоритмы предложены для решения задач планирования процесса перевозок на железнодорожном транспорте. В задачах планирования в машиностроении и других серийных производствах также эффективно применяются метаэвристические подходы, в том числе генетические алгоритмы [11, 12]. Обзор различных задач производственного планирования класса RCPSP (*Resource Constrained Project Scheduling Problem*) и метаэвристические подходы к решению подробно обсуждаются в работах [13, 14]. Также широко распространены гибридные подходы на базе генетических алгоритмов. Например, в работе [15] рассматривается частный случай задачи производственного планирования потокового типа с применением гибридного алгоритма. В [16, 17] для задач с критерием минимизации простоев оборудования предложены эффективные эволюционные алгоритмы. Подходы нечеткой логики развиваются авторами [18] для решения комплексных задач производственного планирования.

Разработка нового генетического алгоритма в настоящей работе продиктована спецификой рассматриваемой задачи, когда этап размещения требований выделяется в отдельную предварительную стадию решения комплексной задачи. С этой точки зрения данная задача и методы ее решения слабо освещены в существующих публикациях.

### Постановка задачи

Пусть заданы множество требований  $S = \{\sigma_1, \dots, \sigma_n\}$  и множество машин  $R = \{r_1, \dots, r_m\}$ .

Для каждого требования определим  $\rho_i \subseteq R$  как подмножество машин, доступных для размещения  $\sigma_i$ . С прикладной точки зрения подмножество машин  $\rho_i$  определяется технологическими характеристиками требования  $\sigma_i$ . Пусть для каждого  $\sigma_i$  определены также  $\tau_i$ ,  $\hat{\tau}_i$  – минимально необходимое и максимально допустимое время ожидания с момента подготовки на специализированном агрегате из множества  $K = \{k_1, \dots, k_l\}$ . Допустимые минимальную и максимальную длительности обработки  $\sigma_i$  на машине из подмножества  $\rho_i$  обозначим соответственно через  $\mu_i$ ,  $\hat{\mu}_i$  (рис. 1).

Определим период времени  $[T_0, T]$ , доступный для размещения требований  $S$  по машинам  $R$ , где  $T_0$  принимает любое произвольное значение и

$$T = T_0 + \max_{j=1, m} \sum_{i=1}^n \{\hat{\mu}_i | r_j \in \rho_i\}. \quad (1)$$

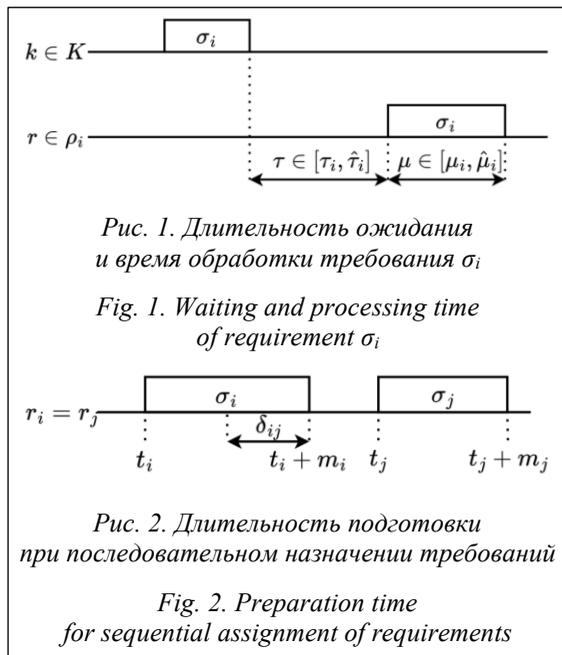
Задача размещения требований  $S$  по машинам  $R$  состоит в построении отображения вида

$$f : S \rightarrow \{(r, t, m) | r \in R, t, m \in \mathbb{Z}^+\}, \quad (2)$$

где  $f(\sigma_i) = (r_i, t_i, m_i)$ , если требование  $\sigma_i$  выполняется на машине  $r_i$  с началом в момент времени  $t_i$  и продолжительностью  $m_i$ , с ограничениями:

- для любого  $f(\sigma_i)$  выполняется  $r_i \in \rho_i$ ;
- для любого  $f(\sigma_i)$  выполняется  $T_0 - \max_{i=1, n} \{\hat{\tau}_i\} \leq t_i \leq T$ ;
- для любого  $f(\sigma_i)$  выполняется  $m_i \in [\mu_i, \hat{\mu}_i]$ ;
- для любых  $f(\sigma_i), f(\sigma_j) | r_i = r_j$  выполняется
 
$$\begin{cases} t_j \geq t_i + m_i, & \text{если } t_i \leq t_j, \\ t_i \geq t_j + m_j & \text{иначе.} \end{cases} \quad (3)$$

Замечание 1. В расширенной постановке задача (2) может включать также и ограничения на длительность  $\delta_{ij}$  подготовки машины  $r_i = r_j$  при последовательном назначении требований  $f(\sigma_i), f(\sigma_j)$  (рис. 2). Авторы не рассматривают данную группу ограничений ввиду декомпозиции множества  $R = \bigcup_{i=1}^n \rho_i$  в предположении, что длительность подготовки постоянна для требо-



ваний с идентичными технологическими характеристиками и включена в интервал  $[t_i, t_i + m_i]$  для любых  $f(\sigma_i), f(\sigma_j)$ , где  $r_i = r_j$  и  $t_i \leq t_j$ .

Замечание 2. В случае  $l \leq m$  задача размещения требований на горизонте  $[T_0, T]$ , где  $T$  определяется по правилу (1), становится тривиальной с точки зрения поиска допустимого отображения вида (2) и может быть решена жадным алгоритмом. В этой связи далее будем полагать, что  $l \leq \lfloor \frac{m}{2} \rfloor$ , где

$$\lfloor \frac{m}{2} \rfloor = \min_{x \in \mathbb{Z}} \{x | x \geq \frac{m}{2}\}.$$

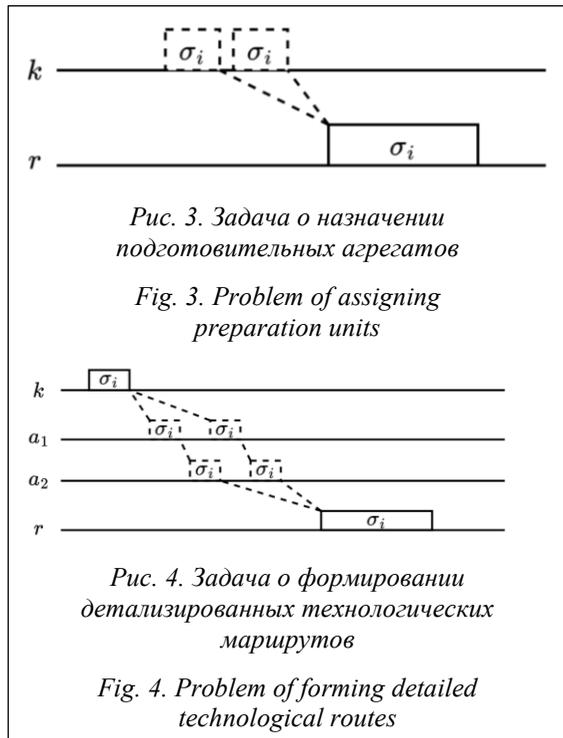
В настоящей работе в качестве критерия оптимизации размещения вида (2) рассматривается потенциальная загрузка агрегатов подготовки требований. Такой подход связан с тем, что практические задачи размещения требований, как правило, рассматриваются отдельно от задач назначения подготовительных агрегатов и последующих машин обработки. Так, в частности, время начала выполнения требования  $\sigma_i$  на машине  $r$  выступает входными данными и фиксировано для задачи о назначении агрегатов подготовки (рис. 3 (пунктирной линией обозначены возможные варианты назначения агрегата  $k$  для подготовки фиксированного требования  $\sigma_i$ )).

В работе [19] для решения задачи о назначении подготовительных агрегатов разрабатывается модель *целочисленного линейного программирования* (ЦЛП).

Далее время завершения подготовки требования  $\sigma_i$  на агрегате  $k$ , а также время начала выполнения на машине  $r$  выступают входными данными и фиксированы для задачи о формировании детализированных технологических маршрутов (рис. 4 (пунктирной линией обозначены варианты назначения машин  $a_1$  и  $a_2$  для обработки фиксированного требования  $\sigma_i$ )).

Для задачи формирования детализированных технологических маршрутов в работе [20] также предлагаются каскадная схема и модели ЦЛП. Важным обстоятельством при этом является то, что уже на этапе назначения подготовительных агрегатов для фиксированного множества требований система ограничений может оказаться несовместной. Именно в этой связи, то есть для снижения рисков противоречивости входных данных, этап размещения требований выделяется в самостоятельную задачу.

Обозначим через  $F$  множество всех возможных назначений вида (2), и для каждого  $f \in F$  определим множество  $I_f$  следующим образом:



$$I_f = \bigcup_{i=1}^n \{t_i - \hat{\tau}_i; t_i - \tau_i\}. \quad (4)$$

Другими словами, множество  $I_f$  есть множество точек пересечения интервалов  $[t_i - \hat{\tau}_i; t_i - \tau_i]$ , а также точек концов интервалов по всем требованиям  $\sigma_i \in S$ . Например, для  $S = \{\sigma_1, \sigma_2\}$  и  $f(\sigma_1), f(\sigma_2)$ , таких, что  $[[t_2 - \hat{\tau}_2; t_2 - \tau_2] \subset [t_1 - \hat{\tau}_1; t_1 - \tau_1]$ , множество  $I_f$  примет вид

$$I_f = \{t_1 - \hat{\tau}_1, t_2 - \hat{\tau}_2, t_2 - \tau_2, t_1 - \tau_1\}.$$

Пусть длительность подготовки любого требования  $\sigma_i \in S$  постоянна для всех агрегатов  $k_1, \dots, k_l$  и равна  $\lambda$ . Для любых  $f \in F$  и  $d^k \in I_f, k = 1, |I_f| - 1$ , определим величину

$$L(d^k, f) = \frac{|d^{k+1} - d^k| l}{\lambda \left| \{f(\sigma_i) : d^k \leq t_i < d^{k+1}\} \right|} \quad (5)$$

и критерий

$$P(f) = \max_{k=1, |I_f|-1} \{L(d^k, f)\}. \quad (6)$$

Величина (5) по сути отражает количество требований, потенциально задействующих каждый из  $l$  агрегатов подготовки в момент времени  $d^k$ . При этом ясно, что внутри интервала  $[d^k, d^{k+1}]$  значение (5) не меняется, чем и обусловлено его вычисление только в точках из множества  $I_f$ . Таким образом, для заданных  $S, R, K$  и  $[T_0, T]$  задача размещения требований  $S$

по машинам  $R$  может быть сведена к оптимизационной задаче поиска отображения вида (2), такого, что

$$P(f) \rightarrow \min_{f \in F}. \quad (7)$$

Для решения задач (2) и (7) разработан генетический алгоритм.

### Генетический алгоритм размещения требований

Идеологически генетический алгоритм включает в себя этапы создания начальной популяции, селекции, скрещивания и мутации.

Пусть заданы множество требований  $S = \{\sigma_1, \dots, \sigma_n\}$ , период планирования  $[T_0, T]$  и множество подготовительных агрегатов  $K = \{k_1, \dots, k_l\}$  с фиксированной длительностью  $\lambda(k_1) = \dots = \lambda(k_l) = \lambda$  для подготовки любого требования  $\sigma_i \in S$ . Под особью, или представителем популяции, будем понимать назначение вида  $f(S) = \{f(\sigma_1), \dots, f(\sigma_n)\}$ . При этом различными генами особи  $f(S)$  являются назначения вида  $f(\sigma_i) = (r_i, t_i, m_i)$ . Обозначим через  $F^i$  популяцию для  $i$ -й итерации алгоритма, где

$$F^i = \{f_1(S), f_2(S), \dots\}.$$

Определим следующий набор параметров генетического алгоритма:

- размерность  $K_1$  для начальной популяции  $F^0$ ;
- отбор  $K_2$  представителей популяции в порядке возрастания значения критерия (6);
- формирование  $K_3$  потомков для каждой пары отобранных представителей;
- мутация  $K_4$  генов;
- ограничение  $\tau$  (мин.) для процедуры мутации генов;
- критерий останова по совокупному числу  $K_5$  эпох;
- критерий  $\varepsilon$  (%) улучшения целевого функционала для последовательных итераций алгоритма;
- критерий останова по числу  $K_6$  эпох (итераций алгоритма) без улучшения целевого функционала.

Далее при описании алгоритмов после символа // дается комментарий к соответствующей строке.

### Генетический алгоритм размещения требований:

1. Выполнить процедуру формирования начальной популяции  $F^0$  из  $K_1$  особей
2.  $k = 0, c = 0$  // счетчики совокупного числа эпох и числа эпох без улучшения целевого функционала

3. Выполнить процедуру отсечения популяции  $F^0$
4. **Пока**  $k \neq K_5$  или  $c \neq K_6$
5. Выполнить процедуру отбора  $K_2$  представителей текущей популяции  $F^k$  в порядке возрастания критерия (6)
6. Выполнить процедуру скрещивания всех  $\frac{K_2 \cdot (K_2 - 1)}{2}$  пар представителей, отобранных на шаге 5 // увеличить счетчик числа эпох, включить в новую популяцию лучшего представителя текущей и сформировать  $K_3$  представитель-потомков для каждой пары
7. Выполнить процедуру мутации генов
8. Выполнить процедуру отсечения текущей популяции  $F^k$
9. **Вернуть**  $f : P(f) \rightarrow \min_{f \in F^k}$  // наилучший по критерию (6) представитель текущей популяции
10. **Если** минимальное значение критерия (6) на шаге 5 отклоняется от значения на шаге 9 менее, чем на  $\varepsilon$  (%), то
11. Увеличить счетчик числа эпох без улучшения целевого функционала

Рассмотрим подробнее этапы данного генетического алгоритма.

**Процедуры формирования начальной популяции и отсечения**

Для формирования начальной популяции  $F^0$  размещение  $f(S)$  требований производится случайным образом посредством выбора некоторых значений для машины, времени начала и длительности обработки из числа допустимых. Далее будем полагать, что время  $T_0, T$  определяются целым числом в формате timestamp (ts). Время  $t_i$  начала обработки каждого требования также будем определять в формате ts, а длительность обработки  $m_i$  – в формате целого числа (количество минут).

**Алгоритм 1.** Процедура формирования начальной популяции

1.  $F^0 = \{ \}$
2. **Для всех**  $k = \overline{1, n}$  // количество особей  $n = K_1$
3.  $f_k(S) = \{ \}$
4. **Для всех**  $\sigma_i \in S$
5.  $t_i = Rnd(T_0, T)$  // случайное число в диапазоне от  $T_0$  до  $T$
6.  $m_i = Rnd(\mu_i, \hat{\mu}_i)$
7.  $r_i = Rnd(1, |\rho_i|)$
8.  $f(\sigma_i) = (r_i, t_i, m_i)$

9.  $f_k(S) = f_k(S) + \{f(\sigma_i)\}$  // сформировать ген  $f(\sigma_i)$  для текущей особи  $f_k(S)$
10.  $F^0 = F^0 + \{f_k(S)\}$
11. **Вернуть**  $F^0 = \{f_1(S), \dots, f_n(S)\}$

Фрагмент начальной популяции из двух особей с пятью генами отображен на рисунке 5. Например, для требования  $\sigma_1$  на итерации  $k = 1$  внешнего цикла алгоритма 1 установлена машина обработки  $r_1 = r^1$ , а на итерации  $k = 2$  машина  $r_1 = r^2$ . Аналогично для итераций  $k = 1$  и  $k = 2$  на рисунке 5 представлены различные значения для  $t_1$  и  $m_1$  и т.д. для всех рассматриваемых требований  $\sigma_i, i = \overline{1, 5}$ .

Таким образом, в строках 5–7 алгоритма 1 для каждого требования  $\sigma_i \in S$  выбираются произвольные время начала  $t_i \in [T_0, T]$ , длительность  $m_i \in [\mu_i, \hat{\mu}_i]$  и машина обработки  $r_i \in \rho_i$ . При этом ясно, что некоторые особи  $f(S) \in F^0$  могут оказаться противоречивыми с точки зрения ограничений (3) задачи (2) (время начала и длительность обработки требований  $\sigma_i, \sigma_j$  при их последовательном размещении на одну и ту же машину  $r_i = r_j$ ). На рисунке 6 представлен пример особи  $f(S)$  с противоречивыми генами  $f(\sigma_1)$  и  $f(\sigma_2)$ , такими, что  $r_1 = r_2, t_1 < t_2$  и  $t_1 + m_1 > t_2$ .

Процедура отсечения популяции выполняется для начальной популяции и на каждой итерации внешнего цикла генетического алгоритма, обеспечивая тем самым рассмотрение только допустимых назначений вида (2). При этом размерность текущей популяции после

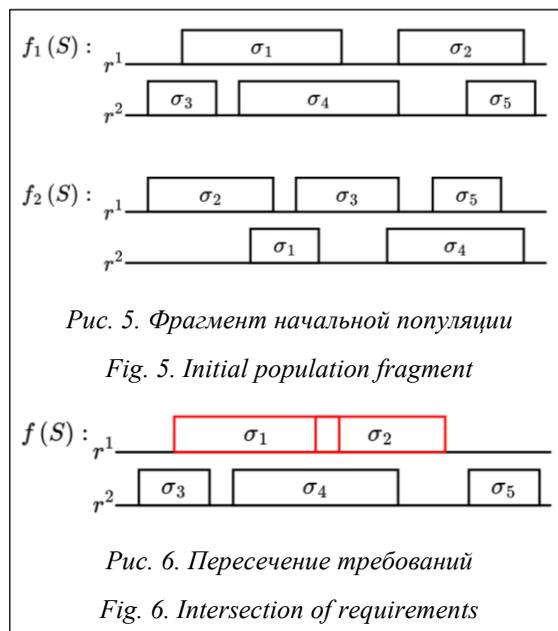


Рис. 5. Фрагмент начальной популяции  
 Fig. 5. Initial population fragment

Рис. 6. Пересечение требований  
 Fig. 6. Intersection of requirements

выполнения процедуры отсечения во многом определяется значениями параметров  $K_1$  и  $K_2$  алгоритма. Так, в случае тенденции вымирания соответствующие параметры могут быть существенно увеличены в целях стабилизации размерности.

**Алгоритм 2.** Процедура отсечения популяции

1. Для всех  $f(S) \in F^k$
2. Для всех  $f(\sigma_i) \in f(S)$  // по всем парам различных генов
3. Для всех  $f(\sigma_j) \in f(S) | \sigma_i \neq \sigma_j$
4. Если  $r_i = r_j$  и  $t_j \leq t_i$  и  $t_i + m_j > t_i$  то // ограничения (3) нарушены
5.  $F^k = F^k - \{f(S)\}$  // исключить особь из текущей популяции
6. Выход из цикла
7. Если  $r_i = r_j$  и  $t_i < t_j$  и  $t_j + m_j > t_i$  то
8.  $F^k = F^k - \{f(S)\}$
9. Выход из цикла
10. Вернуть  $F^k$

**Процедура отбора представителей**

Рассмотрим начальную популяцию:

$$F^0 = \{f_1(S), f_2(S), \dots\},$$

где

$$f_j(S) = \{f_j(\sigma_1), f_j(\sigma_2), \dots\} \text{ для всех } j \in \overline{1, K_1}(S), f_{j_2}(S), \dots\}, \text{ где } L_{j_1} \leq L_{j_2} \leq \dots$$

2, ...  
и

$$f_j(\sigma_i) = (r_i, t_i, m_i) \text{ для всех } \sigma_i \in S.$$

Напомним, что любое размещение требований  $f_j(S) \in F^0$  является допустимым с точки зрения ограничений задачи (2) ввиду структуры алгоритма 1 и с учетом выполнения процедуры отсечения согласно алгоритму 2.

Для каждого представителя  $f_j(S) \in F^0$  определим множество (4) и значение критерия (6) при фиксированных  $l$  и  $\lambda$  для рассматриваемого количества подготовительных агрегатов и длительности подготовки каждого требования соответственно. На рисунке 7 приведен пример множества  $I_f$  для некоторого представителя  $f(S)$  с пятью генами  $f(\sigma_i), i = \overline{1, 5}$  – точки на верхней горизонтальной прямой.

Аналогично множество  $I_f$  и значение критерия (6) могут быть определены для представителей любой популяции  $F^k$  (условие допустимости всех представителей  $f_j(S)$  произвольной популяции  $F^k$  с точки зрения ограничений задачи (2) будут обоснованы несколько позже на этапе обсуждения процедуры мутации генов).

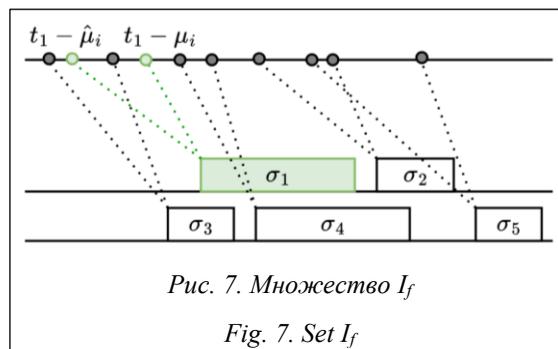


Рис. 7. Множество  $I_f$

Fig. 7. Set  $I_f$

**Алгоритм 3.** Процедура отбора представителей

1. Для всех  $f_j(S) \in F^k$
2.  $f = f_j(S)$
3. Сформировать множество точек  $I_f = \{d^1, d^2, \dots\}$
4. Для всех  $d^n \in I_f$  //  $n = \overline{1, |I_f| - 1}$
5.  $L(d^n, f) = \frac{|d^{k+1} - d^k| l}{\lambda |\{f(\sigma_i) \in f : d^k \leq t_i < d^{k+1}\}|}$
6.  $L_j = \max_n \{L(d^n, f)\}$
7. Вернуть  $\{L_1, L_2, \dots\}$
8. Отсортировать  $F^k$  в порядке возрастания значений  $L_j$  ее представителей:  $\{f_{j_1}(S), \dots, f_{j_n}(S)\}$
9. Вернуть  $F^k = \{f_{j_1}(S), \dots, f_{j_n}(S)\}$  //  $n = K_2$
10. Вернуть  $f^* = f_{j_1}(S)$  // наилучший представитель текущей популяции

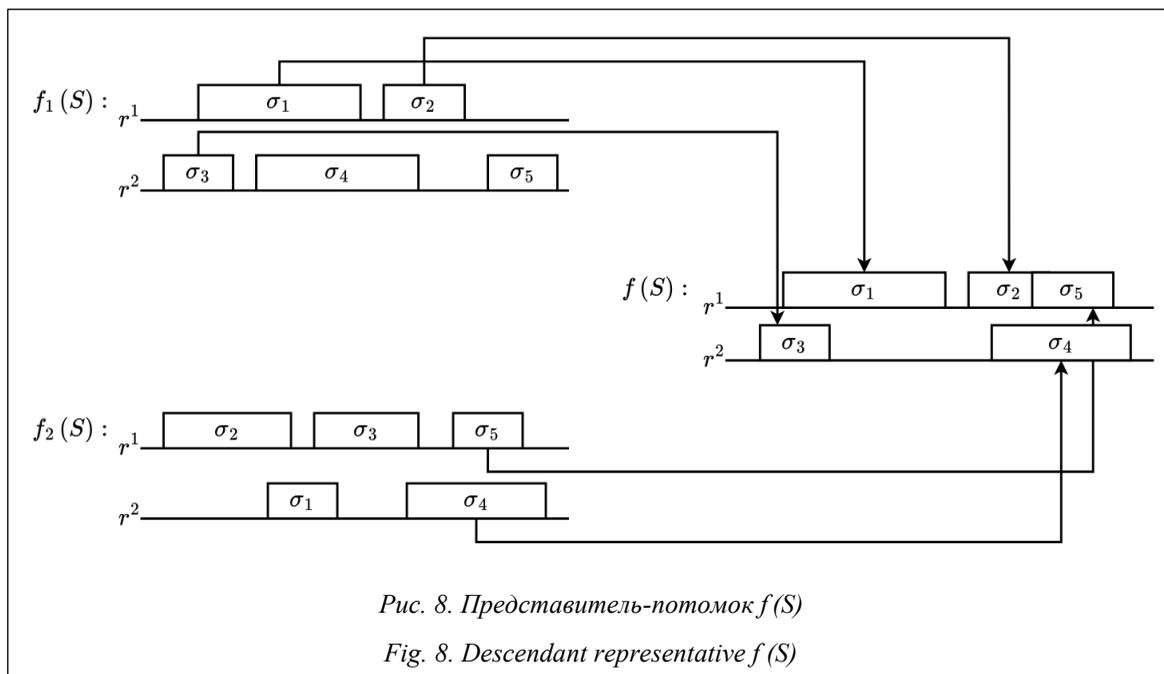
**Процедуры скрещивания и мутации**

Рассмотрим популяцию  $F^k = \{f_{j_1}(S), \dots, f_{j_n}(S)\}$

, построенную в результате выполнения алгоритма 3 как подмножество  $n = K_2$  лучших представителей по критерию (6). Для каждой пары представителей  $f_i(S), f_j(S) \in F^k$  гены представителя-потомка  $f(S)$  определяются посредством прямого наследования от случайным образом выбранного родителя  $f_i(S)$  или  $f_j(S)$ . На рисунке 8 приведен пример формирования представителя потомка  $f(S)$  для родителей  $f_1(S), f_2(S)$  с генами  $f(\sigma_i), i = \overline{1, 5}$ , где гены  $f(\sigma_i), i = \overline{1, 3}$ , наследуются от родителя  $f_1(S)$  и гены  $f(\sigma_i), i = \overline{4, 5}$ , от родителя  $f_2(S)$  соответственно.

**Алгоритм 4.** Процедура скрещивания

1.  $F = F^k, k = k + 1, F^k = \{f^*\}$
2. Для всех  $f_i(S) \in F, i = \overline{1, n}$  //  $n = K_2$



3. Для всех  $f_j(S) \in F, j = \overline{i+1, n}$  // для всех пар родителей
4. Для всех  $l = \overline{1, m}$  // сформировать  $m = K_3$  потомков
5.  $f_l(S) = \{\}$
6. Для всех  $\sigma \in S$
7.  $p = Rnd(0, 1)$
8. Если  $p \leq 0.5$  то
9.  $f(\sigma) = f_i(\sigma)$  // наследовать текущий ген от родителя  $f_i(S)$
10. Иначе
11.  $f(\sigma) = f_j(\sigma)$
12.  $f_l(S) = f_l(S) + \{f(\sigma)\}$
13.  $F^k = F^k + \{f_l(S)\}$
14. Вернуть  $F^k, k$

На шаге 1 алгоритма 4 формируется новая популяция  $F^k$ , в состав которой включается наилучший по критерию (6) представитель  $f^*$  предшествующей популяции  $F$ . Данная процедура обеспечивает сохранение на каждой итерации текущего локально оптимального решения и последующий контроль числа  $K_6$  итераций без существенного улучшения целевого функционала – параметр  $\varepsilon$  (%) алгоритма.

Для каждого представителя текущей популяции  $f(S) \in F^k$  случайным образом выбранные  $K_4$  генов  $f(\sigma)$  мутируют в части времени начала и длительности исполнения соответствующего требования  $\sigma \in S$  по процедуре, описанной в алгоритме 1.

**Алгоритм 5.** Процедура мутации генов

1. Для всех  $f(S) \in F^k$
2. Для всех  $i = \overline{1, n}$  // мутация  $n = K_4$  случайно выбранных генов
3.  $j = Rnd(1, |S|)$  // выбор произвольного  $\sigma \in S$
4.  $t_j = t_j + Rnd(-\tau, \tau)$
5.  $f(\sigma_j) = (r_j, t_j, m_j)$
6. Вернуть  $F^k$

Аналогично алгоритмам 1 и 2 структура алгоритма 5 и последующий вызов процедуры отсечения в строке 8 генетического алгоритма обеспечивают допустимость любого размещения требований  $f(S) \in F^k$  с точки зрения ограничений задачи (2).

**Вычислительный эксперимент**

Предложенный генетический алгоритм размещения требований, включая процедуры, описанные в алгоритмах 1–5, были реализованы на языке Python 3.8 как автономный модуль комплексной системы планирования производственных процессов потокового типа [2]. Анализ эффективности применения генетического алгоритма на предварительном этапе решения комплексной задачи проводится в двух направлениях: с точек зрения оптимизации входных данных для задачи о назначении подготовительных агрегатов (в случае ее исходной противоречивости) и трудоемкости этой задачи (в случае наличия решения для исходной по-

становки). Общая структура вычислительного эксперимента представлена на рисунке 9.

Вычислительный эксперимент был проведен на реальных производственных данных. Рассматриваются годовой период и множество требований  $S = \{\sigma_1, \dots, \sigma_n\}$ , подлежащих исполнению в каждые сутки рассматриваемого периода. Таким образом, для каждого экземпляра входных данных период планирования  $[T_0, T]$  составляет 24 часа. Множество  $S$  содержит  $n = 100$  требований с параметрами:

- $30 \leq \tau_i < \hat{\tau}_i \leq 180$  (мин.),
- $30 \leq \mu_i < \hat{\mu}_i \leq 60$  (мин.),
- $\rho_i = R$

для всех  $i = \overline{1, 100}$ , где  $R = \{r_j\}, j = \overline{1, 5}$  – множество машин, доступных для исполнения требований множества  $S$ . Для множества подготовительных агрегатов  $K = \{k_l\}, l = \overline{1, 3}$ , длительность обработки каждого требования фиксированная и составляет  $\lambda = 40$  (мин.).

Определим параметры генетического алгоритма:

- размерность начальной популяции  $K_1 = 1\,000$  представителей;
- отбор  $K_2 = 30$  представителей популяции в порядке возрастания значения критерия (6);
- формирование  $K_3 = 50$  потомков для каждой пары отобранных представителей;
- мутация  $K_4 = 2$  гена;
- ограничение  $\tau = 5$  (мин.) для процедуры мутации генов;

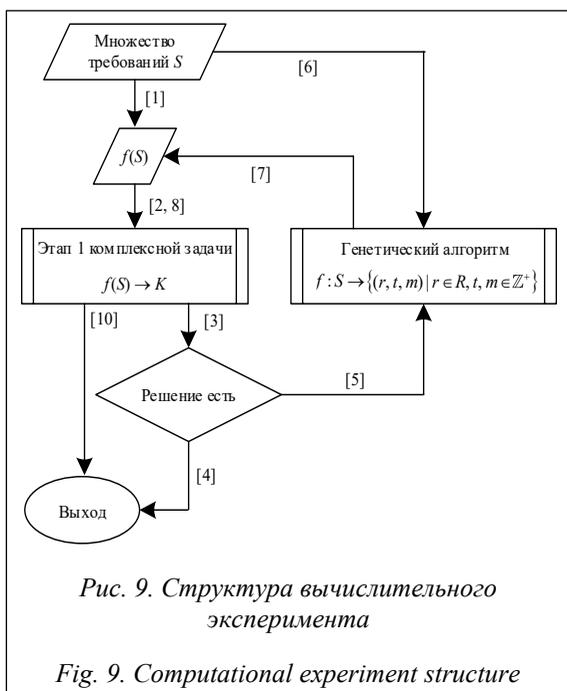


Рис. 9. Структура вычислительного эксперимента

Fig. 9. Computational experiment structure

- критерий останова по числу  $K_5 = 10$  эпох (итераций алгоритма) без улучшения целевого функционала;

- критерий  $\epsilon = 10$  (%) улучшения целевого функционала для последовательных итераций алгоритма;

- критерий останова по совокупному числу  $K_6 = 500$  эпох.

Замечание 3. Если размерность популяции на некоторой итерации алгоритма не превышает значение параметра  $K_2 = 30$ , то процедура формирования  $K_3 = 50$  потомков производится для каждой пары представителей соответствующей популяции.

На рисунке 10 показано соотношение величины [5] в каждой точке множества  $I_f$  соответственно для размещения, сформированного посредством жадного алгоритма (см. поток [1] на рисунке 9), и для размещения, построенного с использованием генетического алгоритма (см. потоки [6] и [7] на рисунке 9).

В среднем значение величины  $L(d^k, f)$  (потенциальная нагрузка на подготовительные агрегаты) составляет 0,6 в обоих случаях. Однако разброс значений по всем точкам  $d^k \in I_f$  для размещения, построенного с использованием генетического алгоритма, существенно меньше и равен [0,35;0,83] по сравнению с [0,3;1,39] соответственно для размещения, сформированного посредством жадного алгоритма.

В примере на графике значение максимума потенциальной нагрузки на подготовительные агрегаты (6) составляет 1,39 и 0,83 для жадного и генетического алгоритмов соответственно. Аналогичные расчеты были проведены для каждого экземпляра входных данных в рассматриваемом годовом периоде. Результаты сравнительного анализа по критерию (6) представлены на графике (<http://www.swsys.ru/uploaded/image/2025-1/25.jpg>).

Диапазон значений критерия (6) в случае размещения, сформированного посредством жадного алгоритма, составляет [1,01;2,8] со средним значением 1,91. В то же время для размещения, построенного с использованием генетического алгоритма, диапазон значений критерия (6) составляет [0,8;1,27] со средним значением 1,07.

На последующем этапе назначения подготовительных агрегатов (см. потоки [2] и [8] на рисунке 9) фиксировалось время в минутах, затраченное системой на поиск решения. Для случаев, когда решение не было найдено, данный показатель принимался равным 0 (см. потоки [3] и [5] на рисунке 9). Проведен анализ

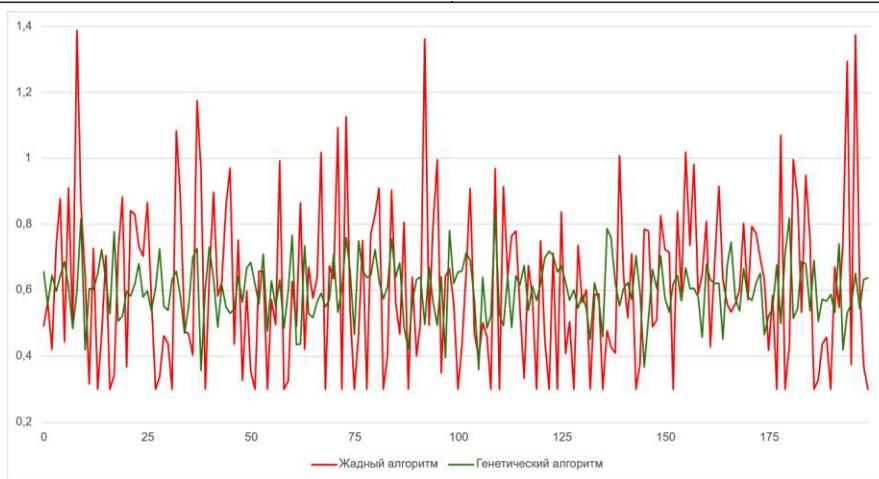


Рис. 10. Значение  $L(d^k, f)$  для точек множества  $I_f$

Fig. 10. The value of  $L(d^k, f)$  for the points of set  $I_f$

данного показателя временных затрат для каждого экземпляра входных данных в рассматриваемом годовом периоде (<http://www.swsys.ru/uploaded/image/2025-1/26.jpg>).

Можно сделать вывод, что решения нет во всех случаях, когда значение критерия (6) превышает 2,1. Доля таких случаев для размещения, построенного с использованием жадного алгоритма, составляет 40 %, а для размещения, сформированного посредством генетического алгоритма, значение критерия (6) не превышает 1,27 в 100 % случаев. В 60 % случаев, когда решение было найдено для размещения, построенного по жадному алгоритму, время поиска решения варьируется в диапазоне [2,01;6,96] со средним значением 2,71 мин. Для размещения, построенного по генетическому алгоритму, диапазон времени поиска решения на этапе назначения подготовительных агрегатов составляет [1;2,92] со средним значением 1,97 мин.

Из результатов проведенного вычислительного эксперимента следует, что применение генетического алгоритма на этапе размещения требований существенно снижает критерий максимальной потенциальной нагрузки на подготовительные агрегаты по сравнению с решением данного этапа посредством жадного алгоритма. Последующий этап назначения подготовительных агрегатов оказывается разрешимым в 100 % случаев в рассматриваемом годовом периоде, в то время как для жадного размещения в 40 % случаев решение не найдено. Наконец, сравнительный анализ временных затрат демонстрирует потенциал ускорения в среднем до 27 % в случаях, когда для

жадного размещения найдено решение на этапе назначения подготовительных агрегатов.

### Заключение

В статье рассмотрена задача размещения требований как начальный этап комплексного планирования производственных процессов потокового типа. Такая декомпозиция обусловлена прикладными аспектами комплексной задачи, а также существенным потенциалом для снижения исходной размерности. Кроме того, рассмотрение вспомогательного этапа размещения требований позволяет ожидать разрешимости последующих этапов в случае противоречивости ограничений. Для задачи размещения требований в рассмотрение введен эвристический критерий потенциальной нагрузки на подготовительные агрегаты и предложен генетический алгоритм решения. Подробно обсуждаются ключевые процедуры алгоритма. Предложенный генетический алгоритм реализуется на языке Python как автономный модуль системы планирования производственных процессов потокового типа. Вычислительный эксперимент с использованием этой системы проводится на реальных производственных данных по двум направлениям: с точек зрения разрешимости последующего этапа назначения подготовительных агрегатов и трудоемкости поиска решения для случаев размещения требований с использованием жадного подхода и генетического алгоритма соответственно. Результаты вычислительного эксперимента демонстрируют высокую эффективность генетического алгоритма по обоим аспектам сравнительного ана-

лиза. В частности, этап назначения подготовительных агрегатов оказывается разрешимым в 100 % случаев с использованием генетического алгоритма для предварительного размещения требований. С точки зрения вычислительной трудоемкости использование генетического алгоритма в проведенном эксперименте влечет ускорение до 27 % в среднем.

Дальнейшее развитие полученных результатов связано с исследованием теоретических вопросов сходимости и устойчивости предложенного генетического алгоритма. Важным этапом при этом выступает проведение мас-

штабного вычислительного эксперимента с использованием данных тестовых библиотек. Также ставится задача разработки и реализации дополнительных оригинальных тестовых задач, позволяющих адекватно оценить поведение и свойства алгоритма. С практической точки зрения направление дальнейшего развития связано с исследованием возможностей применимости предложенного метаэвристического подхода для решения задачи планирования производственных процессов потокового типа на других этапах технологической цепочки.

### Список литературы

1. Ризванов Д.А., Чернышев Е.С. Информационное и алгоритмическое обеспечение планирования производственных мощностей // Интеллектуальные системы в производстве. 2020. Т. 18. № 4. С. 117–125. doi: 10.22213/2410-9304-2020-4-117-125.
2. Махитко В.П., Рамзаев В.М., Гришанов Г.М. Экономические аспекты имитационного моделирования производственных участков в среде Technomatix Plant Simulation // Вестн. Самарского муниципального института управления. 2019. № 3. С. 7–16.
3. Никишечкин П.А., Ивашин С.С., Черненко В.Е. и др. Система имитационного моделирования PlantTwin как инструмент верификации производственных планов и поддержки принятия решений для повышения эффективности производства // Вестн. машиностроения. 2021. № 3. С. 80–85. doi: 10.36652/0042-4633-2021-3-80-85.
4. Александров В.Р., Баранов С.Е., Кузнецов М.И. и др. Искусственный интеллект в задачах планирования производства // Инфокоммуникационные и радиоэлектронные технологии. 2022. Т. 5. № 2. С. 196–208. doi: 10.29039/2587-9936.2022.05.2.14.
5. Гайнанов Д.Н., Беренов Д.А., Рассказова В.А. и др. Data-PLAN: Свид. о регистр. ПрЭВМ № 2021665276. Рос. Федерация, 2021.
6. Частикова В.А., Чич А.И. Генетические алгоритмы и генетическое программирование: особенности реализации // Перспективы науки. 2019. № 1. С. 13–16.
7. Минитаева А.М., Векшин Р.Д., Шатилов А.А. Анализ различных видов генетических алгоритмов в задачах оптимизации // Технологии инженерных и информационных систем. 2022. № 1. С. 21–34.
8. Курейчик В.М., Данильченко В.И. Генетический алгоритм планирования размещения СБИС // Изв. ЮФУ. Технич. науки. 2019. № 2. С. 26–34.
9. Сулов Д.А., Сенченко К.А., Шмаль В.Н. Применение генетических алгоритмов в сфере организации пассажирских перевозок // Дневник науки. 2022. № 9. URL: [https://dnevniknauki.ru/images/publications/2022/9/tech-nics/Suslov\\_Senchenko\\_Shmal.pdf](https://dnevniknauki.ru/images/publications/2022/9/tech-nics/Suslov_Senchenko_Shmal.pdf) (дата обращения: 23.07.2024).
10. Сидоренко В.Г., Сафронов А.И. Применение генетических алгоритмов при решении задач планирования перевозочного процесса городской рельсовой транспортной системы // Автоматика на транспорте. 2023. Т. 9. № 1. С. 49–62. doi: 10.20295/2412-9186-2023-9-01-49-62.
11. Гусев П.Ю., Гусев К.Ю., Вахмин С.Ю. Применение генетических алгоритмов в оптимизации планировочных решений производственных подразделений машиностроительных предприятий // Вестн. ВГТУ. 2019. Т. 15. № 2. С. 22–28.
12. Семенов Г.Е., Кейно П.П. Применение математических моделей на основе генетических алгоритмов в задачах планирования сложных технических объектов // Прикладная информатика. 2019. Т. 14. № 2. С. 56–62.
13. Liang Z., Zhong P., Liu M., Zhang Ch., Zhang Z. A computational efficient optimization of flow shop scheduling problems. *Sci. Reports*, 2022, vol. 12, art. 845. doi: 10.1038/s41598-022-04887-8.
14. Türkakın O.H., Arditi D., Manisali E. Comparison of heuristic priority rules in the solution of the resource-constrained project scheduling problem. *Sustainability*, 2021, vol. 13, no. 17, art. 9956. doi: 10.3390/su13179956.
15. Abdel-Basset M., Manogaran G., El-Shahat D., Mirjalili S. A hybrid whale optimization algorithm based on local search strategy for the permutation flow shop scheduling problem. *FGCS*, 2018, vol. 85, pp. 129–145. doi: 10.1016/j.future.2018.03.020.
16. Kadarkarainadar M.M., Tosun Ö., Geetha M. Hybrid monkey search algorithm for flow shop scheduling problem under makespan and total flow time. *Applied Soft Computing J.*, 2017, vol. 55, pp. 82–92. doi: 10.1016/j.asoc.2017.02.003.
17. Гончаров Е.Н. Алгоритм локального поиска для задачи календарного планирования с ограниченными ресурсами // Дискретный анализ и исследование операций. 2022. Т. 29. № 4. С. 15–37.
18. Ladj A., Tayeb F.B.-S., Varnier C. Hybrid of metaheuristic approaches and fuzzy logic for the integrated flowshop scheduling with predictive maintenance problem under uncertainties. *EJIE*, 2021, vol. 15, no. 5, pp. 675–710. doi: 10.1504/EJIE.2021.117325.
19. Rasskazova V.A. LIP model in solving RCPSP at the flow type production. In: *CCIS. Proc. OPTIMA*, 2023, vol. 1913, pp. 75–88. doi: 10.1007/978-3-031-48751-4\_6.

20. Kibzun A.I., Rasskazova V.A. LIP model as mathematical ware for an optimal flow production planning system at operational scheduling stage. *Automation and Remote Control*, 2023, vol. 84, no. 5, pp. 529–542. doi: 10.1134/S0005117923050065.

Software &amp; Systems

doi: 10.15827/0236-235X.149.027-038

2025, 38(1), pp. 27–38

### Genetic algorithm for placing requirements in a flow-type production process planning problem

Andrey I. Kibzun<sup>1</sup>, Varvara A. Rasskazova<sup>1</sup>✉

<sup>1</sup>Institute of Computer Sciences and Applied Mathematics, Moscow Aviation Institute  
(National Research University) (MAI), Moscow, 125993, Russian Federation

#### For citation

Kibzun, A.I., Rasskazova, V.A. (2025) ‘Genetic algorithm for placing requirements in a flow-type production process planning problem’, *Software & Systems*, 38(1), pp. 27–38 (in Russ.). doi: 10.15827/0236-235X.149.027-038

#### Article info

Received: 18.07.2024

After revision: 14.08.2024

Accepted: 23.08.2024

**Abstract.** The paper discusses the problem of planning flow-type production processes. In terms of a cascade scheme, the complex solution covers the stage of assigning preparatory units and the subsequent stage of forming detailed technological routes to fulfill a given set of requirements on time and taking into account the constraints on permissible processing durations at each processing stage. This scheme comes as a part of a problem-oriented computing complex. However, due to a number of natural reasons, the problem may become inconsistent right at the stage of assigning preparatory units. One of the ways to overcome these difficulties is to develop and implement penalty function algorithms to find the maximum joint subsystems in inconsistent optimization problems. The paper proposes an ideologically different approach for this purpose. It is based on considering the preliminary stage of requirement placement in such a way that the subsequent stages of problem-solving process are guaranteed to be solvable. The requirement placement is formalized as a search for an optimal mapping that minimizes “potential” workload on preparatory units during the planning period. To solve this problem, the authors of the paper have developed a genetic algorithm, which resulted in a significant advantage in terms of speed in comparison with fundamental approaches of mathematical programming (for example, integer linear programming models). In order to reduce the risk of population extinction at each iteration of the genetic algorithm, the authors apply the rule of unconditional migration of a representative with the lowest criterion value. This approach also provides effective convergence indices of the algorithm in terms of the number of iterations without significant improvement of the objective function. The developed genetic algorithm is implemented as a stand-alone module of a computing system for solving process manufacturing scheduling problems. The authors conducted a computational experiment using this module in terms of a comparative analysis of the solution quality of the initial complex problem.

**Keywords:** genetic algorithm, computing complex, manufacturing planning, process manufacturing, schedule theory, combinatorial optimization

#### References

1. Rizvanov, D.A., Chernyshev, E.S. (2020) ‘Information and algorithmic support of production capacity planning’, *Intellekt. Sist. Proizv.*, 18(4), pp. 117–125 (in Russ.). doi: 10.22213/2410-9304-2020-4-117-125.
2. Makhitko, V.P., Ramzaev, V.M., Grishanov, G.M. (2019) ‘Economic aspects of simulation of production sites in the Technomatix Plant Simulation environment’, *Bull. of the Samara Municipal Institute of Management*, (3), pp. 7–16 (in Russ.).
3. Nikishechkin, P.A., Ivashin, S.S., Chernenko, V.E., Malyihanov, A.A., Dolgov, N.V. (2021) ‘PlantTwin simulation system as a tool for verifying production plans and supporting the decision-making to improve production effectiveness’, *Bull. of Mechanical Engineering*, (3), pp. 80–85 (in Russ.). doi: 10.36652/0042-4633-2021-3-80-85.
4. Aleksandrov, V.R., Baranov, S.E., Kuznetsov, M.I. et al. (2022) ‘Artificial intelligence in the tasks of manufacturing planning’, *Infocommunication and Radio Tech.*, 5(2), pp. 196–208 (in Russ.). doi: 10.29039/2587-9936.2022.05.2.14.
5. Gainanov, D.N., Berenov, D.A., Rasskazova, V.A. et al. (2021) *Data-PLAN*, Pat. RF, № 2021665276.
6. Chastikova, V.A., Chich, A.I. (2019) ‘Genetic algorithms and genetic programming: features of realization’, *Sci. Prospects*, (1), pp. 13–16 (in Russ.).
7. Minitaeva, A.M., Vekshin, R.D., Shatilov, A.A. (2022) ‘Analysis of various types of genetic algorithms in optimization problems’, *Tech. of Eng. and Inform. Sys.*, (1), pp. 21–34 (in Russ.).
8. Kureychik, V.M., Danilchenko, V.I. (2019) ‘Genetic algorithm of VLSI placement planning’, *Izv. SFedU. Eng. Sci.*, (2), pp. 26–34 (in Russ.).

9. Suslov, D.A., Senchenko, K.A., Shmal, V.N. (2022) 'Application of genetic algorithms in the field of organizing passenger transportation', *Diary of Sci.*, (9), available at: [https://dnevniknauki.ru/images/publications/2022/9/tech-nics/Suslov\\_Senchenko\\_Shmal.pdf](https://dnevniknauki.ru/images/publications/2022/9/tech-nics/Suslov_Senchenko_Shmal.pdf) (accessed July 23, 2024) (in Russ.).
10. Sidorenko, V.G., Safronov, A.I. (2023) 'Application of genetic algorithms at solution of tasks for transportation process planning of city rail transport system', *Transport Automation Research*, 9(1), pp. 49–62 (in Russ.). doi: 10.20295/2412-9186-2023-9-01-49-62.
11. Gusev, P.Yu., Gusev, K.Yu., Vakhmin, S.Yu. (2019) 'Modeling of the dissolution and growth of sugar crystals', *Bull. of VSTU*, 15(2), pp. 22–28 (in Russ.).
12. Semenov, G.E., Keino, P.P. (2019) 'Mathematical model of genetic algorithm in implementation for scheduling tasks of complex technical objects during pre-production stages', *Applied Inform.*, 14(2), pp. 56–62 (in Russ.).
13. Liang, Z., Zhong, P., Liu, M., Zhang, Ch., Zhang, Z. (2022) 'A computational efficient optimization of flow shop scheduling problems', *Sci. Reports*, 12, art. 845. doi: 10.1038/s41598-022-04887-8.
14. Türkakın, O.H., Arditi, D., Manisali, E. (2021) 'Comparison of heuristic priority rules in the solution of the resource-constrained project scheduling problem', *Sustainability*, 13(17), art. 9956. doi: 10.3390/su13179956.
15. Abdel-Basset, M., Manogaran, G., El-Shahat, D., Mirjalili, S. (2018) 'A hybrid whale optimization algorithm based on local search strategy for the permutation flow shop scheduling problem', *FGCS*, 85, pp. 129–145. doi: 10.1016/j.future.2018.03.020.
16. Kadarkarainadar, M.M., Tosun, Ö., Geetha, M. (2017) 'Hybrid monkey search algorithm for flow shop scheduling problem under makespan and total flow time', *Applied Soft Computing J.*, 55, pp. 82–92. doi: 10.1016/j.asoc.2017.02.003.
17. Goncharov, E.N. (2022) 'A local search algorithm for a scheduling problem with limited resources', *Discrete Analysis and Operations Research*, 29(4), pp. 15–37 (in Russ.).
18. Ladj, A., Tayeb, F.B.-S., Varnier, C. (2021) 'Hybrid of metaheuristic approaches and fuzzy logic for the integrated flowshop scheduling with predictive maintenance problem under uncertainties', *EJIE*, 15(5), pp. 675–710. doi: 10.1504/EJIE.2021.117325.
19. Rasskazova, V.A. (2023) 'LIP model in solving RCPSP at the flow type production', in *CCIS. Proc. OPTIMA*, 1913, pp. 75–88. doi: 10.1007/978-3-031-48751-4\_6.
20. Kibzun, A.I., Rasskazova, V.A., (2023) 'LIP model as mathematical ware for an optimal flow production planning system at operational scheduling stage', *Automation and Remote Control*, 84(5), pp. 529–542. doi: 10.1134/S0005117923050065.

**Авторы****Кибзун Андрей Иванович**<sup>1</sup>,д.ф.-м.н., профессор,  
заведующий кафедрой,  
kibzun@mail.ru**Рассказова Варвара Андреевна**<sup>1</sup>,к.ф.-м.н., доцент,  
varvara.rasskazova@mail.ru**Authors****Andrey I. Kibzun**<sup>1</sup>,Dr.Sci. (Physics and Mathematics),  
Professor, Head of Chair,  
kibzun@mail.ru**Varvara A. Rasskazova**<sup>1</sup>,Cand. of Sci. (Physics and Mathematics),  
Associate Professor, varvara.rasskazova@mail.ru<sup>1</sup> Институт компьютерных наук и прикладной математики, Московский авиационный институт (национальный исследовательский университет) (МАИ), г. Москва, 125993, Россия<sup>1</sup> Institute of Computer Sciences and Applied Mathematics, Moscow Aviation Institute (National Research University) (MAI), Moscow, 125993, Russian Federation

## Гибридный подход к выделению структурированных данных из «Летописи жизни и творчества А.С. Пушкина»

П.П. Кокорин<sup>1</sup>, А.А. Котов<sup>1</sup>, С.В. Кулешов<sup>1</sup>, А.А. Зайцева<sup>1</sup> ✉

<sup>1</sup> Санкт-Петербургский федеральный исследовательский центр РАН, г. Санкт-Петербург, 199178, Россия

### Ссылка для цитирования

Кокорин П.П., Котов А.А., Кулешов С.В., Зайцева А.А. Гибридный подход к выделению структурированных данных из «Летописи жизни и творчества А.С. Пушкина» // Программные продукты и системы. 2025. Т. 38. № 1. С. 39–46. doi: 10.15827/0236-235X.149.039-046

### Информация о статье

Группа специальностей ВАК: 2.3.1

Поступила в редакцию: 12.07.2024

После доработки: 13.08.2024

Принята к публикации: 23.08.2024

**Аннотация.** Статья посвящена решению проблемы создания программной инфраструктуры для систематизации, аннотирования, хранения, поиска и публикации рукописей и иных материалов, представленных в цифровом виде. Исследование построено на материалах, связанных с жизнью и творчеством А.С. Пушкина и составляющих важную часть научно-просветительского ресурса «Пушкин цифровой». Актуальность решаемой проблемы обусловлена необходимостью сохранения авторского наследия русских писателей в условиях цифровой трансформации предметной области филологических, источниковедческих и библиографических исследований их трудов, что является частью национальных проектов Российской Федерации «Образование», «Культура», «Наука и университеты». В данном контексте особую роль играет решение задачи извлечения структурированного текста из растровых изображений страниц томов «Летописи жизни и творчества А.С. Пушкина» для использования в разрабатываемых системах хранения, систематизации, публикации материалов библиотечных, архивных, музейных, фонографических и иных фондов и коллекций и частичной автоматизации филологических, источниковедческих и библиографических исследований. В работе предложен гибридный подход, основанный на использовании априорных данных о структуре элементов верстки страницы, технологиях OCR – распознавание текста на базе библиотеки Tesseract и автоматизированной обработке больших объемов оцифрованных авторских текстов, архивных и других бумажных документов. Особенностью разработанных методов верификации является использование регулярных выражений для извлечения структурированных данных из предварительно распознанного текста и автоматизированного конвейера обработки текстов в сборочной системе GitLab. Приведены результаты применения предложенного гибридного подхода. Показано, что этот подход дает удовлетворительные результаты, обеспечивая минимизацию ручной постобработки полученных данных путем вычитки результатов, размещаемых на научно-просветительском ресурсе. Полученные результаты могут использоваться не только в разрабатываемом ресурсе «Пушкин цифровой», но и в других проектах, в основе реализации которых лежит необходимость распознавания и автоматизированной обработки больших объемов оцифрованных авторских текстов, архивных и других бумажных документов.

**Ключевые слова:** распознавание текста, извлечение данных, структурированные данные, обработка текста  
**Благодарности.** Работа выполнена в рамках реализации Государственного задания № FFZF-2023-0001

**Введение.** Сохранение авторского наследия русских писателей в условиях цифровой трансформации предметной области филологических, источниковедческих и библиографических исследований их трудов является частью национальных проектов Российской Федерации «Образование», «Культура», «Наука и университеты». К 225-летию юбилею А.С. Пушкина с целью сохранения культурного наследия великого поэта разрабатывался научно-просветительский ресурс «Пушкин цифровой». Важная составляющая этой работы – создание программной инфраструктуры для систематизации, аннотирования, хранения, поиска и публикации рукописей и иных материалов, представленных в цифровом виде, связанных с жизнью и творчеством А.С. Пушкина как одного из самых знаковых и узнаваемых во всем мире русских писателей [1–3].

Одним из ключевых источников данных для наполнения портала стала «Летопись жизни и творчества А.С. Пушкина» (далее – Летопись) [4]. К сожалению, исходных машиночитаемых структурированных данных издания не существует. В связи с этим задача извлечения структурированных текстовых данных, пригодных для использования в рамках научно-просветительского ресурса, трансформируется в задачу извлечения текста из растровых изображений страниц томов Летописи. Извлеченные текстовые данные должны быть пригодны для использования в разрабатываемых системах хранения, систематизации, публикации материалов библиотечных, архивных, музейных, фонографических и иных фондов, коллекций и частичной автоматизации филологических, источниковедческих и библиографических исследований.

Оптическое распознавание символов (*Optical Character Recognition, OCR*) – это процесс преобразования изображений в редактируемый текст. OCR-системы используются для сканирования документов и книг, а также для обработки фотографий и других изображений с текстом. Они могут быть применены для различных целей, включая оцифровку архивов, создание электронных книг, перевод текстов на другие языки и многое другое, а также для автоматизации процессов обработки документов, таких как заполнение форм или проверка данных. Однако, несмотря на все преимущества, OCR-системы неидеальны. Некоторые из них сталкиваются с проблемами распознавания сложных шрифтов или рукописного текста.

Существует большое количество коммерческих [5] и открытых [6, 7] программных средств, позволяющих производить OCR для документов и книг. При этом большинство таких систем хорошо справляются с переводом языковых токенов (отдельных слов и фрагментов предложений) в редактируемый текст, но имеют большое количество ошибок при распознавании и воспроизведении структуры (таблиц, абзацев, колонок), связанных с особенностями типографской верстки. Причем большинство исследователей в качестве основного инструмента распознавания используют нейросетевые модели, применяемые с различной степенью качества. Например, в [8] приведена одна из возможных реализаций решения задачи автоматизированного распознавания сущностей, основанная на дообучении языковой модели на архитектуре BERT, подключенной к библиотеке Spacy с использованием Spacy Transformers. Подобные нейросетевые инструменты ограниченно применимы в ситуациях, когда требуется точное соответствие извлекаемых данных структуре документа, поскольку разрабатываемый портал должен отвечать требованиям энциклопедичности и академичности.

Похожая проблема возникает при формировании машиночитаемых электронных словарей, например, в [9] распознавание макроструктуры и микроструктуры словарей основано на выделении границ словарных статей, границ зон внутри словарных статей в исходном тексте и их классификации. При этом авторы делают вывод о необходимости постоянной доработки алгоритмов под конкретную верстку словаря либо проверки распознанных данных вручную. Для облегчения разработки таких словарей в [10, 11] определены правила формирования терминологических кластеров и сетей.

Некоторые варианты атрибутирования текстов на основе анализа структуры рукописей по их отсканированным изображениям, а также возникающие проблемы описаны в [12].

В работе [13] сделана попытка перейти от посимвольного распознавания русскоязычных рукописных текстов к распознаванию строк текста с использованием нейросетевых подходов. Наиболее интересным с точки зрения сохранения структуры распознаваемых документов представляется описанный в [14] подход, направленный на распознавание русских рукописных текстов целыми абзацами. При этом результатом работы предложенных алгоритмов все равно является неструктурированный текст без выделения необходимых в отдельных случаях структурных элементов.

Для сокращения количества ошибок при автоматизированной обработке предлагается гибридная схема, использующая одновременно результат оптического распознавания текста всего документа в целом, не отражающий сведения о структурной разметке текста, а также совокупность результатов распознавания текста, отдельных фрагментов текста согласно шаблону страницы (рис. 1). Результаты объединения текстов сохраняются в БД, содержащей пригодные для использования на портале поля.

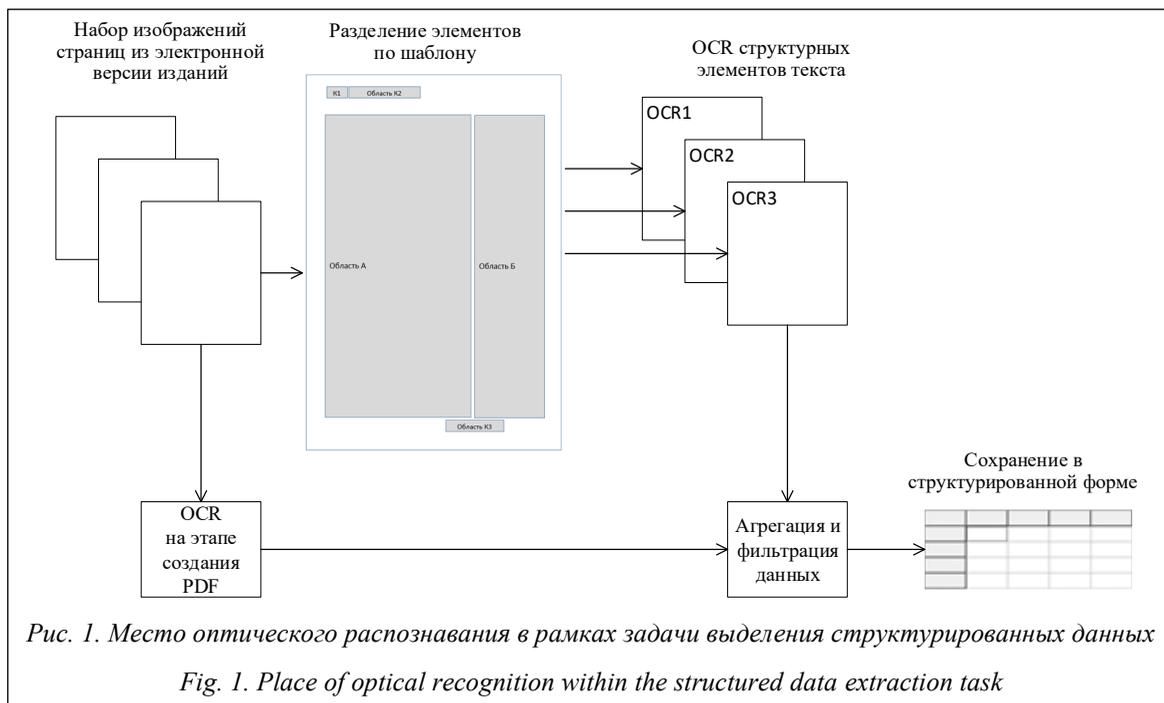
### Особенности исходного материала

Текст Летописи имеет сложную двухколоночную типографскую верстку: 1-я колонка – временной отрезок, место события и описание события Летописи; 2-я колонка – примечания, ссылки на библиографию и другие временные отрезки Летописи.

При этом в форматировании используется чередование (зеркальное отображение) колонок на четных и нечетных страницах (<http://www.swsys.ru/uploaded/image/2025-1/Kokorin.html>). Смена текущего года отмечается в верстке книги появлением отдельного абзаца с цифрами года (<http://www.swsys.ru/uploaded/image/2025-1/23.jpg>).

Кроме того, в верхнем и нижнем колонтитулах содержится дополнительная информация:

- текущий год Летописи;
- текущее место пребывания Пушкина А.С. в это время;
- номер страницы в пределах тома;
- сквозной номер страницы в пределах всех томов.



Все эти поля, а также соответствующий им текст, описывающий события, должны быть сохранены в БД для портала.

### Оптическое распознавание текста

Для предварительного извлечения структурированных данных из текста используются априорные знания о структуре страницы, задаваемые шаблоном верстки. Пример шаблона расположения областей документа для четных страниц приведен на рисунке 2. Для каждой области вызывается внешний модуль оптического распознавания (библиотека Tesseract), работающий с изображениями отдельной области. Конкретные размеры и положения областей распознавания корректируются эвристическим алгоритмом поиска областей, не заполненных текстом, между областями распознавания. После получения результата от модуля оптического распознавания текста записи Летописи и текста примечания (рис. 2) сопоставляются запись Летописи (1-я колонка, область А) с примечанием (2-я колонка, область Б); место, время и номера страниц выделяются из областей колонтитула (К1, К2, К3).

Такая верстка текста ориентирована на чтение человеком и малоприспособлена для автоматизированной компьютерной обработки, что и составляет основную сложность.

Сложность сопоставления записи Летописи и комментария определена различными вариантами взаимного выравнивания абзацев тек-

ста (<http://www.swsys.ru/uploaded/image/2025-1/24.jpg>): по верхнему краю текста Летописи и текста примечания, по середине – текста Летописи.

### Извлечение структурированных данных из текста

После первичного оптического распознавания формируется текстовый документ, содержащий ошибки распознавания и разметки.

Задача извлечения структурированных данных сводится к извлечению из распознанного текста списка записей Летописи с текстом примечания. Каждая запись Летописи должна сопровождаться фиксированным набором атрибутов:

- номер страницы в томе Летописи;
- сквозной номер страницы в пределах всех томов;
- год;
- год и место из верхнего колонтитула страницы;
- диапазон датировок записи;
- текст записи Летописи;
- текст связанного примечания.

Формат диапазона дат имеет большое количество вариантов:

- год (например, 1799);
- год, месяц, день (например, 1800. Декабрь, 2);
- год, месяц, день и время дня;

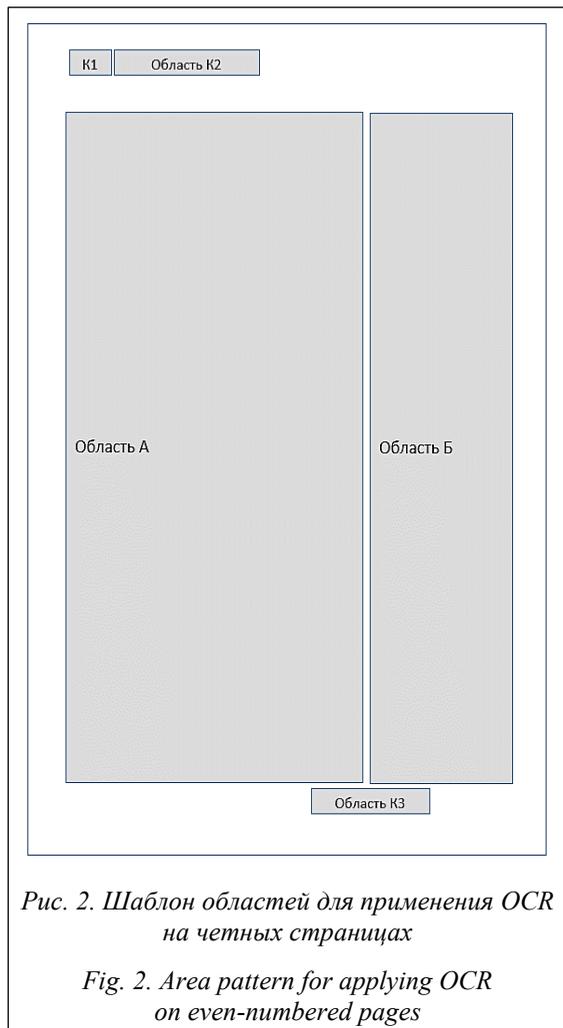


Рис. 2. Шаблон областей для применения OCR на четных страницах

Fig. 2. Area pattern for applying OCR on even-numbered pages

- месяц, день (например, Январь, 9).
- Спецификаторы и сложные структуры дат:
- спецификатор неточной датировки; любой элемент датировки может сопровождаться признаком неточной датировки – (?) (например, 1825 (?). Февраль, 1...3, Январь (?), Февраль, 1(?)...13(?), 1825(?). Февраль(?)...Май(?));
- слово «или» – два и более вариантов дат (например, Сентябрь, до 17 или Октябрь, после 14.);
- спецификаторы верхней/нижней границы дат – слова «не ранее, не позднее, до, конец, начало, после, первая половина» и т.п.; обозначают диапазон дней в месяце (например, Январь, после 18...Март, до 9, Ноябрь, перв. пол. (?));
- символ «/» – используется для разделения даты в григорианском и юлианском календарях (например, 1825. Декабрь, 21/1826. Январь, 2(?));
- символы «...», «-» – диапазон дат (например, 1829–1830 (?), Февраль, 1...10, Апрель, 4–5, Сентябрь, 25 (?)...Октябрь, 5 (?));

– спецификатор времени суток – утро, день, вечер, ночь (например, Январь, ночь с 27-го на 28-е).

Строки с датой могут содержать несколько значений, например:

- 1825. Декабрь, 21/1826. Январь, 2(?)...1826. Июнь, 24/Июль, 6(?);
- Август, перв. пол. ...Сентябрь, до 17 или Октябрь, после 14...Декабрь.

Для извлечения из текста записи Летописи строки, содержащей датировку, разработан набор регулярных выражений. Приведенные правила вывода строк достаточно хорошо покрываются возможностями регулярных выражений [12], благодаря чему удалось избежать разработки и применения специализированных грамматик.

При реализации алгоритмов выделения текста датировок возник ряд сопутствующих проблем, связанных с ошибками OCR:

- цифры: 1 – !, 3 – 3, 3 - } и т.п.;
- скобки: ( – <;
- знаки пунктуации: точки и запяты;
- месяцы: Сеитябрь.

Наиболее частые ошибки OCR были устранены путем их внесения в регулярные выражения для автоматической замены.

Ограничения на диапазоны с вариантами типовых ошибок распознавания:

- год Летописи: 1799–1836. r'1799|18[0-2][0-93з]|18[0-93з][0-7]';
- месяц: r'(?|Январь|Февраль|Март|Апрель|Май|Июнь|Июль|Август|Сентябрь|Октябрь|Ноябрь|Декабрь)';
- день месяца: 1 – 31. r'[0-9!\$}3з][!][0-9!\$Ф}3з]|2[0-9!\$Ф}3з][3}3з][0-1]'

Ограничения библиотеки Tesseract не позволяют обращаться к пользовательским словарям на этапе оптического распознавания, поэтому такие контекстные ошибки также пришлось учесть в регулярных выражениях.

### Реализация регулярных выражений

Реализовать механизм регулярных выражений для извлечения структурированных данных из текста позволяют язык программирования Python 3 и набор стандартных библиотек [15].

Особенности технологии реализации данного механизма отражены во фрагменте кода:

```
_regex_years_range = r'1799|18[0-2][0-93з]|18[0-93з][0-7] '
_regex_years_range_group =
```

```
f'(?:{_regex_years_range})'
_regex_months_range_group = r'(?:(Ян-
варь|Февраль|Март|Ап-
рель|Май|Июнь|Июль|Август|Сен-
тябрь|Октябрь|Ноябрь|Декабрь) '
_regex_day_in_month_range = r'
[0-9!$}Зз]|[1!][0-9!$Ф}Зз]
|2[0-9!$Ф}Зз]|[3}Зз][0-1!]'

# Октябрь, 22.
# Ноябрь. 28(?).
# Март, 2/14.
# Январь, после 18.
# Ноябрь, 28(?).
# Ноябрь (?), 28(?).
_regex_day_in_month_prefix_text =
r',|/|до|ок\.|ок\.\s*\ (не\s*позднее\ )
|по|се|ле|не\s*ран(?:е|ьш)е|не\s*позже'
_regex_month_part_group =
r'(?:(нач\.\.?|начало|конец|втор(?:ая|\.)\ )
\s*пол\.\.?|перв(?:ая|\.)\ )\s*пол(?:овина|\ )
?)|перв(?:ые|\.)\ )\s*чи|се|ла|(?:(ок\.\ |до)
?)\s*сер\.\.?|посл\.\ )\s*числа|не\s*позд-
нее)'
_regex_day_in_month_suffix_text = f'-
e\s*{[_regex_question_date_group]? \
\s*числа'
```

### Разработка методики оценки ошибок и построение автоматизированного конвейера обработки текстов

Для оценки качества работы алгоритмов оптического распознавания и алгоритмов выделения структурированного текста создана методика оценки ошибок и построен автоматизированный конвейер обработки текстов в сборочной системе GitLab [16]. Конвейер реализует следующие шаги.

1. На вход конвейера подаются наборы растровых изображений страниц томов Летописи.

2. На следующем этапе для каждого тома Летописи производится оптическое распознавание растровых изображений в текст файла в формате CSV (ocr\timeline\tesseract\T1out.csv, ocr\timeline\tesseract\T2out.csv, ocr\timeline\tesseract\T3out.csv, ocr\timeline\tesseract\T4out.csv, колонки: text, comment\_text, year\_place, page\_num\_global, page\_num\_local, year).

3. Далее из поля text с помощью регулярных выражений выделяется диапазон дат.

4. Обработанные записи сохраняются в файлах в формате CSV (T1.csv, T2.csv, T3.csv, T4.csv, колонки: page\_num, page\_num\_continuous, year, year\_place, timeline\_range, text, comment\_text).

5. Дополнительно вычисляется статистика обработки текста, которая сохраняется в файл статистики:

– подсчет общего количества записей Летописи в файле;

– подсчет количества записей с пустым полем диапазона дат;

– подсчет количества записей, где в поле текста Летописи встречаются дополнительные вхождения по регулярному выражению диапазона дат; эта проверка нужна для оценки качества структуризации текста Летописи и вычисления количества ошибочно объединенных записей;

– подсчет количества записей, где длина поля диапазона дат превышает заданный порог; необходим для оценки слишком длинных диапазонов дат, вероятно, с ошибочно выделенным текстом диапазона дат;

– запись в файл статистики выявленных аномальных записей.

6. Автоматически сравниваются результирующие файлы T1.csv, T2.csv, T3.csv, T4.csv и файлы статистики с результатами предыдущей успешной сборки конвейера, результаты сохраняются в файлы T1.diff, T2.diff, T3.diff, T4.diff, T1\_stat.diff, T2\_stat.diff, T3\_stat.diff, T4\_stat.diff.

Применение описанного автоматизированного конвейера позволяет контролировать влияние изменений в алгоритмах на всех этапах обработки на качество конечного результата обработки на корпусе входных данных (томов Летописи), добиваясь минимального количества ошибок в выходных данных.

Таким образом, разработчик-исследователь может принять решение о добавлении новых форматов в регулярные выражения, внесении изменений в процедуру коррекции промежуточных результатов оптического распознавания и повторном запуске конвейера, тем самым обеспечивая повышение скорости обработки Летописи и точности получаемых результатов.

### Выводы

В работе рассмотрена задача извлечения структурированных текстовых данных, пригодных для использования в рамках научно-просветительского ресурса, особенностью которой является необходимость использования управляемого оптического распознавания текста с последующей агрегацией полученных данных. Предложена гибридная схема, использующая одновременно результат оптического распознавания текста всего документа в целом, а также совокупность результатов распознавания фрагментов текста согласно шаблону страницы. Для извлечения структурированных дан-

ных из текста использован механизм регулярных выражений. Предложенная методика автоматизации позволяет в процессе работы оценить реальный процент ошибок, связанных с неправильным оптическим распознаванием символов. Например, при ошибке в распознавании месяца или года метод оценки не сможет обнаружить факт повторного вхождения диапазона дат в текст Летописи, что является признаком

неверного разделения ее записей. Кроме того, нет возможности автоматической оценки корректности сопоставления записи Летописи и текста комментария. Для оценки качества работы алгоритмов оптического распознавания и алгоритмов выделения структурированного текста разработана методика оценки ошибок и построен автоматизированный конвейер обработки текстов в сборочной системе GitLab.

#### Список литературы

1. Kassab K., Teslya N. An approach to a linked corpus creation for a literary heritage based on the extraction of entities from texts, *Applied Sci.*, 2024, vol. 14, no. 2, art. 585. doi: 10.3390/app14020585.
2. Тесля Н.Н., Жарков В.М. Структурирование библиотеки произведений А.С. Пушкина через создание базы данных для научно-просветительского портала «Пушкин цифровой» // Наука и технологии в трансформации социально-экономического ландшафта: сб. тр. Междунар. конф. TECHNOPERSECRIVE 2023. 2024. С. 168–171.
3. Тесля Н.Н., Сиповский Г.В. Разработка алгоритма сопоставления сущностей по описательным характеристикам их названий // Наука и технологии в трансформации социально-экономического ландшафта: сб. тр. Междунар. конф. TECHNOPERSECRIVE 2023. 2024. С. 127–134.
4. Летопись жизни и творчества А.С. Пушкина: в 4 т. М., 1999.
5. Tereshchenko V., Rybkin V., Shamis A., Yan D. Principles of handwriting recognition in FineReader. *Pattern Recognition and Image Analysis. Advances in Mathematical Theory and Applications*, 1998, vol. 8, no. 3, pp. 456–457. URL: <https://www.elibrary.ru/item.asp?id=16316347> (дата обращения 11.07.2024).
6. Tafti A.P., Baghaie A., Assefi M., Arabnia H.R., Yu Z., Peissig P. OCR as a service: An experimental evaluation of google docs OCR, Tesseract, ABBYY FineReader, and Transym. In: LNIP. Proc. ISVC, 2016, vol. 10072, pp. 735–746. doi: 10.1007/978-3-319-50835-1\_66.
7. Нейфельд О.А., Рудникович А.С. Возможности OCR-библиотеки Tesseract при распознавании текста по видеозаписям // Сб. избранных статей науч. сессии ТУСУР. 2019. № 1-2. С. 16–19.
8. Prelevikj M., Žitnik S. Multilingual named entity recognition and matching using BERT and Dedupe for Slavic languages. Proc. 8th Workshop on Balto-Slavic Natural Language Processing, 2021, pp. 80–85.
9. Беляева Л.Н., Ефремова А.Н. Автоматическая компиляция базы данных комплексного электронного словаря // Вестн. ВГУ. Сер.: Лингвистика и межкультурная коммуникация. 2015. № 3. С. 42–45.
10. Мальковский М.Г., Соловьев С.Ю. Правила формирования терминологических кластеров // Открытые семантические технологии проектирования интеллектуальных систем. 2014. № 4. С. 169–172.
11. Мальковский М.Г., Соловьев С.Ю. От терминологических сетей к толковым словарям // Открытые семантические технологии проектирования интеллектуальных систем. 2015. № 5. С. 281–284.
12. Чевтаев А.А. Формирование цифровых баз данных рукописей: проблемы и текстологические перспективы. Статья 1 // Новый филологический вестн. 2019. № 1. С. 28–43.
13. Coquenat D., Chatelain C., Paquet T. End-to-end handwritten paragraph text recognition using a vertical attention network. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023, vol. 45, no. 1, pp. 508–524. doi: 10.1109/tpami.2022.3144899.
14. Mohammed S., Teslya N. Handwritten paragraph recognition using spatial information on Russian notebooks dataset. Proc. FRUCT, 2023, pp. 108–113. doi: 10.23919/FRUCT60429.2023.10328173.
15. Фридл Дж. Регулярные выражения; [пер. с англ.]. СПб: Символ-Плюс, 2008. 608 с.
16. Преображенская Т.В., Симонов В.И. Автоматизированное тестирование в среде GitLab CI/CD // Флагман науки. 2023. № 4. С. 803–806.

#### Extracting structured data from the Chronicle of the Life and Work of A.S. Pushkin: Hybrid approach

Pavel P. Kokorin <sup>1</sup>, Aleksandr A. Kotov <sup>1</sup>, Sergey V. Kuleshov <sup>1</sup>, Aleksandra A. Zaytseva <sup>1</sup>✉

<sup>1</sup> St. Petersburg Federal Research Center of the Russian Academy of Sciences, St. Petersburg, 199178, Russian Federation

#### For citation

Kokorin, P.P., Kotov, A.A., Kuleshov, S.V., Zaytseva, A.A. (2025) 'Extracting structured data from the Chronicle of the Life and Work of A.S. Pushkin: Hybrid approach', *Software & Systems*, 38(1), pp. 39–46 (in Russ.). doi: 10.15827/0236-235X.149.039-046

**Article info**

Received: 12.07.2024

After revision: 13.08.2024

Accepted: 23.08.2024

**Abstract.** The paper discusses the problem of creating a software infrastructure for systematization, annotation, storage, search and publication of manuscripts and other digital materials. The research focuses on the materials related to the life and work of A.S. Pushkin. These materials form an important part of the scientific and educational resource "Pushkin Digital". The problem is relevant due to the need to preserve the Russian author's heritage under conditions of digital transformation of philological, source and bibliographic studies into their works. This is a part of the national projects of the Russian Federation "Education", "Culture", "Science and Universities". It is especially important to extract a structured text from bitmap images of pages from A.S. Pushkin's Chronicle of Life and Work volumes to use it in the developing systems of storage, systematization, publication of library, archival, museum, phonographic and other funds and collections and partial automation of philological, source and bibliographic research. The paper proposes a hybrid approach based on the a priori data about the structure of page layout elements, OCR technologies (text recognition based on Tesseract library) and verification methods. The peculiarity of the developed verification methods is using regular expressions for extracting structured data from pre-recognized text and automated text processing pipeline in the GitLab assembly system. The paper demonstrates satisfactory results of the proposed hybrid approach. The approach minimizes the manual post-processing of the obtained data by proofreading the results posted on the research and education resource. The results are useful not only in the research and educational resource Pushkin Digital under development, but also in other projects, which require recognition and automated processing of large volumes of digitized author's texts, archival and other paper documents.

**Keywords:** text recognition, data extraction, structured data, text processing

**Acknowledgements.** The work was carried out within the framework of implementing State Task no. FFZF-2023-0001

**References**

1. Kassab, K., Teslya, N. (2024) 'An approach to a linked corpus creation for a literary heritage based on the extraction of entities from texts', *Applied Sci.*, 14(2), art. 585. doi: 10.3390/app14020585.
2. Teslya, N.N., Zharkov, V.M. (2024) 'Structuring the library of works by A.S. Pushkin through the creation of a database for the scientific and educational portal "Pushkin Digital"', *Proc. Int. Conf. TECHNOSPESCRIVE 2023*, pp. 168–171 (in Russ.).
3. Teslya, N.N., Sipovsky, G.V. (2024) 'Development of an algorithm for matching entities based on the descriptive characteristics of their names', *Proc. Int. Conf. TECHNOSPESCRIVE 2023*, pp. 127–134 (in Russ.).
4. (1999) *Chronicle of the life and work of A.S. Pushkin: In 4 vol.* Moscow (in Russ.).
5. Tereshchenko, V., Rybkin, V., Shamis, A., Yan, D. (1998) 'Principles of handwriting recognition in FineReader', *Pattern Recognition and Image Analysis. Advances in Mathematical Theory and Applications*, vol. 8, no. 3, pp. 456–457, available at: <https://www.elibrary.ru/item.asp?id=16316347> (accessed July 11, 2024).
6. Tafti, A.P., Baghaie, A., Assefi, M., Arabnia, H.R., Yu, Z., Peissig, P. (2016) 'OCR as a service: An experimental evaluation of google docs OCR, Tesseract, ABBYY FineReader, and Transym', *LNIP. Proc. ISVC*, 10072, pp. 735–746. doi: 10.1007/978-3-319-50835-1\_66.
7. Neifeld, O.A., Rudnikovich, A.S. (2019) 'Capabilities of the Tesseract OCR library in text recognition from video images', *Proc. Collection of Selected Papers from the Scientific Session of TUSUR*, (1-2), pp. 16–19 (in Russ.).
8. Prelevikj, M., Žitnik, S. (2021) 'Multilingual named entity recognition and matching using BERT and Dedupe for Slavic languages', *Proc. 8th Workshop on Balto-Slavic Natural Language Processing*, pp. 80–85.
9. Belyaeva, L.N., Efremova, A.N. (2015) 'Automatic compilation of the database of a comprehensive electronic dictionary', *Proc. of VSU. Ser.: Linguistics and Intercultural Communication*, (3), pp. 42–45 (in Russ.).
10. Malkovsky, M.G., Soloviev, S.Yu. (2014) 'Rules for terminological clusters creations', *OSTIS*, (4), pp. 169–172 (in Russ.).
11. Malkovsky, M.G., Soloviev, S.Yu. (2015) 'From terminological networks to the explanatory dictionaries', *OSTIS*, (5), pp. 281–284 (in Russ.).
12. Chevtayev, A.A. (2019) 'Formation of digital databases of manuscripts: Problems and textological perspectives. Article 1', *The New Philological Bull.*, (1), pp. 28–43 (in Russ.).
13. Coquenot, D., Chatelain, C., Paquet, T. (2023) 'End-to-end handwritten paragraph text recognition using a vertical attention network', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(1), pp. 508–524. doi: 10.1109/tpami.2022.3144899.
14. Mohammed, S., Teslya, N. (2023) 'Handwritten paragraph recognition using spatial information on Russian notebooks dataset', *Proc. FRUCT*, pp. 108–113. doi: 10.23919/FRUCT60429.2023.10328173.

15. Friedl, J.E.F. (2006) *Mastering Regular Expressions*. O'Reilly Media Publ., 535 p. (Russ. ed.: (2008) St. Petersburg, 608 p.).
16. Preobrazhenskaya, T.V., Simonov, V.I. (2023) 'Automated testing in the GitLab CI/CD environment', *Flagship Science*, pp. 803–806 (in Russ.).

**Авторы**

**Кокорин Павел Петрович**<sup>1</sup>, к.т.н.,  
старший научный сотрудник,  
kokorin.p@iiias.spb.su

**Котов Александр Александрович**<sup>1</sup>,  
младший научный сотрудник,  
alexanderkotovspb@gmail.com

**Кулешов Сергей Викторович**<sup>1</sup>, д.т.н.,  
профессор РАН, главный научный сотрудник,  
kuleshov@iiias.spb.su

**Зайцева Александра Алексеевна**<sup>1</sup>,  
к.т.н., старший научный сотрудник,  
cher@iiias.spb.su

<sup>1</sup> Санкт-Петербургский федеральный  
исследовательский центр РАН,  
г. Санкт-Петербург, 199178, Россия

**Authors**

**Pavel P. Kokorin**<sup>1</sup>,  
Cand. of Sci. (Engineering),  
Senior Researcher, kokorin.p@iiias.spb.su

**Aleksandr A. Kotov**<sup>1</sup>,  
Junior Researcher,  
alexanderkotovspb@gmail.com

**Sergey V. Kuleshov**<sup>1</sup>,  
Dr.Sci. (Engineering), Professor RAS,  
Chief Researcher, kuleshov@iiias.spb.su

**Aleksandra A. Zaytseva**<sup>1</sup>,  
Cand. of Sci. (Engineering),  
Senior Researcher, cher@iiias.spb.su

<sup>1</sup> St. Petersburg Federal Research Center  
of the Russian Academy of Sciences,  
St. Petersburg, 199178, Russian Federation

## Моделирование надежности программных компонентов киберфизических систем

А.Н. Привалов<sup>1</sup>, Е.В. Ларкин<sup>2</sup>, А.В. Богомолов<sup>3</sup>✉

<sup>1</sup> Тульский государственный педагогический университет им. Л.Н. Толстого, г. Тула, 300026, Россия

<sup>2</sup> Тульский государственный университет, г. Тула, 300012, Россия

<sup>3</sup> Федеральный исследовательский центр «Информатика и управление» РАН, г. Москва, 119333, Россия

### Ссылка для цитирования

Привалов А.Н., Ларкин Е.В., Богомолов А.В. Моделирование надежности программных компонентов киберфизических систем // Программные продукты и системы. 2025. Т. 38. № 1. С. 47–54. doi: 10.15827/0236-235X.149.047-054

### Информация о статье

Группа специальностей ВАК: 2.3.5

Поступила в редакцию: 22.07.2024

После доработки: 16.08.2024

Принята к публикации: 23.08.2024

**Аннотация.** Актуальность данного исследования обусловлена тем, что надежность программных компонентов киберфизических систем является ключевой составляющей их эффективного функционирования. Ее адекватное математическое моделирование имеет существенное значение для прогресса цифровизации экономики. Работа направлена на устранение недостатков известных подходов к моделированию надежности программных компонентов, при реализации которых оценки характеристик надежности формируются на основе эмпирических данных о количестве ошибок, выявленных при тестировании программ. Поэтому результаты тестирования существенно зависят как от его длительности, так и от полноты покрытия области обрабатываемых данных подобластью данных, генерируемых при тестировании, что снижает эффективность оценивания надежности. Предметом исследования стали методы моделирования надежности программных компонентов киберфизических систем, характеризующейся временем запаздывания в цепи обратной связи между компонентами. В работе использованы методы программной инженерии, теории надежности, теории вероятностей и марковских процессов. Основным результатом являются математические модели надежности программных компонентов киберфизических систем, объединяющей полумарковские модели программных компонентов, генерации их сбоев и отказов. Разработанные математические модели основаны на структурно-параметрической полумарковской модели сбоев и отказов программного обеспечения, параметры которой определяются вычислительной сложностью и требованиями, предъявляемыми к программному обеспечению с учетом его функционального назначения. Получены формализованные описания пуассоновских потоков сбоев и отказов программных компонентов киберфизической системы. Практическая значимость работы определяется ее ориентированностью на применение для определения надежности программных компонентов на всех этапах жизненного цикла киберфизических систем, элементы которых взаимодействуют, самонастраиваются и адаптируются к изменениям с помощью стандартных программно реализуемых протоколов.

**Ключевые слова:** киберфизическая система, программное обеспечение, надежность программы, сбой программы, отказ программы, моделирование надежности, полумарковский процесс

**Благодарности.** Работа выполнена при поддержке гранта РФФИ № 24-29-20256

**Введение.** Ключевые достижения и тренды научно-технического прогресса связаны с цифровизацией, предполагающей повсеместное применение цифровых технологий сбора и обработки информации [1]. Ускоренное внедрение цифровых технологий в экономику и социальную сферу является одной из национальных целей развития России. Ее достижение необходимо в интересах создания условий для высокотехнологичного бизнеса, повышения конкурентоспособности страны на глобальном рынке, укрепления национальной безопасности и повышения качества жизни людей. Одним из приоритетов цифровизации является внедрение киберфизических систем, отличительная особенность которых в интеграции вычисли-

тельных ресурсов в инженерно-технические системы [2, 3]. Причем вычислительная компонента киберфизической системы распределена по всей системе и синергетически увязана с составляющими ее элементами [4, 5]. Элементы киберфизических систем и отдельные киберфизические системы взаимодействуют, самонастраиваются и адаптируются к изменениям с помощью стандартных программно реализуемых интернет-протоколов [6, 7]. Поэтому надежность программных компонентов киберфизических систем является ключевой составляющей их эффективного функционирования, а ее адекватное моделирование имеет существенное значение для прогресса цифровизации.

## Обзор существующих подходов

Причиной сбоев/отказов любого ПО являются ошибки, допущенные человеком, в том числе новые, возникшие при исправлении ранее сделанных ошибок и приводящие к отказам/сбоям того же самого или другого типа [8, 9]. Концепция моделирования надежности программных компонентов киберфизических систем должна быть единой, например, привязанной к их аппаратной части, а ввиду специфики причин отказов программная часть должна исследоваться в контексте моделей и задач обработки данных, сопровождающих функционирование киберфизических систем [10–12].

Недостатком известных подходов к моделированию надежности программных компонентов является то, что оценки характеристик надежности формируются на основе эмпирических данных о количестве ошибок, выявленных при тестировании программ [2, 13, 14]. Поэтому результаты тестирования существенно зависят как от его длительности, так и от полноты покрытия области обрабатываемых данных под областью данных, генерируемых при тестировании [12, 15].

Программные компоненты киберфизических систем решают задачи обеспечения их взаимодействия, самонастройки и адаптации к изменениям [16, 17]. В простейшем случае обработка данных по заданному алгоритму осуществляется последовательно, а целью обработки является формирование управляющих воздействий, направленных на обеспечение заданных характеристик: устойчивости, времени достижения установившегося режима функционирования киберфизической системы, перерегулирования, точности поддержания установившегося режима и т.п. [1, 18, 19]. Кратковременный выход перечисленных характеристик за заданные пределы означает сбой функционирования программных компонентов киберфизической системы, а устойчивое несоблюдение требований – их отказ.

Структурно-параметрической моделью функционирования программных компонентов киберфизической системы является полумарковский процесс [20, 21], состояния которого – абстрактный аналог операторов алгоритма функционирования. Время пребывания в состояниях определяется вычислительной сложностью реализуемых алгоритмов, а вероятность переключения в сопряженные состояния – спецификой обрабатываемых данных и логическими условиями выбора каждого состояния [22].

Таким образом, моделирование надежности программных компонентов киберфизической системы сводится к синтезу структурно-параметрической модели отказов ПО, применимой на этапе разработки программных компонентов. Недостаточно полная разработка моделирования надежности программных компонентов киберфизических систем обуславливает актуальность проведенных исследований, отличительные особенности которых определяются ориентацией на устранение недостатков известных подходов к моделированию надежности программных компонентов, при реализации которых оценки характеристик надежности формируются только на основе эмпирических данных о количестве ошибок, выявленных при тестировании программ.

## Метод исследования

Анализ опыта эксплуатации киберфизических систем показывает, что наиболее часто мониторинг состояния их элементов и компонентов реализуется с помощью циклического опроса (поллинга) [2, 22]. Поллинг реализует алгоритм, включающий множество операторов:

$$S = \{s_0, s_1, \dots, s_n, \dots, s_N, s_{N+1}\},$$

где  $s_0$  и  $s_{N+1}$  – фиктивные (неисполняемые) операторы, соответствующие операциям «Начало» и «Конец» запуска цикла поллинга, а остальные операторы  $s_n$  – исполняемые (иницирующие действия) операторы. Обратная связь между двумя неисполняемыми операторами придает цикличность алгоритму поллинга, а общее количество обращений ( $N$ ) к элементам и компонентам киберфизической системы в цикле поллинга определяется количеством связей ( $K$ ) между ними:

$$N = 2K.$$

Поскольку цикл поллинга реализует случайный процесс, эволюция (порядок обращений к элементам и компонентам киберфизической системы) которого после любого  $s_n$  не зависит от эволюции, предшествовавшей выполнению  $s_n$ , а время выполнения операции  $s_n$  имеет произвольные функции распределения, процедура поллинга может быть представлена в виде полумарковского процесса:

$$\mu = \{S, h(t)\},$$

где  $S$  – множество состояний, совпадающих с множеством операторов  $\{s_n\}$ ,  $n \in [1, \dots, N]$ ;  $h(t) = p \otimes f(t)$  – полумарковская матрица;  $f(t) = \{f_{j,n}(t)\}$  – матрица плотностей распределе-

ния времени выполнения операторов поллинга;  $p = \{p_{j,n}\}$  – стохастическая матрица.

При разорванной обратной связи (связи между  $s_0$  и  $s_{N+1}$ ) полумарковский процесс имеет одно стартовое ( $s_0$ ) и одно поглощающее ( $s_{N+1}$ ) состояния и будет эргодическим, только если для любой пары состояний  $(s_j, s_n), j, n \in [0, N+1]$ , существует хотя бы один путь из состояния  $s_j$  в состояние  $s_n$  и хотя бы один путь из состояния  $s_n$  в состояние  $s_j$ . Свойство эргодичности следует из требований к процедуре поллинга, которая в течение каждого цикла должна иметь возможность хотя бы по одному разу опросить каждый компонент киберфизической системы.

В общем случае плотности распределения  $f_{j,n}(t)$  время пребывания полумарковского процесса в состоянии  $s_j$ , если априорно известно, что следующим его состоянием будет  $s_n$ , определяется сверткой

$$f_{j,n}(t) = f_{j,n}^c(t) \odot f_{j,n}^d(t),$$

где  $f_{j,n}^c(t)$  – плотность распределения времени программной подготовки данных для обмена с  $j$ -м элементом киберфизической системы в процессе поллинга;  $f_{j,n}^d(t)$  – плотность распределения времени активации программного драйвера  $j$ -го элемента киберфизической системы в процессе поллинга;  $\odot$  – операция свертки.

Вероятности переключения полумарковского процесса из состояния  $s_j$  в состояние  $s_n$  определяются плотностями распределения данных, обрабатываемых конкретным оператором, и логическими условиями перехода процедуры поллинга в сопряженные операторы (алгоритмом поллинга).

В простейшем случае поллинг представляет собой последовательный однократный опрос каждого элемента киберфизической системы, реализуемый подачей на вход вектора входных сигналов с последующим анализом времени получения и компонентов вектора выходных сигналов. Кортеж  $\langle s_0, s_1, \dots, s_N, s_{N+1} \rangle$  образует на полном ориентированном графе процедуры поллинга гамильтонов (посещающий каждую вершину ровно один раз) путь из  $s_0$  в  $s_{N+1}$ . Одна из очевидных реализаций гамильтонова пути получается исключением из кортежа состояний  $s_0$  и  $s_{N+1}$ . Замыкание (связь) состояний  $s_1$  и  $s_N$  гамильтонова пути порождает гамильтонов цикл, в соответствии с которым производится опрос элементов киберфизической системы при поллинге.

Конкретная последовательность опроса элементов определяется разработчиком алгоритма поллинга с учетом специфики эксплуатации конкретной киберфизической системы. Плотности распределения времени пребывания процесса в состояниях  $s_0$  и  $s_{N+1}$  определяются вырожденным законом распределения:

$$f_{N+1,0}(t) = f_{0,j}(t) = \delta(t),$$

где  $\delta(t)$  –  $\delta$ -функция Дирака. Как и ранее, предполагается, что обработка данных производится в операторах  $\{s_i\}, i \in [1 \dots N]$ .

Плотность распределения времени возврата в любой из операторов  $\{s_1, \dots, s_N\}$ , образующих при замыкании цикл, равна

$$f_{s_j}(t) = L^{-1} \prod_{j=1}^N L(f_j(t)),$$

где  $L[\dots]$  и  $L^{-1}[\dots]$  – прямое и обратное преобразования Лапласа соответственно;  $f_j(t) \in f(t)$  – плотность распределения времени выполнения оператора  $s_j$ , выбираемая из матрицы  $f(t)$  в соответствии с построенным гамильтоновым циклом. Период цикла, характеризуемый оценкой математического ожидания времени возврата в начало после цикла поллинга, равен

$$T = \sum_{j=1}^N T_j = \sum_{j=1}^N \int_0^{\infty} t f_j(t) dt,$$

где  $T_j$  – математическое ожидание времени выполнения операции  $s_j, j \in [1 \dots N]$ .

Поскольку при опросе элементов киберфизической системы программа поллинга выполняет процедуру дискретизации входного сигнала, период  $T$  должен удовлетворять условию

$$\frac{2\pi}{T} > \Omega(k), k \in \{1, \dots, K\},$$

где  $\Omega(k)$  – минимальная круговая частота дискретизации  $k$ -го параметра, определяемая теоремой Котельникова для спектра вектора входных сигналов [23, 24]. На практике это условие, как правило, не выполняется, поэтому ошибка дискретизации при ее реализации с помощью процедуры поллинга должна быть меньше допустимой, то есть удовлетворять условию

$$\varepsilon \leq \min_{k \in \{1, \dots, K\}} \varepsilon_k,$$

где  $\varepsilon$  – допустимая ошибка дискретизации;  $\varepsilon_k$  – допустимая ошибка дискретизации  $k$ -го параметра.

При выходе периода дискретизации за порог, оцениваемый величиной  $\varepsilon_k$ , возникает ситуация, которая может привести к сбою/отказу киберфизической системы вследствие накопления ошибки дискретизации. Вероятность появления такой ошибки вычисляется как

$$p_k = \int_{\varepsilon}^{\infty} f_{S_j}(t) dt.$$

При организации обратной связи в киберфизической системе плотность распределения времени запаздывания вывода значений выходного (получаемого в результате поллинга) сигнала относительно ввода значений входного сигнала определяется выражением

$$f_z(t) = L^{-1} \left[ L \left[ f_{S_j}(t) \right]^M \right],$$

где  $M$  – порядок конечно-разностного уравнения в случае, если вектор выходного сигнала определяется как решение системы конечно-разностных уравнений, или количество отсчетов, необходимых для вычисления свертки, если вектор выходного сигнала определяется через вычисление свертки. Тогда математическое ожидание времени запаздывания будет

$$T_z = \int_0^{\infty} t f_z(t) dt.$$

Временной интервал между вводом входного сигнала и выводом выходного представляет собой чистое запаздывание в контуре обратной связи [25–27]. С точностью, достаточной для практических целей, можно считать, что запаздывание по времени во всех цепях обратной связи одинаково и определяется как временной интервал, начинающийся от чтения из буфера входного сигнала и заканчивающийся выводом выходного сигнала.

Если  $\tau_k^1, \tau_k^2$  – пороговые значения времени запаздывания в цепи обратной связи, при которых перерегулирование параметра входного сигнала достигает критических значений, а  $\tau_k^3$  – порог, при достижении которого киберфизическая система теряет устойчивость, то вероятность сбоя/отказа киберфизической системы определяется как

$$p_z = \int_{\tau}^{\infty} f_z(t) dt,$$

где

$$\tau = \min_{k \in \{1, \dots, K\}} \{ \tau_k^1, \tau_k^2, \tau_k^3 \}.$$

### Обсуждение результатов

Оценки вероятностей  $p_k$  и  $p_z$ , а также плотностей распределения  $f_k(t)$  и  $f_z(t)$  являются основой для построения полумарковской модели отказов программных компонентов киберфизической системы. На рисунке показан алгоритм определения надежности программных компонентов киберфизической системы, харак-

теризуемой временем запаздывания в цепи обратной связи.

По результатам подсчета надежности программных компонентов надежность киберфизической системы может быть оценена как высокая (ни одного выхода, то есть бессбойная работа программных компонентов), невысокая (от одного до  $N$  выходов, перемежающихся с периодами бессбойной работы, то есть сбойная работа программных компонентов) или низкая (больше  $N$  выходов, то есть отказ программных компонентов).

С помощью полумарковской модели времени наработки программных компонентов киберфизической системы до сбоя ( $T_s$ ) и до отказа ( $T_o$ ) могут быть оценены как

$$T_s = \frac{T_z}{p_z},$$

$$T_o = NT_z + \frac{T_z(1-p_z)(1-p_z^N) \sum_{i=1}^N i p_z^{i-1}}{p_z^N},$$

где  $T_z$  – математическое ожидание времени цикла поллинга;  $p_z$  – вероятность отказа программных компонентов в текущем цикле наблюдения.

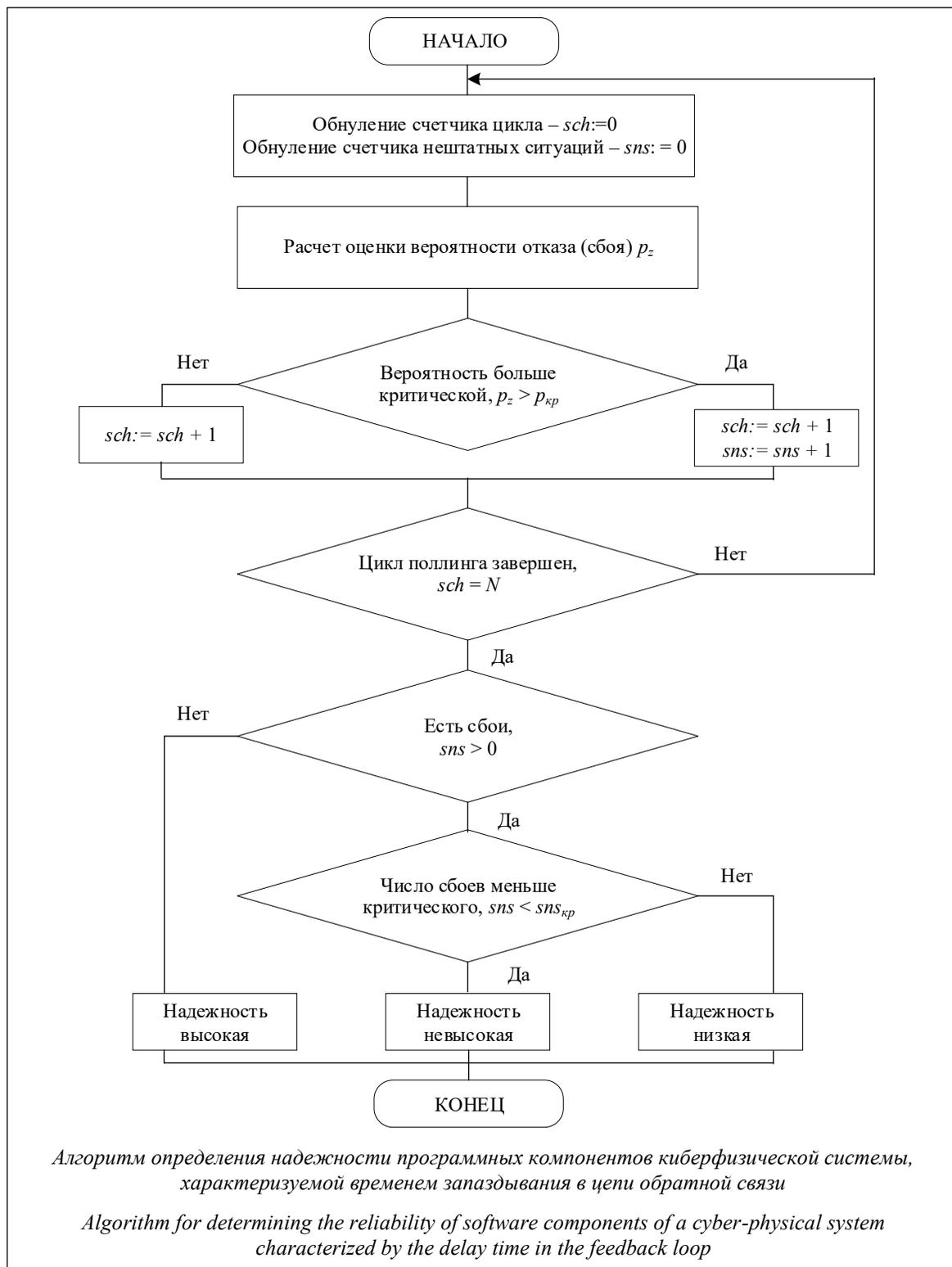
Поток выходов времени запаздывания формирования вектора выходных сигналов относительно вектора входных сигналов в цепи обратной связи за установленные пределы, представляющий собой сумму потоков с разными значениями времени запаздывания, стремится к пуассоновскому [28–30]. Плотность распределения времени между двумя выходами времени запаздывания за порог в пуассоновском потоке равна

$$f_{T_s}(t) = \frac{1}{T_s} e^{-\frac{t}{T_s}}, \quad f_{T_o}(t) = \frac{1}{T_o} e^{-\frac{t}{T_o}},$$

где  $f_{T_s}(t)$  и  $f_{T_o}(t)$  – распределения, описывающие пуассоновские потоки сбоев и отказов программных компонентов киберфизической системы.

### Выводы

В результате исследования разработан подход к моделированию надежности программных компонентов киберфизических систем. Надежность определяется сбоями и отказами, обусловленными некорректным учетом временной сложности реализуемых алгоритмов функционирования. Предлагаемый подход основан на применении структурно-параметрической полумарковской модели сбоев/отказов



ПО, параметры которой определяются вычислительной сложностью и требованиями, пред-

ъявляемыми к ПО с учетом его функционального назначения.

Список литературы

1. Логинов И.В., Федоров В.И. Управление функциональным развитием многофункциональных киберфизических систем // ИСУ. 2023. № 2. С. 108–120.

2. Ларкин Е.В., Богомолов А.В., Привалов А.Н., Акименко Т.А. Математическое обеспечение иерархического цифрового управления сложным технологическим объектом // *Вестн. ЮУрГУ. Сер. Математика. Механика. Физика.* 2024. Т. 16. № 4. С. 43–55. doi: 10.14529/mmp240406.
3. Данилюк А.Ю., Минаков Е.П., Привалов А.Е., Александров М.А. Аналитический метод оценивания эффективности управления киберфизическими системами с учетом характеристик их цифровых двойников // *Изв. РАН. 2023. № 4. С. 79–89.*
4. Обычайко Д.С., Шихин В.А. Анализ и принятие решений по эксплуатационной надежности киберфизических систем // *Вестн. МЭИ.* 2023. № 3. С. 154–162. doi: 10.24160/1993-6982-2023-3-154-162.
5. Shatnawi O. Testing-effort dependent software reliability model for distributed systems. *IJDST*, 2013, vol. 4, no. 2, pp. 1–14. doi: 10.4018/jdst.2013040101.
6. Aital P., Sashikala P. Role of software reliability models in performance improvement and management. *J. of Software Engineering and Applications*, 2021, vol. 5, no. 9, pp. 737–742. doi: 10.4236/jsea.2012.59086.
7. Barack O., Huang L. Assessment and prediction of software reliability in mobile applications. *J. of software engineering and applications*, 2020, vol. 13, no. 9, pp. 179–190. doi: 10.4236/jsea.2020.139012.
8. Лысов Д.В. Анализ надежности в жизненном цикле программного обеспечения // *СУИТ.* 2020. № 2. С. 76–82.
9. Солдатов Е.С., Богомолов А.В., Ларкин Е.В., Солдатов А.С. Сенсорные сети контроля состояния авиационной техники при испытаниях и эксплуатации // *Авиакосмическое приборостроение.* 2024. № 2. С. 61–68.
10. Соловьев С.Ю. Об одном методе генерации страниц-карт для веб-сайтов // *Информационные процессы.* 2008. Т. 8. № 1. С. 24–29.
11. Соловьев С.Ю., Стельмашенко Д.Е. Метод экспертной классификации как инструмент анализа формальных понятий // *ИИИПР.* 2013. № 4. С. 57–61.
12. Ларкин Е.В., Привалов А.Н., Богомолов А.В. Дискретный подход к моделированию синхронизированных эстафет // *НТИ. Сер. 2: Информ. процессы и системы.* 2020. № 2. С. 17–26.
13. Golosovskiy M., Tobin D., Balandov M., Khlopotov R. Architecture of software platform for testing software of cyber-physical systems. In: *LNNS. Proc. CoMeSySo*, 2023, vol. 597, pp. 488–494.
14. Тельнов Ю.Ф., Казаков В.А., Данилов А.В., Брызгалов А.А. Разработка моделей производственных и бизнес-процессов сетевых предприятий на основе многоагентных систем // *Программные продукты и системы.* 2023. Т. 37. № 4. С. 632–643. doi: 10.15827/0236-235X.144.632-643.
15. Голосовский М.С. Информационно-логическая модель процесса разработки программного обеспечения // *Программные системы и вычислительные методы.* 2015. Т. 28. № 1. С. 59–68. doi: 10.7256/2305-6061.2015.1.14119.
16. Голосовский М.С., Богомолов А.В., Тобин Д.С. Алгоритм настройки систем нечеткого логического вывода на основе статистических данных // *НТИ. Сер. 2: Информ. процессы и системы.* 2023. № 1. С. 1–9. doi: 10.36535/0548-0027-2023-01-1.
17. Уласень А.Ф., Стальнов А.Н., Зуева А.Н. Надежность функционального программного обеспечения информационно-управляющих систем реального времени // *Информационно-измерительные и управляющие системы.* 2022. Т. 20. № 6. С. 67–71. doi: 10.18127/j20700814-202206-08.
18. Мальковский М.Г., Соловьев С.Ю. Конструирование отношения псевдохронологического порядка в терминологических сетях // *Открытые семантические технологии проектирования интеллектуальных систем.* 2016. № 6. С. 183–186.
19. Ларкин Е.В., Солдатов Е.С., Богомолов А.В. Математическое обеспечение мониторинга состояния и управления режимами эксплуатации систем хранения криогенных продуктов // *Вестн. ЮУрГУ. Сер.: Математика. Механика. Физика.* 2024. Т. 16. № 1. С. 23–31.
20. Bielecki T.R., Jakubowski J., Niewęłowski M. Conditional Markov chains: Properties, construction and structured dependence. *Stochastic Processes and their Applications*, 2017, vol. 127, no. 4, pp. 1125–1170. doi: 10.1016/j.spa.2016.07.010.
21. Lu H., Pang G., Mandjes M. A functional central limit theorem for Markov additive arrival processes and its applications to queueing systems. *QUESTA*, 2016, vol. 84, no. 3, pp. 381–406.
22. Larkin E., Privalov A., Bogomolov A., Akimenko T. Digital control of continuous production with dry friction at actuators. In: *SIST. Proc. Electromechanics and Robotics*, 2022, vol. 232, pp. 427–436. doi: 10.1007/978-981-16-2814-6\_37.
23. Hayes M.H. *Statistical digital signal processing and modeling.* John Wiley & Sons Publ., NY, 1996, 624 p.
24. Stranneby D., Walker W. *Digital signal processing and applications.* Elsevier Publ., London, 2004, 357 p.
25. Briat C. Stability and performance analysis of linear positive systems with delays using input-output methods. *Int. J. of Control*, 2018, vol. 91, no. 7, pp. 1669–1692. doi: 10.1080/00207179.2017.1326628.
26. Li D., Chen G. Impulses-induced p-exponential input-to-state stability for a class of stochastic delayed partial differential equations. *Int. J. of Control*, 2019, vol. 92, no. 8, pp. 1805–1814. doi: 10.1080/00207179.2017.1412089.
27. Sanz R., García P., Albertos P., Fridman E. Robust predictive extended state observer for a class of nonlinear systems with time-varying input delay. *Int. J. of Control*, 2020, vol. 93, no. 2, pp. 217–222. doi: 10.1080/00207179.2018.1562204.
28. Григелионис Б.И. О сходимости сумм ступенчатых случайных процессов к пуассоновскому // *Теория вероятностей и ее применения.* 1963. Т. 8. № 2. С. 189–194.
29. Ларкин Е.В., Акименко Т.А., Богомолов А.В. Моделирование надежности бортового оборудования мобильного робота // *Изв. Сарат. ун-та. Нов. сер. Сер.: Математика. Механика. Информатика.* 2021. Т. 21. № 3. С. 390–399. (на англ.). doi: 10.18500/1816-9791-2021-21-3-390-399.
30. Kobayashi H., Marl B.L., Turin W. *Probability, random processes and statistical analysis.* Cambridge University Press, London, 2011, 812 p. doi: 10.1017/CBO9780511977770.

## Modeling the reliability of software components of cyber-physical systems

Aleksandr N. Privalov<sup>1</sup>, Evgeny V. Larkin<sup>2</sup>, Aleksey V. Bogomolov<sup>3</sup> ✉

<sup>1</sup> Tolstoy Tula State Pedagogical University, Tula, 300026, Russian Federation

<sup>2</sup> Tula State University, Tula, 300012, Russian Federation

<sup>3</sup> Federal Research Center “Computer Science and Control” of RAS, Moscow, 119333, Russian Federation

### For citation

Privalov, A.N., Larkin, E.V., Bogomolov, A.V. (2025) ‘Modeling the reliability of software components of cyber-physical systems’, *Software & Systems*, 38(1), pp. 47–54 (in Russ.). doi: 10.15827/0236-235X.149.047-054

### Article info

Received: 22.07.2024

After revision: 16.08.2024

Accepted: 23.08.2024

**Abstract.** This relevance of the research is due to the reliability of software components of cyber-physical systems being a key component of their effective functioning. Its appropriate mathematical modeling is essential for the economy digitalization. The paper aims to eliminate the disadvantages of known approaches to modeling the reliability of software components, when the estimates of reliability characteristics are based on empirical data on the errors detected during the testing of programs. Hence, the testing results mostly depend on test duration and on the completeness of the processed data area coverage by the sub-area of the data generated during testing. This reduces the efficiency of reliability estimation. The research focuses on the reliability modeling methods for software components of cyber-physical systems. The reliability is characterized by the delay time in the feedback loop between components. The authors of the paper used methods of software engineering, reliability theory, probability theory and Markov processes. The main result is mathematical reliability models of software components of cyber-physical systems. They combine semi-Markov models of software components, generations of their faults and failures. The developed mathematical models are based on the structural-parametric semi-Markov model of software faults and failures. Its parameters are determined by the computational complexity and requirements to the software taking into account its functional purpose. The authors obtained formalized descriptions of Poisson flows of faults and failures of software components of a cyber-physical system. The practical relevance of the paper is due to its application for determining the reliability of software components at all stages of a cyber-physical system life cycle, where elements interact, self-adjust and adapt to changes using standard software-implemented protocols.

**Keywords:** cyber-physical system, software, program reliability, program fault, program failure, reliability modeling, semi-Markov process

**Acknowledgements.** The work was supported by the Russian Science Foundation grant no. 24-29-20256

### References

1. Loginov, I.V., Fedorov, V.I. (2023) ‘Management of functional development of multifunctional cyber-physical systems’, *IS&CS*, (2), pp. 108–120 (in Russ.).
2. Larkin, E.V., Bogomolov, A.V., Privalov, A.N., Akimenko, T.A. (2024) ‘Mathematical support for hierarchical digital control of a complex technological object’, *Bull. of the SUrSU. Ser. Math. Mechanics. Physics*, 16(4), pp. 43–55 (in Russ.). doi: 10.14529/mmph240406.
3. Danilyuk, A.Yu., Minakov, E.P., Privalov, A.E., Aleksandrov, M.A. (2023) ‘Analytical method of evaluation of efficiency of control of cyberphysical systems taking into account characteristics of their digital counterparts’, *Bull. of RARAS*, (4), pp. 79–89 (in Russ.).
4. Obychaiko, D.S., Shikhin, V.A. (2023) ‘Analysis of and decision-making on the operational reliability of cyber-physical systems’, *Bull. of MPEI*, (3), pp. 154–162 (in Russ.). doi: 10.24160/1993-6982-2023-3-154-162.
5. Shatnawi, O. (2013) ‘Testing-effort dependent software reliability model for distributed systems’, *IJDST*, 4(2), pp. 1–14. doi: 10.4018/jdst.2013040101.
6. Aital, P., Sashikala, P. (2021) ‘Role of software reliability models in performance improvement and management’. *J. of Software Engineering and Applications*, 5(9), pp. 737–742. doi: 10.4236/jsea.2012.59086.
7. Barack, O., Huang, L. (2020) ‘Assessment and prediction of software reliability in mobile applications’, *J. of Software Engineering and Applications*, 13(9), pp. 179–190. doi: 10.4236/jsea.2020.139012.
8. Lysov, D.V. (2020) ‘Reliability analysis in the software life cycle’, *Control Systems and Inform. Tech.*, (2), pp. 76–82 (in Russ.).
9. Soldatov, E.S., Bogomolov, A.V., Larkin, E.V., Soldatov, A.S. (2024) ‘Sensor networks for monitoring the condition of aviation equipment during testing and operation’, *Aerospace Instrument-Making*, (2), pp. 61–68 (in Russ.).
10. Soloviev, S.Yu. (2008) ‘On one method of generating page maps for websites’, *Information Processes*, 8(1), pp. 24–29 (in Russ.).
11. Soloviev, S.Yu., Stelmashenko, D.E. (2013) ‘The expert classification method as a for formal concept analysis tool’, *AIDM*, (4), pp. 57–61 (in Russ.).
12. Larkin, E.V., Privalov, A.N., Bogomolov, A.V. (2020) ‘Discrete approach to modeling synchronized relay races’, *Sci. and Tech. Inform. Ser. 2: Inform. Processes and Systems*, (2), pp. 17–26 (in Russ.).
13. Golosovskiy, M., Tobin, D., Balandov, M., Khlopotov, R. (2023) ‘Architecture of software platform for testing software of cyber-physical systems’, in *LNNS. Proc. CoMeSySo*, 597, pp. 488–494.

14. Telnov, Yu.F., Kazakov, V.A., Danilov, A.V., Bryzgalov, A.A. (2023) 'Network enterprises: Production and business process models based on multi-agent systems', *Software & Systems*, 37(4), pp. 632–643 (in Russ.). doi: 10.15827/0236-235X.144.632-643.
15. Golosovsky, M.S. (2015) 'Information and logical model of the software development process', *Software Systems and Computational Methods*, 28(1), pp. 59–68 (in Russ.). doi: 10.7256/2305-6061.2015.1.14119.
16. Golosovsky, M.S., Bogomolov, A.V., Tobin, D.S. (2023) 'Algorithm for setting fuzzy logical inclusion systems based on statistical data', *Sci. and Tech. Inform. Ser. 2: Inform. Processes and Systems*, (1), pp. 1–9 (in Russ.). doi: 10.36535/0548-0027-2023-01-1.
17. Ulasen, A.F., Stalnov, A.N., Zueva, A.N. (2022) 'Ensuring the reliability of the functional software of informatively-managing systems real time', *Information-Measuring and Control Systems*, 20(6), pp. 67–71 (in Russ.).
18. Malkovsky, M.G., Soloviev, S.Yu. (2016) 'The designing of pseudo chronological order relation in the thermodynamical networks', *OSTIS*, (6), pp. 183–186 (in Russ.).
19. Larkin, E.V., Soldatov, E.S., Bogomolov, A.V. (2024) 'Mathematical support for monitoring the state and controlling the operating modes of cryogenic product storage systems', *Bull. of the SUrSU. Ser. Math. Mechanics. Physics*, 16(1), pp. 23–31 (in Russ.).
20. Bielecki, T.R., Jakubowski, J., Niewęglowski, M. (2017) 'Conditional Markov chains: Properties, construction and structured dependence', *Stochastic Processes and their Applications*, 127(4), pp. 1125–1170. doi: 10.1016/j.spa.2016.07.010.
21. Lu, H., Pang, G., Mandjes, M. (2016) 'A functional central limit theorem for Markov additive arrival processes and its applications to queuing systems', *QUESTA*, 84(3), pp. 381–406.
22. Larkin, E., Privalov, A., Bogomolov, A., Akimenko, T. (2022) 'Digital control of continuous production with dry friction at actuators', in *SIST. Proc. Electromechanics and Robotics*, 232, pp. 427–436. doi: 10.1007/978-981-16-2814-6\_37.
23. Hayes, M.H. (1996) *Statistical Digital Signal Processing and Modeling*, NY: John Wiley & Sons Publ., 624 p.
24. Stranneby, D., Walker, W. (2004) *Digital Signal Processing and Applications*. London: Elsevier Publ., 357 p.
25. Briat, C. (2018) 'Stability and performance analysis of linear positive systems with delays using input-output methods', *Int. J. of Control*, 91(7), pp. 1669–1692. doi: 10.1080/00207179.2017.1326628.
26. Li, D., Chen, G. (2019) 'Impulses-induced p-exponential input-to-state stability for a class of stochastic delayed partial differential equations', *Int. J. of Control*, 92(8), pp. 1805–1814. doi: 10.1080/00207179.2017.1412089.
27. Sanz, R., García, P., Albertos, P., Fridman, E. (2020) 'Robust predictive extended state observer for a class of nonlinear systems with time-varying input delay', *Int. J. of Control*, 93(2), pp. 217–222. doi: 10.1080/00207179.2018.1562204.
28. Grigelionis, B.I. (1963) 'On the convergence of sums of step random processes to a Poisson process', *Theory of Probability and its Applications*, 8(2), pp. 177–182. doi: 10.1137/1108017.
29. Larkin, E.V., Akimenko, T.A., Bogomolov, A.V. (2021) 'Modeling the reliability of the onboard equipment of a mobile robot', *Izv. Sarat. Univ. Math. Mech. Inform.*, 21(3), pp. 390–399. doi: 10.18500/1816-9791-2021-21-3-390-399.
30. Kobayashi, H., Marl, B.L., Turin, W. (2011) *Probability, Random Processes and Statistical Analysis*. London: Cambridge University Press, 812 p.

**Авторы**

**Привалов Александр Николаевич**<sup>1</sup>, д.т.н.,  
 профессор, директор института,  
 privalov.61@mail.ru  
**Ларкин Евгений Васильевич**<sup>2</sup>, д.т.н.,  
 профессор, elarkin@mail.ru  
**Богомолов Алексей Валерьевич**<sup>3</sup>, д.т.н.,  
 профессор, ведущий научный сотрудник,  
 a.v.bogomolov@gmail.com

**Authors**

**Aleksandr N. Privalov**<sup>1</sup>, Dr.Sci. (Engineering),  
 Professor, Director of the Institute,  
 privalov.61@mail.ru  
**Evgeny V. Larkin**<sup>2</sup>, Dr.Sci. (Engineering),  
 Professor, elarkin@mail.ru  
**Aleksey V. Bogomolov**<sup>3</sup>, Dr.Sci. (Engineering),  
 Professor, Leading Researcher,  
 a.v.bogomolov@gmail.com

<sup>1</sup> Тульский государственный педагогический университет им. Л.Н. Толстого,  
 г. Тула, 300026, Россия

<sup>2</sup> Тульский государственный университет,  
 г. Тула, 300012, Россия

<sup>3</sup> Федеральное исследовательское учреждение  
 «Информатика и управление» РАН,  
 г. Москва, 119333, Россия

<sup>1</sup> Tolstoy Tula State Pedagogical University,  
 Tula, 300026,  
 Russian Federation

<sup>2</sup> Tula State University, Tula,  
 300012, Russian Federation

<sup>3</sup> Federal Research Center  
 "Computer Science and Control" of RAS,  
 Moscow, 119333, Russian Federation

## Планирование вычислений в системах реального времени: эффективные алгоритмы построения оптимальных расписаний

Д.А. Кононов<sup>1</sup>, М.Г. Фуругян<sup>2</sup>✉

<sup>1</sup> Институт проблем управления им. В.А. Трапезникова РАН,  
г. Москва, 117997, Россия

<sup>2</sup> Федеральный исследовательский центр «Информатика и управление» РАН,  
г. Москва, 119333, Россия

### Ссылка для цитирования

Кононов Д.А., Фуругян М.Г. Планирование вычислений в системах реального времени: эффективные алгоритмы построения оптимальных расписаний // Программные продукты и системы. 2025. Т. 38. № 1. С. 55–64. doi: 10.15827/0236-235X.149.055-064

### Информация о статье

Группа специальностей ВАК: 1.2.3

Поступила в редакцию: 02.08.2024

После доработки: 13.08.2024

Принята к публикации: 23.08.2024

**Аннотация.** В статье рассматриваются вопросы, связанные с разработкой одного из основных блоков вычислительной системы реального времени – блока планирования вычислений. Предлагаются алгоритмы построения оптимальных расписаний для различных случаев в зависимости от числа процессоров и характеристик работ и ресурсов вычислительной системы. Для однопроцессорного случая с прерываниями и директивными интервалами усовершенствован алгоритм относительной срочности путем использования кучи для хранения данных. Это способствовало понижению вычислительной сложности алгоритма. Разработан алгоритм для задачи с частичным порядком выполнения работ, основанный на предварительной коррекции моментов готовности и директивных сроков и на сведении исходной задачи к задаче без отношений предшествования. Для многопроцессорного случая с прерываниями и директивными интервалами предложен приближенный алгоритм, основанный на обобщении однопроцессорного алгоритма относительной срочности на случай нескольких процессоров. Проведен сравнительный анализ с точным потоковым алгоритмом. Доказано, что в случае учета временных издержек на прерывания и переключения задача является NP-трудной. Для многопроцессорного случая без прерываний и переключений с общим директивным интервалом для всех работ и идентичными процессорами разработан псевдополиномиальный алгоритм, основанный на ограниченном переборе вариантов. Создан приближенный алгоритм для системы с возобновляемыми и невозобновляемыми ресурсами, а также для комплекса со смешанным набором работ (как непрерываемых, так и допускающих прерывания и переключения). Алгоритм основан на сетевом моделировании и сведении исследуемой задачи к поиску потока с определенными свойствами в специальной сети.

**Ключевые слова:** вычислительная система реального времени, однопроцессорная и многопроцессорная системы, потоковая сеть, допустимое расписание, возобновляемые и невозобновляемые ресурсы

**Введение.** Вычислительные системы реального времени стали широко внедряться в различные сферы деятельности человека с конца 70-х гг. прошлого века. Они находят применение в тех областях, где на проведение вычислений отводится строго ограниченное время. В частности, при проектировании и эксплуатации сложных технических объектов (самолеты, электростанции, атомные реакторы) это время может составлять доли секунды. В ряде случаев на вычисления может отводиться существенно большее время, например, при обработке информации экономического и экологического характера.

Известны два основных подхода к разработке вычислительных систем реального времени. Первый подход основан на предварительном расчете расписания выполнения вычислений. Так, авторами [1–3] разработана методика построения допустимых расписаний с дирек-

тивными сроками в многоядерной вычислительной системе реального времени. Согласно этой методике, сначала строится временная диаграмма работы системы, а затем с ее помощью проверяется выполнение каждого задания в своем директивном интервале. Кроме того, разработана имитационная модель системы, основанная на использовании сетей Петри и обобщенных конечных автоматов с остановкой таймера.

Авторы данной статьи в разное время принимали участие в проектах по созданию вычислительных систем реального времени: для обработки информации при проведении летных испытаний, для функционирования системы противоздушной и космической обороны, при разработке газовых месторождений и строительных работах. При этом также использовался подход, основанный на предварительном расчете расписания выполнения вычислений,

как это было сделано при разработке системы автоматизации программирования вычислительных систем реального времени, предназначенных для обработки циклически поступающей информации. На вход системы поступает программа пользователя, в которой описаны прикладные модули, предназначенные для вычисления, их длительности, входные и выходные параметры, частоты обращения к ним, длительности их выполнения и другие параметры. Блок генерации кодов переводит информацию, содержащуюся в этой программе, в таблицы, удобные для дальнейшей обработки. Далее работает генератор сетевой модели, который строит ориентированную сеть. Узлами сети являются прикладные модули, а дуги указывают отношения частичного порядка их выполнения. Затем подключается генератор расписаний, который, получая на вход эту сетевую модель, определяет, существует ли допустимое расписание выполнения прикладных модулей, и строит его в случае положительного ответа. Управляющая программа запускает в реальном времени прикладные модули согласно построенному расписанию. Одним из основных блоков этой системы является генератор расписаний.

При втором подходе планирование вычислений осуществляется в режиме реального времени. Так, в [4] исследована задача планирования комплекса работ, требования на выполнение которых поступают в известные моменты времени. Однако характеристики работ, такие как длительность и директивные сроки, становятся известными в момент поступления каждого запроса. В этом случае возникает необходимость составлять расписание выполнения вычислений в режиме реального времени. Авторами разработан полиномиальный алгоритм построения допустимого расписания для данного случая.

Таким образом, оба подхода к созданию вычислительных систем реального времени требуют эффективных алгоритмов составления расписаний. С появлением многопроцессорной и многоядерной вычислительной техники проблемы планирования вычислений стали еще более актуальными. Этому посвящено немало публикаций. Так, в [5] приводится большое число алгоритмов как для однопроцессорных, так и для многопроцессорных систем. В частности, впервые решена задача построения расписания для прерываемых работ с директивными интервалами в системе с идентичными процессорами путем сведения ее к потоковой

задаче. Рассмотрены задачи как с независимыми заданиями, так и с множеством работ с частичным порядком их выполнения. В [6] классифицированы задачи по теории расписаний, исследуются некоторые задачи дискретной оптимизации, метод ветвей и границ. В работах [5, 6] проведен анализ вычислительной сложности задач и алгоритмов.

В работах [7–9] рассматривается ряд экономических задач планирования. Предполагается, что некоторые параметры (длительность выполнения работ, объем ресурсов) не являются фиксированными, а могут принимать значения из заданных интервалов либо представляют собой случайные величины. Для решения этих задач авторы используют метод ветвей и границ. В [10] на основе понятия расстояния между задачами разработаны методы решения ряда NP-трудных задач для критерия минимизации максимального запаздывания, а также для задач на быстродействие.

В работе [11] рассмотрены системы реального времени с интегрированной модульной архитектурой. Авторы предлагают алгоритмы составления расписаний, основанные на построении транспортной сети и нахождении в ней максимального потока. Создание алгоритма решения задачи на быстродействие в многопроцессорной системе при ограничении на количество передач данных между процессорами описано в [12]. Алгоритм основан на использовании метода имитации отжига.

В [13] описана процедура построения расписания, состоящая из двух шагов. На первом шаге с помощью эвристического алгоритма, основанного на методе ветвей и границ, определяется последовательность выполнения работ. На втором в построенный график добавляются интервалы простоя с учетом директивных сроков начала и окончания операций. Эта задача сводится к задаче линейного программирования. В [14] рассмотрены проблемы, связанные с установлением директивных сроков выполнения работ и допустимых отклонений от этих сроков. Предложена модель, показывающая связь директивных сроков и стохастической изменчивости объемов ресурсов, необходимых для выполнения работ. В [15] решается задача оптимизации работы и маршрутов двух роботов, которые доставляют продукты в определенные места и должны вернуться в исходное положение. Рассмотрена задача на быстродействие и доказана ее NP-трудность. Для ее решения используются методы целочисленного линейного программирования, а также генети-

ческий алгоритм. Для оценки качества решений применяется динамическое программирование.

В настоящей статье предлагаются алгоритмы составления однопроцессорных и многопроцессорных расписаний с директивными сроками для работ, допускающих прерывания и переключения с одного процессора на другой, а также для непрерываемых работ. Особенность разработки в том, что исследованы задачи, совокупность работ которых включает не только прерываемые, но и непрерываемые задания, а комплекс ресурсов состоит из возобновляемых и невозобновляемых ресурсов. При этом показано, как выбор подходящей структуры данных позволяет сократить вычислительную сложность алгоритмов.

### Составление однопроцессорных и многопроцессорных расписаний с прерываниями и директивными сроками

- *Построение однопроцессорного расписания*

Имеется совокупность работ (заданий)  $W = \{w_1, w_2, \dots, w_n\}$ , которые должны быть выполнены с помощью  $m$  идентичных процессоров. Известны длительности  $t_i$  выполнения работ  $w_i$  и их директивные интервалы  $[b_i; f_i]$ ,  $t_i \leq f_i - b_i$  (работа  $w_i$  не может быть начата раньше момента ее готовности  $b_i$  и должна быть завершена не позднее ее директивного срока  $f_i$ ). При выполнении работ допускаются прерывания и переключения с одного процессора на другой. Предполагается, что они не требуют временных затрат. Не допускаются параллельное выполнение одного задания несколькими процессорами и одновременное выполнение нескольких работ одним процессором. Требуется определить, существует ли допустимое расписание выполнения работ, и найти его, если оно существует. Допустимое расписание – это такое расписание, при котором каждая работа выполняется в своем директивном интервале.

**Алгоритм относительной срочности.** Для однопроцессорных систем ( $m = 1$ ) известен алгоритм [16], который выглядит следующим образом.

Алгоритм 1.

Пусть  $A(t)$  – множество активных работ в момент времени  $t$ , то есть таких работ  $w_i$ , которые еще не завершены (или не начаты), и при этом  $t \in [b_i; f_i]$ . Процессор выделяется активной работе  $w_i$  с минимальным директивным сроком

$f : f_{i_0} = \min_{i: w_i \in A(t)} f_i$ . Если таких работ несколько, то выбирается любая из них. Работа  $w_i$  выполняется до тех пор, пока либо она не завершится, либо не появится активная работа  $w_{j_0}$  с меньшим директивным сроком  $f_{j_0}$ . В последнем случае выполнение работы  $w_{i_0}$  прерывается и процессор передается заданию  $w_{j_0}$ .

Поскольку выполнение работ может прерываться только в моменты времени  $b_i$ , число прерываний не более  $n$ . В отличие от [16] величины  $f_i$  для активных работ будем хранить в виде двоичной кучи с минимальным элементом в вершине. В этом случае вычислительная сложность алгоритма 1 составляет  $O(n \log n)$ . Алгоритм, описанный в [16], не использует двоичную кучу и имеет вычислительную сложность  $O(n^2)$ . В работе доказано, что данный алгоритм является корректным: если он не находит решение, то оно не существует.

**Построение однопроцессорного расписания с отношением частичного порядка на множестве работ.** Сформулированную постановку задачи для однопроцессорной системы дополним еще одним условием. Будем предполагать, что на множестве работ  $W$  задано отношение частичного порядка в виде ориентированного графа без циклов  $G = (W, A)$ , где  $A$  – множество ориентированных дуг. Если  $(w_i, w_j) \in A$ , значит, работа  $w_j$  может быть начата только после завершения работы  $w_i$ . Здесь  $w_i$  – непосредственный предшественник работы  $w_j$ , а  $w_j$  – непосредственный последователь работы  $w_i$ . Предлагаемый алгоритм основан на коррекции директивных интервалов с последующим применением алгоритма 1.

В основе коррекции директивных интервалов лежит следующее простое правило: если  $(w_i, w_j) \in A$ , то момент готовности  $b_j$  работы  $w_j$  и директивный срок  $f_i$  работы  $w_i$  пересчитываются по следующим формулам:

$$b_j = \max(b_j, b_i + t_i), \quad f_i = \min(f_i, f_j - t_j).$$

Действительно,  $b_i + t_i$  – это наиболее ранний возможный срок окончания работы  $w_i$ , а  $f_j - t_j$  – наиболее поздний допустимый срок завершения работы  $w_i$ . Перед выполнением процедуры коррекции множество  $W$  необходимо разбить на уровни следующим образом. Пусть  $W_0$  – это множество работ, не имеющих непосредственных предшественников. Если построены множества  $W_0, W_1, \dots, W_{k-1}$ , то  $W_k$  – это множество работ, имеющих непосредственных предшественников только в множествах  $W_0, W_1, \dots, W_{k-1}$ .

Предположим, что построено разбиение множества  $W$  на уровни  $W_0, W_1, \dots, W_p, p \leq n - 1$ . Коррекция моментов готовности работ проводится начиная с  $W_1$ , затем  $W_2$  и т.д. следующим образом: если  $w_j \in W_k$ , то принять

$$b_j = \max(b_j, \max_{i: (w_i, w_j) \in A} (b_i + t_i)), k = \overline{1, p}.$$

Коррекция директивных сроков работ проводится начиная с  $W_{p-1}$ , затем  $W_{p-2}$  и т.д. следующим образом: если  $w_i \in W_k$ , то принять

$$f_i = \min(f_i, \min_{j: (w_i, w_j) \in A} (f_j - t_j)), k = \overline{p-1, 0}.$$

Вычислительная сложность процедуры коррекции директивных интервалов составляет  $O(n^2)$ .

После выполнения процедуры коррекции множество  $W$  будет обладать следующим свойством: если  $(w_i, w_j) \in A$ , то  $b_i < b_j, f_i < f_j$ . Следовательно, для поиска допустимого расписания может быть использован алгоритм относительной срочности (алгоритм 1). Таким образом, алгоритм решения поставленной в настоящем разделе задачи следующий.

Алгоритм 2.

Шаг 1. Выполнить процедуру коррекции директивных интервалов.

Шаг 2. Применить алгоритм 1.

Вычислительная сложность алгоритма 2 составляет  $O(n^2)$ .

**Построение многопроцессорных расписаний.** Для случая  $m \geq 2$  может быть использован потоковый алгоритм [5] (алгоритм 3), основанный на сведении исходной задачи к нахождению максимального потока в ориентированной сети специального вида. Вычислительная сложность алгоритма 3 составляет  $O(n^3)$  (если при нахождении максимального потока использовать алгоритм кубической сложности). Алгоритм 3, как и алгоритм 1, является корректным.

Идея алгоритма относительной срочности в том, что процессор всегда занят активной работой с наименьшим директивным сроком и она легко может быть перенесена на многопроцессорный случай ( $m \geq 2$ ).

Алгоритм 4.

Процессоры всегда должны занимать  $m$  активных работ с наименьшими директивными сроками (если таковые имеются). Если активных работ меньше  $m$ , то все они должны выполняться. Если появляется новая активная работа с директивным сроком, меньшим наибольшего директивного срока из выполняемых работ, то она заменяет последнюю. В этом случае следует иметь две двоичные кучи. Первая содержит директивные сроки работ, которые

выполняются в данный момент (их не более  $m$ ). В вершине этой кучи находится максимальный элемент. Вторая куча содержит директивные сроки активных работ, которые в данный момент не выполняются. В вершине этой кучи находится минимальный элемент. Возникающая новая активная работа включается во вторую кучу. Если при этом она оказывается в вершине кучи и ее директивный срок меньше директивного срока, находящегося в вершине второй кучи, то элементы, находящиеся в вершинах обеих куч, меняются местами. После этого каждый из них становится на свое место в соответствующей куче. Вычислительная сложность алгоритма 4 составляет  $O((n \log n) \log m)$ . Алгоритм, описанный в [17], является аналогичным обобщением алгоритма 1 на многопроцессорный случай. Однако он не использует двоичные кучи и имеет более высокую вычислительную сложность:  $O(n(m \log m + \log n))$ .

Поскольку можно считать, что  $m \ll n$ , вычислительная сложность алгоритма 4 существенно меньше вычислительной сложности потокового алгоритма 3. Рассмотрим простой пример, показывающий работу алгоритма 4.

*Пример.* Пусть  $n = 3, m = 2, b_i = 0, f_i = 3, t_i = 2$  при всех  $i = 1, 2, 3$ . Пусть  $R_{ij}$  – временной интервал, в котором работа  $w_i$  выполняется  $j$ -м процессором. Тогда алгоритм 4 построит следующее расписание:  $R_{11} = [0; 2], R_{22} = [0; 2], R_{13} = [2; 4]$ . Это означает, что работа  $w_3$  не успевает завершиться к своему директивному сроку  $f_3 = 3$ , то есть алгоритм 4 не находит допустимого расписания. Однако на самом деле допустимое расписание существует:  $R_{11} = [0; 2], R_{22} = [0; 1], R_{21} = [2; 3], R_{13} = [1; 3]$ . В этом случае все работы успевают завершиться к своему директивному сроку  $f_i = 3, i = 1, 2, 3$ .

Следовательно, алгоритм 4 не всегда работает корректно. Как показано в [17], на основании результатов 25 000 численных экспериментов с рандомизированными и плавно варьируемыми переменными можно сделать вывод, что при больших размерностях задачи алгоритм 4 работает в тысячи раз быстрее алгоритма 3. В то же время некорректная работа алгоритма 4 была отмечена не более чем в 3 % случаев. Поэтому можно предложить следующий алгоритм решения задачи: сначала запускается быстрый алгоритм 4; если решение не найдено, запускается алгоритм 3.

*Замечание.* Если предположить, что прерывания и переключения с одного процессора на другой требуют временных затрат, то задача становится NP-трудной. Покажем, что в этом

случае к рассматриваемой задаче полиномиально сводится известная NP-полная задача о разбиении (имеются  $n$  натуральных чисел  $a_1, a_2, \dots, a_n$ , и пусть  $B = \sum_{i=1}^n a_i$  – четное число; можно ли разбить это множество на два непересекающихся подмножества с одинаковой суммой элементов, то есть существует ли такое подмножество  $\bar{N} \subseteq N = \{1, 2, \dots, n\}$ , что  $\sum_{i \in \bar{N}} a_i = \sum_{i \in N \setminus \bar{N}} a_i = B/2$ ) [18]. Если в рассматриваемой задаче о построении расписания принять  $m = 2, b_i = 0, f_i = B/2, i = \overline{1, n}$ , то допустимое расписание будет существовать в том и только том случае, когда в задаче о разбиении ответ положительный.

### Составление многопроцессорных расписаний без прерываний с общим директивным интервалом

Предположим, что работы не допускают прерываний и переключений с одного процессора на другой, длительности  $t_i$  принимают натуральные значения, а директивные интервалы всех работ совпадают:  $b_i = 0, f_i = F, t_i \leq F$ . Известно, что такая задача является NP-трудной в сильном смысле [18]. Однако при фиксированном числе процессоров  $m$  существует псевдополиномиальный алгоритм.

Пусть  $X = \{x = (x_1, x_2, \dots, x_m): 0 \leq x_j \leq F, j = \overline{1, m}\}$ . Здесь  $x_j$  – временная загруженность  $j$ -го процессора,  $j = \overline{1, m}$ . Пусть  $X_k \subseteq X, k = \overline{1, n}$  – непересекающиеся подмножества множества  $X$ , которые определяются следующим образом:

$X_1 = \{x \in X: x_i = 0 \text{ при } i \neq j, x_j = t_j, j = \overline{1, m}\}$ , то есть  $X_1$  – множество векторов из  $X$ , у которых все компоненты, кроме  $x_j$ , равны 0, а  $x_j = t_j, j = \overline{1, m}$ ;

$X_k = \{x \in X: x_j = \bar{x}_j + t_k, x_j \leq F, j = \overline{1, m}, \bar{x} \in X_{k-1}\}$ ,  $k = \overline{2, n}$ , то есть каждый вектор из  $X_k$  получается прибавлением величины  $t_k$  к  $j$ -й компоненте некоторого вектора  $\bar{x} \in X_{k-1}, j = \overline{1, m}$ . Каждому вектору  $x \in X_k$  приписывается метка, указывающая, из какого вектора  $\bar{x} \in X_{k-1}$  он получен.

Важно отметить: если при построении множества  $X_k$  для некоторой компоненты  $x_j$  вектора  $x \in X_k$  выполняется неравенство  $x_j > F$ , то вектор  $x$  не включается в  $X_k$ . К тому же число векторов в  $X_k$  не превосходит  $(F + 1)^m$ . Алгоритм

решения задачи (алгоритм 5) заключается в построении множеств  $X_k, k = \overline{1, n}$ . Если  $X_n \neq \emptyset$ , то допустимое расписание существует и может быть построено путем следования по меткам, указанным при построении множеств  $X_k, k = \overline{2, n}$ .

Алгоритм 5.

Шаг 1. Построить множества  $X_k, k = \overline{1, n}$ .

Шаг 2. Если  $X_n \neq \emptyset$ , перейти на шаг 3. В противном случае перейти на шаг 5.

Шаг 3. С помощью меток, указанных при построении множеств  $X_k, k = \overline{2, n}$ , построить последовательность векторов  $x^k \in X_k, k = \overline{n, 1}$ .

Шаг 4. Если  $x^k$  и  $x^{k-1}$  отличаются  $j$ -й компонентой, то работа  $w_k$  приписывается процессору  $j, k = \overline{n, 1}$ . Завершение алгоритма.

Шаг 5. Допустимого расписания не существует. Завершение алгоритма.

Далее работы, приписанные процессору  $j, j = \overline{1, m}$ , выполняются этим процессором в произвольном порядке. Поскольку число элементов в каждом множестве  $X_k, k = \overline{1, n-1}$ , не превосходит число элементов в  $X$ , то есть величины  $(F + 1)^m$ , и для каждого из них строится не более  $m$  элементов в  $X_{k+1}$ , то вычислительная сложность алгоритма 5 составляет  $O(mn(F + 1)^m)$ , или  $O(mnF^m)$ . Следовательно, при фиксированном числе процессоров  $m$  алгоритм 5 является псевдополиномиальным.

### Составление расписаний в системе с неоднородными ресурсами

В работе исследовались задачи с возобновляемыми ресурсами. Это ресурсы, которые могут использоваться многократно, например, процессоры, приборы, машины и т.д. Кроме того, все работы либо допускали прерывания и переключения с одного процессора на другой, либо были непрерываемыми. Рассмотрим задачу планирования работ в системе, где, помимо процессоров, имеются невозобновляемые ресурсы. Эти ресурсы не могут использоваться повторно, например, электроэнергия, финансы, дополнительная память или специализированные устройства, выделяемые конкретному программному модулю и не используемые другими модулями. Кроме того, исследуем задачу, в которой часть работ допускают прерывания и переключения с одного процессора на другой, а часть не допускают.

- *Смешанный комплекс работ, общий директивный интервал*

Рассмотрим комплекс работ  $W = W_1 \cup W_2$ , где  $W_1 = \{w_{11}, w_{12}, \dots, w_{1n_1}\}$  – работы, допускающие прерывания и переключения с одного процессора на другой,  $W_2 = \{w_{21}, w_{22}, \dots, w_{2n_2}\}$  – непрерываемые работы. Для работ  $W_1$  и  $W_2$  остаются предположения, сделанные ранее. Для всех работ установлен общий директивный интервал  $[0; F]$ . Для выполнения работ имеются  $m$  идентичных процессоров и  $L$  типов невозобновляемых ресурсов, объемы которых составляют  $R_1, R_2, \dots, R_L$ . Если работе  $w_{ki}$  выделен ресурс  $l$ -го типа в количестве  $r_{kil}$ ,  $l = \overline{1, L}$ , то длительность ее выполнения составляет

$$t_{ki} = t_{ki}^0 - \sum_{l=1}^L a_{kil} r_{kil}, \quad k = 1, 2, i = \overline{1, n_k}, \quad (1)$$

где  $t_{ki}^0$  – длительность выполнения работы  $w_{ki}$  в случае, если невозобновляемые ресурсы ей не выделяются;  $t_{ki}^0 \leq F$ ,  $k = 1, 2, i = \overline{1, n_k}$ ,  $a_{kil} > 0$  – заданные величины. На величины  $r_{kil}$  накладываются следующие ограничения:

$$r_{kil}^1 \leq r_{kil}^2, \quad k = 1, 2, i = \overline{1, n_k}, l = \overline{1, L}, \quad (2)$$

$$\sum_{k=1}^2 \sum_{i=1}^{n_k} r_{kil} \leq R_l, \quad l = \overline{1, L}, \quad (3)$$

$$t_{ki}^0 - \sum_{l=1}^L a_{kil} r_{kil}^2 > 0, \quad k = 1, 2, i = \overline{1, n_k}. \quad (4)$$

Неравенство (2) определяет ограничения на объемы ресурсов, выделяемых каждой работе. Неравенство (3) ограничивает общий объем выделяемых ресурсов каждого типа. Согласно неравенству (4), при выделении работе максимально допустимого объема ресурсов каждого типа ее длительность остается положительной. Распределение ресурсов  $r_{kil}$ ,  $k = 1, 2, i = \overline{1, n_k}$ ,  $l = \overline{1, L}$ , удовлетворяющее ограничениям (2)–(4), будем называть допустимым. Задача заключается в поиске допустимого распределения ресурсов и допустимого расписания.

Алгоритм 6 решения задачи состоит из трех этапов. На первом этапе выполняется распределение невозобновляемых ресурсов, после чего будут определены длительности выполнения работ. На втором этапе процессоры делятся на две группы: первая – для выполнения работ  $W_1$ , вторая – для выполнения работ  $W_2$ . На третьем этапе строится расписание, отдельное для  $W_1$  и  $W_2$ .

Для каждого  $l = \overline{1, L}$  ресурс  $l$ -го типа распределяется следующим образом. Определяется

$\max_{k,i} a_{kil} = a_{k_0 i_0}$ , и работе  $w_{k_0 i_0}$  выделяется максимально возможное количество ресурса  $l$ -го типа (то есть такое количество, при котором не нарушаются ограничения (2), (3)). Оставшаяся неиспользованная часть ресурса  $l$ -го типа распределяется среди оставшихся работ по аналогичному правилу. Такое распределение ресурсов позволяет максимально сократить длительности работ. Если для каждого  $l = \overline{1, L}$  величины  $a_{kil}$  отсортировать по невозрастанию, то вычислительная сложность первого этапа будет составлять  $O(Ln \log n)$ .

После распределения невозобновляемого ресурса между работами вычисляются их длительности по (1) и величины

$$m_1 = \left\lfloor \frac{\sum_{i=1}^{n_1} t_{1i}}{\sum_{i=1}^{n_1} t_{1i} + \sum_{i=1}^{n_2} t_{2i}} \right\rfloor, \quad m_2 = m - m_1. \quad (5)$$

Работы  $W_1$  будут выполняться первыми  $m_1$  процессорами, а работы  $W_2$  – оставшимися  $m_2$  процессорами. По формуле (5) распределяются процессоры между работами  $W_1$  и  $W_2$  пропорционально суммарной длительности выполнения работ этих множеств. Вычислительная сложность второго этапа составляет  $O(n)$ , где  $n = n_1 + n_2$ .

Для построения расписания выполнения работ  $W_1$  и  $W_2$  соответственно могут быть использованы алгоритм упаковки [5], вычислительная сложность которого составляет  $O(n_1)$ , и рассмотренный псевдополиномиальный алгоритм, вычислительная сложность которого составляет  $O(m_2 n_2 F^{m_2})$ .

- *Прерываемые работы, произвольные директивные интервалы*

Предположим, что множество  $W$  однородное: каждая работа  $w_i \in W$  допускает прерывания и переключения с одного процессора на другой и имеет директивный интервал  $[b_i; f_i]$ . Для ресурса  $l$ -го типа известен коэффициент  $a_l$ , показывающий, на какую величину сократится время выполнения работы, если ей будет выделена единица этого ресурса. К тому же выполнены приведенные далее ограничения, аналогичные условиям (1)–(4), а именно: если работе  $w_i$  выделено  $r_{il}$  ресурсов  $l$ -го типа, то справедливы соотношения:

$$t_i = t_i^0 - \sum_{l=1}^L a_l r_{il}, \quad i = \overline{1, n}, \quad (6)$$

$$r_{il} \leq r_{il}^2, \quad i = \overline{1, n}, \quad l = \overline{1, L}, \quad (7)$$

$$\sum_{i=1}^n r_{il} \leq R_l, l = \overline{1, L}, \quad (8)$$

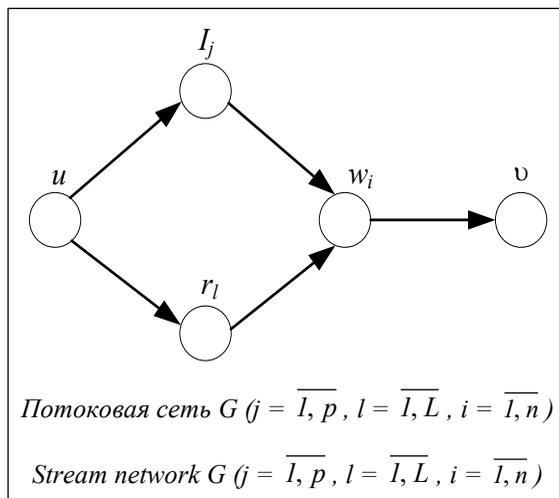
$$t_i^0 - \sum_{l=1}^L a_l r_{il}^2 > 0, i = \overline{1, n}. \quad (9)$$

Величины  $t_i^0$  и  $r_{il}^2$  определяются по аналогии с тем, как это сделано для смешанного комплекса работ.

Для решения задачи по примеру [5] построим потоковую сеть  $G = (N, A)$ , дополнив ее элементами, соответствующими невозобновляемым ресурсам (см. рисунок). Множество узлов определим как  $N = \{u, I_j, r_l, w_i, v, j = \overline{1, p}, l = \overline{1, L}, i = \overline{1, n}\}$ , где  $I_j = [y_{j-1}; y_j], y_0 < y_1 < \dots < y_p$  – все различные величины  $b_i, f_i; i = \overline{1, n}; u$  – источник;  $v$  – сток;  $r_l$  соответствует ресурсу  $l$ -го типа. Множество дуг  $A$  определим как  $A = \{(u, I_j), (I_j, w_i), (r_l, w_i), (w_i, v), j = \overline{1, p}, l = \overline{1, L}, i = \overline{1, n}\}$ . Дуга  $(I_j, w_i)$  вводится в сеть  $G$  в том случае, если  $I_j \subseteq [b_i; f_i]$ .

Пропускные способности  $U$  дуг определим следующим образом:  $U(u, I_j) = m(y_{j+1} - y_j), U(u, r_l) = R_l, U(I_j, w_i) = y_{j+1} - y_j, U(r_l, w_i) = a_l r_{il}^2, U(w_i, v) = t_i, j = \overline{1, p}, l = \overline{1, L}, i = \overline{1, n}$ . В отличие от [5] будем рассматривать обобщенные сети, в некоторых внутренних узлах которых величина выходящего потока равна величине входящего потока, умноженной на некоторый коэффициент (коэффициент выигрыша), постоянный для данного узла. Так, для рассматриваемой сети  $G$  в узлах  $I_j$  и  $w_i$  выполняется условие сохранения потока, а в узле  $r_l$  коэффициент выигрыша равен  $a_l$ .

По аналогии с тем, как это сделано в [5], можно показать, что решение задачи существует только в том случае, когда максималь-



ный поток  $g$  в сети  $G$  насыщает все дуги  $(w_i, v)$ , то есть

$$(w_i, v) = t_i \quad (10)$$

при всех  $i = \overline{1, n}$ . Если это условие выполнено, то потоки по дугам определяют расписание выполнения работ  $W$  процессорами [5]. Потоки по дугам  $(r_l, w_i)$  определяют распределение ресурса  $l$ -го типа:

$$r_{il} = g(r_l, w_i)/a_l, l = \overline{1, L}, i = \overline{1, n}. \quad (11)$$

Таким образом, для решения рассматриваемой задачи разработан следующий алгоритм.

Алгоритм 7.

Шаг 1. Построить сеть  $G$ .

Шаг 2. Найти максимальный поток  $g$  в сети  $G$ .

Шаг 3. Если выполнено условие (10), решение существует; перейти на шаг 4, в противном случае – на шаг 5.

Шаг 4. Расписание выполнения работ  $W$  определяется с помощью величин  $g(I_j, w_i)$  и алгоритма упаковки [5]. Распределение невозобновляемых ресурсов определяется по формуле (11). Алгоритм завершен.

Шаг 5. Решения не существует. Алгоритм завершен.

Если при нахождении максимального потока в сети  $G$  использован алгоритм кубической сложности, то вычислительная сложность алгоритма 7 составляет  $O((n + L)^3)$ , поскольку число узлов в сети  $G$  равно  $O(n + L)$ . Отметим, что в [19] предполагается, что длительность выполнения каждой работы выражается убывающей по каждой переменной функцией от количества предоставленных этой работе невозобновляемых ресурсов. Эта задача сводится к минимизации некоторой функции с  $O(n(n + L))$  переменными при  $O(n^2)$  линейных ограничений. Предположение о линейности длительностей от выделенных ресурсов позволило свести задачу к потоковой.

### Заключение

В статье представлены разработанные алгоритмы планирования вычислений в системах реального времени. Для работ, допускающих прерывания и переключения с одного процессора на другой, предложены более быстрые алгоритмы по сравнению с известными за счет использования подходящей структуры данных. Доказана NP-трудность задачи в случае, когда учитываются временные издержки на обработку прерываний и переключений. Для случая с отношениями предшествования предложен алгоритм, основанный на коррекции директив-

ных интервалов и использовании алгоритма относительной срочности. Для непрерываемых работ разработан псевдополиномиальный алгоритм, основанный на ограниченном переборе. Исследована задача распределения смешанного комплекса ресурсов (возобновляемых и невозобновляемых). Для случая общего директивного интервала и смешанного набора работ (прерываемых и непрерываемых) разра-

ботан псевдополиномиальный алгоритм. Для случая прерываемых работ и произвольных директивных интервалов разработан полиномиальный алгоритм, основанный на сведении исходной задачи к потоковой.

В дальнейшем авторы планируют исследовать обобщения рассмотренных задач на случаи наличия неопределенных факторов и вероятностных характеристик параметров.

#### Список литературы

1. Глонина А.Б. Обобщенная модель функционирования модульных вычислительных систем реального времени для проверки допустимости конфигураций таких систем // Вестн. ЮУрГУ. Сер. Вычисл. математика и информатика. 2017. Т. 6. № 4. С. 43–59. doi: 10.14529/cmse170404.
2. Глонина А.Б., Балашов В.В. О корректности моделирования модульных вычислительных систем реального времени с помощью сетей временных автоматов // МАИС. 2018. Т. 25. № 2. С. 174–192. doi: 10.18255/1818-1015-2018-2-174-192.
3. Глонина А.Б. Инструментальная система проверки выполнения ограничений реального времени для конфигураций модульных вычислительных систем // Вестн. МГУ. Сер. 15. Вычисл. матем. и киберн. 2020. № 3. С. 16–29.
4. Кононов Д.А., Фуругян М.Г. Региональное управление: оперативное планирование в режиме реального времени // Управление развитием крупномасштабных систем: тр. Междунар. конф. 2023. № 1. С. 1138–1144.
5. Танаев В.С., Гордон В.С., Шафранский Я.М. Теория расписаний. Одностадийные системы. М.: Наука, 1984. 383 с.
6. Brucker P. Scheduling Algorithms. Springer Publ., 2007, 383 p.
7. Кошелев П.С., Мищенко А.В. Оптимизация управления работами логистического проекта в условиях неопределенности // Изв. РАН. ТиСУ. 2021. № 4. С. 123–134. doi: 10.31857/S0002338821040089.
8. Горский М.А., Мищенко А.В., Нестерович Л.Г., Халиков М.А. Некоторые модификации целочисленных оптимизационных задач с учетом неопределенности и риска // Изв. РАН. ТиСУ. 2022. Т. 6. № 6. С. 65–76. doi: 10.31857/S0002338822050079.
9. Косоруков О.А., Лемтюжников Д.В., Мищенко А.В. Методы и модели управления ресурсами проекта в условиях неопределенности // Изв. РАН. ТиСУ. 2023. № 3. С. 38–56. doi: 10.31857/S0002338823020117.
10. Лазарев А.А. Теория расписаний. Методы и алгоритмы. М.: ИПУ РАН, 2019. 408 с.
11. Костенко В.А., Смирнов А.С. Поточковые алгоритмы планирования вычислений в интегрированной модульной авионике // Изв. РАН. ТиСУ. 2019. № 3. С. 77–86. doi: 10.1134/S0002338819030119.
12. Балашов В.В., Костенко В.А., Федоренко И.А., Гао Ц., Сун Ч.М., Сун Ц. Алгоритм имитации отжига для построения списочных расписаний с ограничениями на количество межпроцессорных передач данных // Автоматика и телемеханика. 2023. № 8. С. 138–152. doi: 10.31857/S0005231023080093.
13. Gorman M.F., Conway D.G. A tutorial of integrating duality and branch and bound in earliness–tardiness scheduling with idle insertion time problems. IJPR, 2018, vol. 56, no. 1-2, pp. 262–277. doi: 10.1080/00207543.2017.1397794.
14. Graves S.C. How to think about planned lead times. SSRN Electronic J., 2022. doi: 10.2139/ssrn.3485059. URL: [https://papers.ssrn.com/sol3/papers.cfm?abstract\\_id=3485059](https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3485059) (дата обращения: 10.07.2024).
15. Thomasson O., Battarra M., Erdoğan G., Laporte G. Scheduling twin robots in a palletising problem. IJPR, 2018, vol. 56, no. 1-2, pp. 518–542. doi: 10.1080/00207543.2017.1401249.
16. Коффман Э.Г. Теория расписаний и вычислительные машины; [пер. с англ.]. М.: Наука, 1984. 336 с.
17. Фуругян М.Г. Некоторые алгоритмы анализа и синтеза многопроцессорных вычислительных систем реального времени // Программирование. 2014. № 1. С. 36–44.
18. Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К. Алгоритмы. Построение и анализ; [пер. с англ.]. М.: Вильямс, 2011. 1296 с.
19. Kononov D.A., Furugyan M.G. Distribution of heterogeneous complex of resources on a multiprocessor system. Proc. Int. Conf. SUMMA, 2023, pp. 368–372. doi: 10.1109/SUMMA60232.2023.10349570.

#### Planning computations in real-time systems: Efficient algorithms for constructing optimal schedules

Dmitry A. Kononov <sup>1</sup>, Meran G. Furugyan <sup>2</sup>✉

<sup>1</sup> Trapeznikov Institute of Control Sciences of the Russian Academy of Sciences, Moscow, 117997, Russian Federation

<sup>2</sup> Federal Research Center “Computer Science and Control” of the Russian Academy of Sciences, Moscow, 119333, Russian Federation

**For citation**

Kononov, D.A., Furugyan, M.G. (2025) 'Planning computations in real-time systems: Efficient algorithms for constructing optimal schedules', *Software & Systems*, 38(1), pp. 55–64 (in Russ.). doi: 10.15827/0236-235X.149.055-064

**Article info**

Received: 02.08.2024

After revision: 13.08.2024

Accepted: 23.08.2024

**Abstract.** The paper discusses issues related developing one of the main blocks of a real-time computing system, specifically the computation scheduling block. The authors propose algorithms for constructing optimal schedules for different cases depending on the number of processors and characteristics of works and computing system resources. For the single-processor case with interruptions and directive intervals, they improved the relative urgency algorithm using a heap for data storage. This contributed to lowering the algorithm computational complexity. The authors also developed an algorithm for a problem with a partial order of job execution. It bases on the pre-correction of ready moments and directive deadlines and on the reduction of the original task to a task without precedence relations. For the multiprocessor case with interruptions and directive intervals, the authors proposed an approximate algorithm that is based on a generalization of the single-processor relative urgency algorithm to the multi-processor case. The authors performed a comparative analysis with the exact stream algorithm. They proved that the problem is NP-hard when interruption and switching time costs are taken into account. For the multiprocessor case without interruptions and switches with a common directive interval for all works and identical processors, the authors developed a pseudo-polynomial algorithm based on a limited search of options. The authors also created an approximate algorithm for a system with renewable and non-renewable resources, as well as for a complex with a mixed set of works (both continuous and allowing interruptions and switching). The algorithm is based on network modeling and reducing the problem under study to the search for a stream with certain properties in a special network.

**Keywords:** real-time computing system, single-processor and multiprocessor systems, stream network, acceptable schedule, renewable and non-renewable resources

**References**

1. Glonina, A.B. (2017) 'General model of real-time modular computer systems operation for checking acceptability of such systems configurations', *Bull. of the SUrSU. Ser. Math. Mechanics. Physics*, 6(4), pp. 43–59 (in Russ.). doi: 10.14529/cmsel70404.
2. Glonina, A.B., Balashov, V.V. (2018) 'On the correctness of real-time modular computer systems modeling with stopwatch automata networks', *Modeling and Analysis of Information Systems*, 25(2), pp. 174–192 (in Russ.). doi: 10.18255/1818-1015-2018-2-174-192.
3. Glonina, A.B. (2020) 'A tool system for schedulability analysis of modular computer systems configurations', *Bull. of MSU. Computational Math. and Cybernetics*, (3), pp. 16–29 (in Russ.).
4. Kononov, D.A., Furugyan, M.G. (2023) 'Situational operational planning in real-time systems', *Proc. Int. Conf. MLSD*, (1), pp. 1138–1144 (in Russ.).
5. Tanaev, V.S., Gordon, V.S., Shafranskii, Ya.M. (1984) *Theory of Scheduling: Single-Stage Systems*. Moscow, 383 p. (in Russ.).
6. Brucker, P. (2007) *Scheduling Algorithms*, Springer Publ., 378 p.
7. Koshelev, P.S., Mishchenko, A.V. (2021) 'Optimizing management of jobs in a logistic project under conditions of uncertainty', *J. of Computer and Systems Sci. Int.*, 60(4), pp. 595–609. doi: 10.1134/S1064230721040079.
8. Gorskii, M.A., Mishchenko, A.V., Nesterovich, L.G., Khalikov, M.A. (2022) 'Some modifications of integer optimization problems with uncertainty and risk', *J. of Computer and Systems Sci. Int.*, 6(6), pp. 65–76 (in Russ.). doi: 10.31857/S0002338822050079.
9. Kosorukova, O.A., Lemtyuzhnikova, D.V., Mishchenko, A.V. (2023) 'Methods and models of project resource management under uncertainty', *J. of Computer and Systems Sci. Int.*, (3), pp. 38–56 (in Russ.). doi: 10.31857/S0002338823020117.
10. Lazarev, A.A. (2019) *Scheduling Theory. Methods and Algorithms*. Moscow, 408 p. (in Russ.).
11. Kostenko, V.A., Smirnov, A.S. (2019) 'Flow algorithms for scheduling computations in integrated modular avionics', *J. of Computer and Systems Sci. Int.*, 58(3), pp. 404–414. doi: 10.1134/S1064230719030110.
12. Balashov, V.V., Kostenko, V.A., Fedorenko, I.A., Gao, C., Sun, C.M., Sun, C. (2023) 'Simulated annealing algorithm for constructing list schedules with restrictions on the number of interprocessor data transfers', *Automation and Remote Control*, (8), pp. 138–152 (in Russ.). doi: 10.31857/S0005231023080093.
13. Gorman, M.F., Conway, D.G. (2018) 'A tutorial of integrating duality and branch and bound in earliness–tardiness scheduling with idle insertion time problems', *IJPR*, 56(1-2), pp. 262–277. doi: 10.1080/00207543.2017.1397794.
14. Graves, S.C. (2022) 'How to think about planned lead times', *SSRN Electronic J.* doi: 10.2139/ssrn.3485059, available at: [https://papers.ssrn.com/sol3/papers.cfm?abstract\\_id=3485059](https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3485059) (accessed July 10, 2024).
15. Thomasson, O., Battarra, M., Erdoğan, G., Laporte, G. (2018) 'Scheduling twin robots in a palletising problem', *IJPR*, 56(1-2), pp. 518–542. doi: 10.1080/00207543.2017.1401249.
16. Coffman, E.G. (1976) *Computer and Job-Shop Scheduling Theory*. John Wiley & Sons Publ., 299 p. (Russ. ed.: (1984) Moscow, 336 p.).

17. Furugyan, M.G. (2014) 'Some algorithms for analysis and synthesis of real-time multiprocessor computing systems', *Programming and Computer Software*, 40(1), pp. 21–27. doi: 10.1134/S0361768814010034.
18. Cormen, T.H., Leiserson, Ch.E., Rivest R.L., Stein, C. (2011) *Introduction to Algorithms*. The MIT Press, 1312 p. (Russ. ed.: (2011) Moscow, 1296 p.).
19. Kononov, D.A., Furugyan, M.G. (2023) 'Distribution of heterogeneous complex of resources on a multiprocessor system', *Proc. Int. Conf. SUMMA*, pp. 368–372. doi: 10.1109/SUMMA60232.2023.10349570.

**Авторы**

**Кононов Дмитрий Алексеевич**<sup>1</sup>, д.т.н.,  
доцент, ведущий научный сотрудник,  
dmitrykon52@gmail.com

**Фуругян Меран Габибллаевич**<sup>2</sup>, к.ф.-м.н.,  
доцент, старший научный сотрудник,  
rtscas@yandex.ru

<sup>1</sup> Институт проблем управления  
им. В.А. Трапезникова РАН,  
г. Москва, 117997, Россия

<sup>2</sup> Федеральный исследовательский центр  
«Информатика и управление» РАН,  
г. Москва, 119333, Россия

**Authors**

**Dmitry A. Kononov**<sup>1</sup>, Dr.Sci. (Engineering),  
Associate Professor, Leading Researcher,  
dmitrykon52@gmail.com

**Meran G. Furugyan**<sup>2</sup>,  
Cand. of Sci. (Physics), Associate Professor,  
Senior Researcher, rtscas@yandex.ru

<sup>1</sup> Trapeznikov Institute of Control Sciences  
of the Russian Academy of Sciences,  
Moscow, 117997, Russian Federation

<sup>2</sup> Federal Research Center "Computer Science  
and Control" of the Russian Academy of Sciences,  
Moscow, 119333, Russian Federation

## Система верифицируемых спецификаций программных компонентов с поддержкой встраивания и извлечения

П.А. Шапкин <sup>1</sup>✉<sup>1</sup> Национальный исследовательский ядерный университет «МИФИ», г. Москва, 115409, Россия

### Ссылка для цитирования

Шапкин П.А. Система верифицируемых спецификаций программных компонентов с поддержкой встраивания и извлечения // Программные продукты и системы. 2025. Т. 38. № 1. С. 65–76. doi: 10.15827/0236-235X.149.065-076

### Информация о статье

Группа специальностей ВАК: 2.3.5

Поступила в редакцию: 06.03.2024

После доработки: 12.07.2024

Принята к публикации: 24.07.2024

**Аннотация.** Объектами данного исследования являются спецификация и верификация программных систем и их компонентов. Предмет исследования – унифицированный язык спецификаций, оснащенный соотношением как с системами случайного тестирования, так и со средствами статической верификации на основе систем типов. Разнообразие языков программирования, систем конфигурирования, развертывания и другие инструменты требуют от разработчиков усилий по их интеграции. Упростить задачу помогает наличие верифицируемых спецификаций компонентов. В работе предложен подход к унифицированному представлению спецификаций, интегрированному с системами как для статической проверки типов, так и для динамического тестирования. Это решение опирается на методы аппликативных вычислительных систем и теории типов и предоставляет понятийный каркас для построения спецификаций, встраиваемых в различные программные среды. Недостаток возможностей статической верификации из-за ограниченности систем типов до некоторой степени устраняется за счет динамического тестирования. Тестирование осуществляется посредством интерпретации спецификаций в определении для систем случайного тестирования на основе свойств. Практическая значимость предлагаемого подхода состоит, в частности, в автоматизации процесса построения типизированных оберток, или фасадов, необходимых для использования компонентов из менее типизированных сред в языках программирования с более выразительными системами типов. Автоматизируются как верификация таких оберток, так и способы их построения за счет определения операций уточнения спецификаций. На практике это позволяет выявлять ошибки в типизации сторонних компонентов на ранних стадиях разработки. В статье приведены примеры спецификаций программ с побочными эффектами. В качестве основы для спецификаций использованы формализации из теории категорий. Проанализированы подходы к транслированию спецификаций в другие представления и к итеративному усовершенствованию спецификаций путем их трансформации.

**Ключевые слова:** программные системы, спецификация, верификация, семантика, логика, типизация

**Введение.** Важность задачи спецификации и верификации программных компонентов неуклонно растет, что обусловлено рядом тенденций реализации информационных технологий. Развитие *искусственного интеллекта* (ИИ) привело к тому, что уже более десятой доли программного кода генерируется автоматически, а неточная природа получаемых от ИИ решений означает необходимость создания и совершенствования средств спецификации задач и верификации сгенерированных компонентов. Появление и развитие новых платформ и подходов к вычислениям, таких как облачные технологии, блокчейн, дают толчок к созданию новых языков и инфраструктур программных компонентов, требующих интеграции между собой и с традиционными языками общего профиля [1], а значит, спецификации должны быть переносимыми и погружаемыми в различные вычислительные среды. Достижения в области логики и теории типов [2] обусловили создание мощных средств для построения ста-

тически верифицируемых программ, однако они сложны для использования программистами среднего или начального уровня, а прогнозируемое двукратное ускорение роста численности начинающих программистов вряд ли улучшит данную ситуацию. Существуют системы и языки, такие как TLA+ Why3, OpenJML, Frama-C, CPAchecker, специально предназначенные для описания и верификации спецификаций, а также подходы к верификации путем преобразования программного кода к представлению на языке спецификаций [3, 4], однако их применение не всегда возможно в силу высокой сложности полной трансляции кода программы. Более просты в использовании средства, верифицирующие спецификации путем динамического анализа, такие как Ortac [5], но обычно они являются узкоспециализированными.

В настоящей работе описывается подход к спецификации программных компонентов, реализованный автором в библиотеке Quasi-

Туре и дающий в зависимости от возможностей целевой среды отчуждаемые спецификации, оснащенные средствами погружения в различные вычислительные среды для дальнейшей динамической или статической верификации. Целью является выполнение таких задач, как верификация решений, построенных на нетипизированных компонентах, взаимодействующих через интеграционное приложение на типизированном языке. В качестве примера может рассматриваться технология Scala.js, позволяющая использовать типизированный язык Scala для написания кода JavaScript (JS), нетипизированные внешние компоненты которого подключаются посредством типизированных оберток, или фасадов. Поскольку JS-код внешних компонентов недоступен для типизации, указанные в фасадах типы могут не соответствовать действительному поведению программы и требовать дополнительной верификации. При такой верификации объявленные структуры типов выступают в роли спецификаций, что приводит к необходимости решения задачи отображения типов на спецификации.

### Обзор подходов к спецификации и верификации программ

С формальной точки зрения спецификация является логическим высказыванием, программа задается при помощи некоторого вычислительного формализма, такого как машина Тьюринга, алгоритм Маркова, рекурсивная функция, терм аппликативной вычислительной системы и т.д., верификация представляет собой процедуру проверки истинности заданного высказывания для заданной программы.

В качестве примера можно рассмотреть спецификацию и верификацию функций сортировки списка. Существует спецификация, формализующая понятие сортировки списка в виде логического высказывания. Все алгоритмы сортировки (быстрая, прямая вставка, пузырьковая и другие) должны удовлетворять этой спецификации. Однако, выполняя одну и ту же задачу, отдельные алгоритмы различаются в частных аспектах, например, в области операционных свойств, таких как вычислительная сложность по времени, по памяти. Соответственно, эти различия могут быть формализованы дополнительными спецификациями, более специфичными, чем общая спецификация сортировки.

Из приведенного примера видно, что на спецификациях задано отношение частичного по-

рядка, соответствующее родовидовым связям обобщения-специализации, которые будем называть вложенностью спецификаций. При этом программы можно считать спецификациями наиболее специализированного вида. Точнее, программа может сама по себе выступать в роли спецификации, и в этом случае она понимается как эталонная реализация, которой должны соответствовать верифицируемые относительно нее программы.

Для представления программ будем использовать подход *аппликативных вычислительных систем* (ABC): лямбда-исчисления [6] и комбинаторную логику [7]. С инженерной точки зрения данный подход выражается парадигмой функционального программирования – трактовкой программы как чистой математической функции. В отличие от императивного подхода, когда программа рассматривается как набор инструкций, выполнение которых ведет к изменению состояния, вычисление значения функции зависит только от входных значений и не подразумевает использование изменяемого состояния. В этом заключается основная причина выбора данного подхода: он дает возможность напрямую применять весь соответствующий математический аппарат как для синтеза программ, так и для анализа их логических свойств. Предлагаемое решение прежде всего нацелено на применение в связке с языками Scala, Coq, Agda и другими, ориентированными на использование теоретико-типového и функционального подходов, основанных на ABC. Типизированные обертки для сторонних компонентов также должны следовать функциональному стилю.

Проверка истинности, выполняемая при верификации, может быть как исчерпывающей, так и частичной, покрывающей только часть диапазона входных значений.

Полноту проверки в общем случае может гарантировать только статическая верификация, предметом которой является синтаксическая структура терма программы. Данная структура предоставляет исчерпывающую информацию о программе. Другими словами, проверка осуществляется по принципу белого ящика. В рамках ABC статическую верификацию можно свести к типизации программы [8]. Связь между типизацией и проверкой логических высказываний о программе устанавливается в теории типов через так называемый изоморфизм Карри–Говарда [9] или соответствие Брауэра–Гейтинга–Колмогорова [10, 11], которые связывают типизированные вычислительные фор-

мализмы с логическими системами: типы – с высказываниями, а термы – с предметами и доказательствами.

Полная статическая верификация на соответствие спецификации уровня логики предикатов возможна в языках программирования с зависимыми типами: Coq, Agda и т.п., но при этом нередко необходимо оснащение программы дополнительным кодом доказательств корректности. Значительно большее распространение имеют языки программирования, в которых возможности статической типизации ограничены либо отсутствуют. Это означает, что исчерпывающе может быть проверен только узкий набор свойств программ. Среди функциональных языков программирования распространены языки с полиморфными системами типов уровня системы  $F$ , позволяющими выражать спецификации, оперирующие типами аргументов функций, но не отдельными значениями внутри этих типов.

Под динамической верификацией программы понимается тестирование, то есть проверка результатов исполнения программы на тестовых данных. Поскольку осуществляется анализ внешнего поведения, а не внутренней структуры программы, можно назвать тестирование верификацией по принципу черного ящика. Построение тестовых примеров проводится так, чтобы удостовериться, что программа ведет себя в соответствии с некоторой эталонной конструкцией, которая также называется тестовым оракулом. Таким образом, можно отметить, что динамическая верификация основана на математической модели, то есть на денотационной семантике программы.

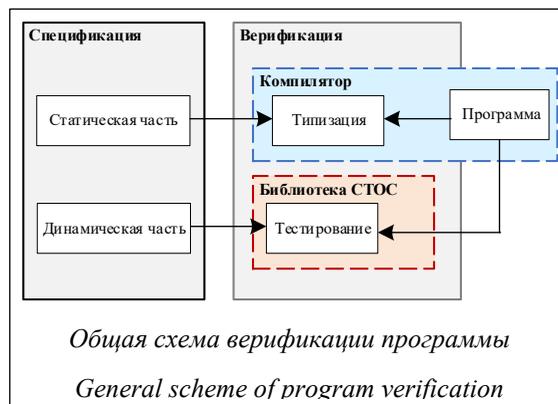
Частичность динамической верификации обусловлена тем, что в большинстве случаев за конечное время невозможно провести исчерпывающую проверку всех вариантов поведения программы из-за слишком большого или неограниченного числа вариантов входных параметров.

В рамках функционального программирования появился и затем распространился на большинство популярных языков программирования подход *случайного тестирования на основе свойств* (СТОС) [12]. Он интересен тем, что позволяет описывать тесты в виде не конкретных примеров, а свойств, имеющих форму логических высказываний. Верификация осуществляется путем генерации заданного числа случайных тестовых примеров на основе таких высказываний. Таким образом, данный подход дает возможность напрямую формулировать

и проверять спецификации программ в виде логических высказываний, а полнота верификации управляется изменением числа генерируемых тестовых примеров.

Сформулируем общую схему процесса верификации, приведенную на рисунке. В зависимости от выразительности системы типов заданного языка программирования полная спецификация программы или ее часть может быть представлена соответствующими типовыми конструкциями и полностью верифицирована средствами статической типизации. Статическая верификация может потребовать дополнительной работы по написанию доказательств, которые будут проверены компилятором. Часть спецификации программы, не поддающаяся статической верификации, может быть представлена с использованием системы СТОС в виде свойств и протестирована динамически [13]. Динамическая верификация не требует дополнительного труда со стороны программиста, но при этом является частичной. В языках с выразительной системой типов, таких как Coq, возможно представление всей спецификации в виде типовой конструкции, однако по решению программиста она может быть частично верифицирована статически, а частично динамически (например, при недостатке временных ресурсов для разработки полноценных доказательств или для быстрого опровержения спецификации с целью ее дальнейшего уточнения). В языках с отсутствующими или менее выразительными системами типов динамическая часть спецификации должна формулироваться отдельно средствами библиотеки СТОС.

Распространенным способом динамической верификации сегодня является фаззинг [14]. Особенность фаззинга в том, что он не основывается на спецификации: случайные тесты проводятся таким образом, чтобы в результате итеративного процесса найти диапазоны входных



данных, на которых работает тестируемая система. С точки зрения верификации и спецификации как недостаток можно рассматривать то, что явно или неявно найденная в результате фаззинга спецификация поведения программы обычно отбрасывается и никак не фиксируется.

### Сравнение подходов

Введем набор критериев для сравнения подходов к верификации:

- полнота верификации (возможность полной или только частичной верификации спецификации путем подбора контрпримера для ее опровержения);
- необходимость разработки доказательств (требуется ли разработка дополнительного кода доказательств для верификации программы);
- выразительные спецификации (возможность описания спецификаций на уровне логики предикатов);
- итеративный вывод спецификаций (возможность вывода уточненных спецификаций по результатам верификации);
- отчуждаемость спецификаций (возможность переноса спецификаций между различными языками программирования и системами тестирования).

Сравнить предлагаемое решение (QuasiType) на основе выбранных критериев с описанными выше подходами позволяет таблица.

В настоящем исследовании решается задача разработки унифицированного представления спецификаций уровня логики предикатов, оснащенного средствами импорта/экспорта для дальнейшей статической или динамической верификации. Даются общий каркас для совмещения различных подходов к верификации на основе единой спецификации, а также средства

для итеративного уточнения или вывода спецификации на основе динамического тестирования.

### Метод построения спецификаций

На абстрактном уровне спецификация представляет собой логическое высказывание относительно какого-либо вычислительного объекта. Алгебра спецификаций задает систему операций, при помощи которых возможно построение таких спецификаций.

**Основная структура.** Для построения спецификаций используем систему теории множеств и логики предикатов. Под специфицируемыми объектами будем понимать предметы, являющиеся элементами некоторого множества  $U$ , называемого универсумом. Через  $Set$  обозначим совокупность всех подмножеств  $U$ . Под  $Prop$  будем понимать множество всех высказываний.

Рассмотрим основные предикаты, то есть способы построения высказываний из предметов. Классификация элементов  $U$  по принадлежности его подмножеств выполняется посредством функции  $- \in -: U \times Set \rightarrow Prop$ . Другим основным предикатом является предикат равенства предметов, который определяется отдельно для каждого (индекс  $s$  будет опускаться, если его значение ясно из контекста)  $s$  из  $Set$ :  $- =_s -: U \times U \rightarrow Prop$ .

Для построения выразительного языка спецификаций необходимо ввести в рассмотрение переменные. Множество всех переменных обозначим как  $Var$ . Переменные могут быть использованы в выражениях вместо конкретных предметов. Такие выражения будем называть формулами. Множество высказывательных формул (имеющих значение высказывания)

### Общая схема верификации программы

#### General scheme of program verification

Критерий	Типизация уровня системы $F$	Типизация с зависимыми типами	СТОС	Фаззинг	QuasiType
Полнота верификации	Да	Да	Нет	Нет	Да/Нет
Не требует доказательств	Да	Нет	Да	Да	Да
Выразительные спецификации	Нет	Да	Да	Нет	Да
Итеративный вывод спецификаций	Нет	Нет	Нет	Да	Да
Отчуждаемость спецификаций	Нет	Нет	Нет	Нет	Да

обозначим как  $PForm$ . Множество предметных формул (имеющих значение предмета) обозначим через  $UForm$ . В частности, переменные можно считать простейшими предметными формулами.

Важным способом построения высказываний является утверждение о том, что некоторая формула  $F$  дает истинное высказывание при подстановке любых предметов из заданного множества  $S$  вместо некоторой переменной  $x$ . С точки зрения логики данный способ аналогичен всеобщей квантификации:  $\forall x: S(F)$ . Однако аналогичный смысл имеет и определенный вид суждений:  $x: S \vdash F$ . Следуя стилю, принятому в области алгебры и теории категорий, будем использовать второй из указанных подходов и введем операцию  $(- : - \vdash -): Var \times Set \times PForm \rightarrow Prop$ .

Заданных выше операций достаточно, чтобы определить простейшую спецификацию  $eqRefl$ , утверждающую рефлексивность равенства для заданного множества:  $eqRefl(S: Set) := x: S \vdash x =_S x$ .

**Категорная структура.** Дальнейшим шагом в построении алгебры спецификаций является определение примитивов для конструирования специфицируемых предметов. В качестве основы для последующих построений примем подход теории категорий [15, 16], в частности, категории множеств  $Set$ . В рамках данного подхода предметами спецификаций становятся функции, которые также называют морфизмами или стрелками, а основным способом их построения – операция композиции. С формальной точки зрения это говорит о том, что введенное выше  $U$  понимается как множество морфизмов, а  $Set$  – как множество объектов категории. Операция композиции задается как  $- \circ -: U \times U \rightarrow U$ .

Необходимо заметить, что, строго говоря, композиция является частично определенной функцией, так как задана только для совместимых морфизмов, но для простоты изложения пока опустим это требование.

В рамках категории множеств объекты воспринимаются как множества и могут выступать в роли области определения (домена,  $dom$ ) или области значений (кодомена,  $cod$ ) морфизмов. Выделяются так называемые  $Hom$ -множества, содержащие морфизмы между заданными объектами:  $Hom(A, B : Set) := \{f \mid dom(f) = A \wedge cod(f) = B\}$ .

$Hom$ -множества будут использоваться в спецификациях для ограничения множества специфицируемых морфизмов.

Композиция должна быть ассоциативна, что формулируется в виде спецификации  $compAss(A, B, C, D: Set) := f : Hom(C, D), g : Hom(B, C), h : Hom(A, B) \vdash (f \circ g) \circ h =_{Hom(A, D)} f \circ (g \circ h)$ .

Для каждого объекта  $A$  задается так называемый единичный морфизм, или идентичность,  $id_A$ , обладающий следующими свойствами:  $idL(A, B: Set) := f: Hom(A, B) \vdash id_B \circ f = f$ ,  $idR(A, B: Set) := f: Hom(A, B) \vdash f \circ id_A = f$ .

Несложно показать, что из указанных свойств следует однозначность идентичности для каждого объекта: пусть  $i$  и  $j$  – две идентичности для некоторого объекта, тогда  $i =_{(idR)} i \circ j =_{(idL)} j$ , то есть из того, что оба морфизма являются идентичностями, следует их неотличимость. Значит, уместно говорить об уникальной стрелке  $id_A$  для каждого объекта  $A$ .

**Формулы и связывание.** Рассмотрим подробнее использование переменных и формул. Следует заметить, что приведенные спецификации по своей синтаксической структуре противоречат определению утверждений. В частности, согласно определению, справа от символа  $\vdash$  должна находиться формула из  $PForm$ , в то время как все приведенные спецификации содержат справа от  $\vdash$  результат композиции, то есть элемент  $U$ . В свою очередь и аргументы композиции в этих спецификациях, строго говоря, не соответствуют ее области определения:  $f$  и  $g$  являются переменными, то есть элементами  $Var$ , в то время как композиция определена на элементах  $U$ .

Устранение указанных несоответствий состоит в проработке механизмов построения формул и связывания переменных со значениями. Это является важной задачей при построении структуры спецификаций, которая допускает не только погружение в какую-либо среду, уже оснащенную способами связывания, но и, наоборот, выгрузку спецификаций в виде самостоятельных конструкций для дальнейшего анализа.

Напомним, что формулы понимаются как конструкции, содержащие переменные. Формула приобретает значение, или означает, в среде – некотором контексте, связывающем переменные с конкретными значениями. Введем следующие механизмы построения формул.

Во-первых, любое значение порождает формулу, цитирующую это значение, то есть принимающую это значение в любой среде:  $quote: U \rightarrow UForm$ .

Во-вторых, переменные понимаются как другой простейший вид формул, наоборот, полностью зависящих от среды:  $Var \subseteq UForm$ .

В-третьих, более сложные формулы могут получаться из более простых путем применения особых версий обычных операций, таких как композиция, и предикатов, таких как равенство, «повышенных» до уровня операций на формулах. Соответствующую операцию, преобразующую функции на  $U$  и  $Prop$  в функции на  $UForm$  и  $PForm$ , обозначим  $[-]^F$ . Таким образом, более точное определение, к примеру,  $idL$  имеет следующий вид:  $idL(A, B: Set) := f: Hom(A, B) \vdash quote(id_B)[\circ]^F \mathcal{L} = ]^F f$ .

Далее будем, как и ранее, подразумевать данную структуру неявным образом.

**Дополнительные структуры.** Рассмотрим дополнительные структуры, используемые в спецификациях. За основу взяты стандартные определения из теории категорий, которые адаптированы к представлению в виде спецификаций на основе определенной выше структуры.

**Терминальный объект.** Терминальный объект будем обозначать как  $1$ , он оснащается следующей спецификацией:  $unit(A: Set) := f, g: Hom(A, 1) \vdash f = g$ .

**Произведение.** Декартово произведение объектов  $A$  и  $B$  обозначается как  $A \times B$  и оснащается стрелками-проекциями  $\pi_1: A \times B \rightarrow A$ ,  $\pi_2: A \times B \rightarrow B$ . Для каждого объекта  $X$  и пары совместимых  $f: X \rightarrow A$  и  $g: X \rightarrow B$  определена стрелка  $\langle f, g \rangle$  таким образом, что выполняются спецификации  $\pi_1(X, A, B: Set) := f: Hom(X, A)$ ,  $g: Hom(X, B) \vdash \pi_1 \circ \langle f, g \rangle = f$  и  $\pi_2(X, A, B: Set) := f: Hom(X, A)$ ,  $g: Hom(X, B) \vdash \pi_2 \circ \langle f, g \rangle = g$ .

**Экспоненцирование.** Для любых двух объектов  $A$  и  $B$  задан экспоненциальный объект  $B^A$  вместе со стрелкой  $\varepsilon_{AB}: B^A \times A \rightarrow B$ . Также для каждого  $f: A \times B \rightarrow C$  задана стрелка  $\lambda_{ABCf}: A \rightarrow C^B$ . При этом должна выполняться спецификация  $applyLam(A, B, C: Set) := f: A \times B \rightarrow C \vdash \varepsilon_{BC} \circ (\lambda_{ABCf} \times id_B) = f$ .

**Функтор**  $F$  представляет собой пару функций  $F_o: Set \rightarrow Set$  и  $F_m: U \rightarrow U$ , таких, что справедливы спецификации  $functorId(F)(A: Set) := F_m(id_A) = id_{F_o(A)}$  и  $functorComp(F)(A, B, C: Set) := f: Hom(B, C), g: Hom(A, B) \vdash F_m(f \circ g) = F_m(f) \circ F_m(g)$ .

Поскольку функции  $F_o$  и  $F_m$  применимы, соответственно, к объектам и стрелкам, а значит, используются в разных местах, обычно их можно отличить исходя из контекста, а индексы опустить. Исходя из того, что фактически работа ведется в одной и той же категории, все функторы являются так называемыми эндофункторами. Идентичным функтором называется функтор  $Id$ , такой, что функции  $Id_o$  и  $Id_m$  являются идентичностями.

**Композиция функторов**  $F$  и  $G$  представляет собой такой функтор  $F \circ G$ , что  $(F \circ G)_o(A: Set) := F_o(G_o(A))$ ,  $(F \circ G)_m(f: U) := F_m(G_m(f))$ .

Композиция функторов обладает свойством ассоциативности.

**Естественное преобразование** из функтора  $F$  в функтор  $G$  обозначается как  $\eta: F \rightarrow G$  и представляет собой функцию  $\eta: Set \rightarrow U$ , порождающую для каждого объекта  $A$  стрелку  $\eta_A$  таким образом, что справедливы спецификации  $ntIn(\eta)(A: Set) := \eta_A \in Hom(F(A), G(A))$ ,  $ntComp(\eta)(A, B: Set) := f: Hom(A, B) \vdash \eta_B \circ F(f) = G(f) \circ \eta_A$ .

**Монада** состоит из эндофунктора  $F$  и двух естественных преобразований  $\eta_F: Id \rightarrow F$  и  $\mu: F^2 \rightarrow F$ , таких, что выполняются спецификации  $muComp(F)(A: Set) := \mu_F(A) \circ F(\mu_F(A)) = \mu_F(A) \circ \mu_F(F(A))$ ,  $muEtaCompR(F)(A: Set) := \mu_F(A) \circ F(\eta_F(A)) = id_{F(A)}$ ,  $muEtaCompL(F)(A: Set) := \mu_F(A) \circ \eta_F(F(A)) = id_{F(A)}$ .

Таким образом, определение монады основано на вышеописанных определениях функторов и естественных преобразований, но дает мощное средство для оперирования большим количеством разнородных структур, используемых в программировании, – от списков значений до побочных эффектов и потоков ввода-вывода.

### Примеры спецификаций

Рассмотрим примеры использования описанной алгебры для спецификации некоторых простых программных интерфейсов. Определение спецификаций в алгебраическом стиле позволяет решать многие задачи простым применением конкретных интерпретаторов к абстрактным спецификациям. В эту категорию попадают задачи, опирающиеся на естественную структуру спецификаций. К примеру, спецификации могут быть интерпретированы как тесты соответствующих компонентов, а также как генераторы текстовых описаний на естественном или некотором внешнем формальном языке, используемом для выгрузки спецификаций в сторонние системы для дальнейшей интерпретации, формальной верификации и т.п.

**Спецификация изменяемых значений.** Приведем пример простейшей спецификации стандартного интерфейса для работы с изменяемым значением – так называемых аксессоров  $get$  и  $set$ . Этот довольно простой пример в то же время является достаточно общим, практически напрямую применяемым к другим задачам чтения и изменения данных в различных про-

граммных интерфейсах, таких как работа с БД, операции REST API и другие.

Предложим следующие спецификации:  $get := get \in Hom(1, \mathbb{N})$ ,  $set := set \in Hom(\mathbb{N}, 1)$ ,  $setGet := get \circ set = id_{\mathbb{N}}$ .

Спецификации  $get$  и  $set$  задают типы соответствующих функций, в то время как спецификация  $setGet$  характеризует смысловую связь этих операций.

Недостатком рассмотренной спецификации является то, что она не учитывает побочные эффекты, которые неизбежно связаны с обработкой изменяемых значений. Вследствие этого при интерпретации данной спецификации в виде тестов при их параллельном выполнении тест  $setGet$  в большинстве случаев возвращает ошибку.

Более точно это несоответствие выражается нарушением спецификации вида  $pureness := i : Hom(1, \mathbb{N}) \vdash get = get \circ set \circ i$ , справедливость которой требуется исходя из свойств терминального объекта  $1$  и тождественного морфизма. Поскольку из любого объекта существует только единственный морфизм, любой морфизм типа  $1 \rightarrow 1$  должен совпадать с тождественным морфизмом  $id_1$ . Одним из таких морфизмов является  $set \circ i$  для любого  $i \in Hom(1, \mathbb{N})$ . Но из того, что  $set \circ i = id_1$ , следует, что  $get \circ set \circ i = get \circ id_1 = get$ . Очевидно, что это равенство нарушается в зависимости от того, изменяется ли начальное значение при выполнении операции  $set$ . В совокупности вышесказанное означает, что неверно использовать  $1$  в роли области значений для  $set$ . Содержательно при трактовке  $1$  как одноэлементного множества такое определение говорило бы о том, что результат выполнения  $set$  всегда одинаков, однако это не так: выполнение  $set$  с различными аргументами дает различные состояния, хотя содержание этих состояний не возвращается в явном виде. Аналогично неверно рассматривать  $1$  и как значение  $get$ , так как различные начальные состояния могут приводить к разным результатам выполнения  $get$ .

**Учет побочных эффектов.** Исправить приведенную выше спецификацию можно, введя в рассмотрение побочные эффекты вычислений, представленные через соответствующую монаду. Пусть  $IO$  – это функтор и оснащающая его монада, соответствующая реализации побочных эффектов. Рассмотрим спецификацию аксессуаров, учитывающую связь операций  $get$  и  $set$  с побочными эффектами:  $getPure := get \in Hom(1, IO(\mathbb{N}))$ ,  $setPure := set \in Hom(\mathbb{N}, IO(1))$ ,  $setGetPure := \mu_{IO(\mathbb{N})} \circ IO(\mathbb{N}) \circ set = \eta_{IO(\mathbb{N})}$ .

Данная спецификация разрешает несоответствие области значений аксессуаров терминальному объекту. Содержательно замена  $1$  на  $IO(1)$  явно описывает то, что результат этих операций не предопределен, а скрыто зависит от внешнего состояния или же изменяет его. Соответственно, в отличие от  $1$  объект  $IO(1)$  не обладает свойствами, которые приводили к противоречиям. В то же время монадическая структура  $IO$  позволяет проводить необходимые композиции операций. С точки зрения интерпретации спецификаций как тестов обновленные спецификации сами по себе не делают успешными тесты, ранее приводящие к ошибкам. Например, параллельное выполнение тестов все еще может завершаться неудачей. Однако обновленные спецификации в явном виде сообщают внешней среде о наличии побочных эффектов, и, имея эти данные, среда выполнения тестов может принимать решения о том, какие дополнительные свойства эффектов проверяются. Конечно, возможно составление дополнительных спецификаций поведения операций при параллельном выполнении.

## Язык спецификаций

Представленная алгебраическая структура спецификаций хорошо подходит для интерпретации спецификаций в разных целях. Однако интерпретация возможна, если существует закодированная структура конкретной спецификации. Если определение спецификации отсутствует и стоит задача считывания ее в виде данных или же, наоборот, сохранения закодированной спецификации в виде данных для дальнейшего считывания, то требуется язык спецификаций, на котором можно записать такие данные. Являясь двойственными представлениями одной и той же сущности, язык и алгебра спецификаций строятся похожим образом и могут быть выведены друг из друга.

Наиболее простым способом построения языка спецификаций является ввод всех описанных выше конструкций в виде отдельных синтаксических единиц.

Начнем с базовых возможностей, позволяющих описывать высказывания ( $P$ ) относительно предметов ( $U$ ) и множеств ( $S$ ), а также строить высказывательные формулы ( $PF$ ):

$$P ::= U \in S \mid U = {}_S U \mid PF^\downarrow,$$

$$PF ::= S \vdash PF.$$

Далее в грамматику вводятся средства определения предметов и множеств, основанные на категорном подходе, необходимые для форму-

лировки спецификаций через композиции функций (морфизмов), а также с учетом областей определения и значений:

$$U ::= U \circ U \mid id_S, \\ S ::= Hom(S, S).$$

Полученную грамматику можно расширять, добавляя новые конструкции. Приведем пример добавления терминального объекта и произведений:

$$U ::= \dots \mid \langle U, U \rangle \mid \pi_1 \mid \pi_2, \\ S ::= \dots \mid S \times S \mid 1.$$

Отдельная синтаксическая категория формул  $PF$  введена как представление результата квантификации  $\vdash$  и требует наличия методов представления связывания, описанных выше. Введем их следующим образом:

$$UF ::= 'U' \mid \bar{n} \mid UF \circ UF \mid \langle UF, UF \rangle, \\ PF ::= \dots \mid UF \in S \mid UF = {}_sUF.$$

Здесь  $UF$  представляет собой синтаксическую категорию предметных формул, которые могут содержать связанные переменные. При этом связывание осуществляется не по именам связанных переменных, а по соответствующим им индексам де Брейна [17]:  $n$ -й индекс обозначается как  $\bar{n}$ . Формула, цитирующая некоторый предмет  $u$ , обозначена как  $'u'$ . Кроме того, расширение грамматики содержит повышенные до уровня формул аналоги операций  $\circ$ ,  $\langle -, - \rangle$ ,  $- \in -$  и  $- = -$ , которые используют тот же символ, что и оригинальная операция, но отличаются друг от друга окружающим контекстом.

## Интерпретация

Представленную алгебру спецификаций можно интерпретировать для разных целей. Технически используется подход Tagless Final [18, 19]. В рамках объектно-ориентированного подхода Scala это означает, что основная алгебра реализуется в виде системы абстрактных классов и трейтов, а интерпретаторы – в виде конкретных реализаций этих абстрактных определений. На формальном уровне интерпретация понимается в смысле, близком к понятию интерпретации формальной системы: всем носителям абстрактной алгебраической системы ставятся в соответствие конкретные множества или типы программных объектов, а всем операциям – конкретные операции на соответствующих типах.

**Интерпретация для тестирования.** Целью данного проекта является построение системы описания спецификаций, погружаемой в различные вычислительные среды для верифика-

ции конкретных программных компонентов. Соответственно, основной формой интерпретации спецификаций является генерация тестов в рамках подхода СТООС.

Настоящий проект реализуется на языке Scala, в связи с чем спецификации интерпретируются в виде определений свойств ScalaCheck. Фактически разработанная структура спецификаций во многом повторяет структуру свойств Prop системы ScalaCheck и основана на ней. В то же время спецификации QuasiType являются более высокоуровневыми, что позволяет проводить их выгрузку для последующего погружения в другие среды – в реализации подхода СТООС на других языках. С этой точки зрения алгебра QuasiType предоставляет общую структуру спецификаций, относительно которой QuickCheck, ScalaCheck и другие играют роль реализаций в определенных вычислительных средах, а соответствующие интерпретаторы QuasiType устанавливают соответствие между абстрактной алгеброй и конкретной реализацией.

Основными составными частями систем СТООС являются структуры для определения свойств, как правило, имеющих тип Prop, и средства генерации случайных значений различных типов, обычно сводящихся к структурам типов Gen (генераторы) и Cogen (когенераторы).

Например, свойство ассоциативности композиции в ScalaCheck выглядит как

```
forall((f: Int => Int, g: Int =>
=> Int, h: Int => Int, x: Int) =>
  f.compose(g).compose(h)(x) ==
==f.compose(g.compose(h))(x)) :
Prop,
```

и его тестирование состоит в выполнении содержащейся в нем функции для набора случайных аргументов. Для генерации случайных аргументов типа  $T$  используется генератор типа  $Gen[T]$ , и он передается в forall в качестве неявного параметра. Для обеспечения повторяемости тестов генерация псевдослучайных значений управляется особым значением типа Seed. Таким образом,  $Gen[T]$  фактически представляет собой функцию  $Seed \Rightarrow T$ . Заметим, что для проверки ассоциативности композиции требуется генерация случайных значений функционального типа  $Int \Rightarrow Int$ . Случайная функция, с одной стороны, должна возвращать случайный результат, но, с другой, не должна однозначно зависеть от аргумента функции. Таким образом, для генерации случайной функции типа  $A \Rightarrow B$  необходим способ случайного значения типа  $B$ , в то же время управ-

ляемого значением типа  $A$ . Исходя из того, что все случайные значения управляются значением  $Seed$ , данная задача сводится к наличию способа получения  $Seed$  из  $A$  – то есть функции типа  $A \Rightarrow Seed$ , который также именуется  $Cogen[T]$ . Таким образом, эти когенераторы необходимы для построения генераторов функциональных значений с соответствующей областью определения. Поскольку поддерживаются спецификации функций с побочными эффектами, выполненная реализация использует библиотеку `scalacheck-effect`, дополняющую `ScalaCheck` возможностями использования побочных эффектов на основе библиотек `cats` и `cats-effect` путем подмены обычных свойств типа `Prop` свойствами с эффектом `PropF[TEff]`.

Интерпретация основывается на следующих носителях:

- высказываниям `Prop` ставятся в соответствие значения типа `PropF[IO]`;
- морфизмам из  $U$  ставится в соответствие тип функций `Any => Any`, позволяющий избегать ошибок типизации, опираясь на соответствующие динамические проверки;
- множествам `Set` ставятся в соответствие структуры, позволяющие, с одной стороны, генерировать значения соответствующего типа для выполнения тестов, а с другой, проверять истинность базовых предикатов.

Класс `Set`, интерпретирующий множества из `Set`, представляет собой класс кортежей вида

```
( gen: Gen[Any],
  cogen: Cogen[Any],
  elem(m: Any): PropF[IO],
  eq(x: Any, y: Any):
  PropF[IO] ).
```

Здесь `gen` и `cogen` представляют соответственно генератор и когенератор. Текущая реализация интерпретатора является нетипизированной в том смысле, что не использует или, точнее, не сохраняет конкретные типы, заменяя их общим типом `Any`: каждое множество соотносится с конкретным генератором значений конкретного типа, однако затем этот тип стирается, не влияя на работу созданного генератора. Значения `elem` и `eq` интерпретируют, соответственно, предикаты  $\in_s$  и  $=_s$  для соответствующего множества  $s$ .

Помимо структуры `Set`, остальная часть интерпретатора практически напрямую ставит в соответствие операциям на морфизмах соответствующие им операции на функциях `Scala`, а также интерпретирует остальные возможности методами стандартной библиотеки, биб-

лиотек `cats` и других. Например, произведение интерпретируется стандартным классом пар и оснащающими его функциями проекций. Монада, определяемая некоторым функтором и операциями  $\eta$  и  $\mu$ , интерпретируется монадой из `cats` и соответствующими операциями `fmap`, `pure` и `flatten`.

После применения интерпретатора спецификации, описывающие поведение операций  $\eta$ ,  $\mu$  и других, автоматически становятся исполняемыми тестами соответствующих методов функций `pure` и `flatten` и т.д.

**Интерпретация для транслирования.** Возможность выгрузки и переноса спецификаций между различными средами или языками программирования (транслирование спецификаций) достигается путем представления их в виде не абстрактных выражений, а конкретных значений. Структура таких значений соответствует грамматике языка спецификаций, приведенной ранее.

Интерпретатор для целей транслирования состоит из иерархии классов, представляющих собой синтаксические сущности грамматики, и реализации абстрактной алгебры, ставящей в соответствие каждой операции конструктор соответствующего класса. Такой интерпретатор используется для экспорта спецификаций в виде конкретных структур, которые затем могут быть сериализованы с использованием стандартных коммуникативных форматов, таких как `JSON`, `YAML`, `XML` и другие.

Помимо интерпретатора, позволяющего выгружать спецификации, целесообразно реализовать анализатор для считывания спецификаций из внешних источников. Часть анализатора, отвечающая за чтение или десериализацию из внешних форматов, может быть автоматически сгенерирована стандартными средствами на основе структуры классов грамматики языка. Остается реализовать парсер, переводящий полученные значения в соответствующие им выражения в терминах алгебры спецификаций. Полученные выражения затем могут быть интерпретированы другими интерпретаторами для тех или иных целей.

Алгебра и язык спецификаций изоморфны друг другу. Значит, импорт любой спецификации с ее последующим экспортом должен соответствовать тождественному преобразованию, то есть давать в качестве результата исходную спецификацию. Выполненные в составе системы `QuasiType` реализации интерпретатора и парсера оснащены соответствующей спецификацией, проверяющей указанную изоморфность.

**Интерпретация для статической верификации.** Рассмотрим, каким образом предложенное представление спецификации может быть использовано не для динамической, а статической верификации программного кода при помощи типизации. Подход к статической проверке состоит в построении интерпретатора абстрактных спецификаций, связывающего используемые в спецификациях множества с типами целевого языка. Возможности верификации такого интерпретатора ограничиваются возможностями, предоставляемыми соответствующей системой типов: например, поддерживается ли типизация функциональных значений. Если же язык поддерживает в том числе и зависимые типы, то подобный интерпретатор может связывать с типовыми выражениями всю спецификацию, включая используемые в ней предикаты, и дальнейшая верификация может быть целиком осуществлена статически в форме написания доказательств. В свою очередь, верифицируемый код встраивается в спецификацию и таким образом должен проходить статическую проверку соответствующих типов при компиляции.

Следует отметить, что интерпретаторы для тестирования и для статической проверки являются граничными случаями и востребованными могут оказаться гибридные интерпретаторы, покрывающие случайными тестами те части спецификации, которые не выразить в рамках системы типов целевого языка программирования.

### Обсуждение: применение подхода к уточнению спецификаций

Возможности декомпозиции, преобразования и повторной интерпретации спецификаций позволяют строить правила и алгоритмы их уточнения и совершенствования.

В качестве примера рассмотрим задачу выявления в спецификациях программных интерфейсов функций-аксессоров для работы с изменяемыми значениями с последующим уточнением этих спецификаций. Поскольку преобразование основано в том числе и на информации об именах функций, будем считать, что задана функция  $name: U \rightarrow String$ , возвращающая имя заданного морфизма. Алгоритм данного преобразования можно представить в виде следующего правила вывода:

$$\begin{aligned} name(g) &= "get\langle n \rangle" & g &\in Hom(N, 1), \\ name(s) &= "set\langle n \rangle" & s &\in Hom(1, N) \\ \Rightarrow g \circ s &= id_N, \end{aligned}$$

где  $n$  является изменяемым значением, доступ к которому предоставляют  $g$  и  $s$ .

Данное правило выводит спецификацию согласованности «геттера» и «сеттера». Приведена версия, не учитывающая побочные эффекты. Идея состоит в том, чтобы на основе анализа имен и типов, например, методов некоторого класса, делать вывод о возможности воспринимать их в качестве аксессоров для некоторых значений и затем проверять справедливость этого вывода путем запуска уточненной спецификации в виде тестов.

### Заключение

В настоящей работе представлен подход к спецификации программных компонентов, дающий отчуждаемые спецификации, оснащенные средствами погружения в различные вычислительные среды для дальнейшей динамической или статической верификации. Он является основой для построения экосистемы программных средств с целью верификации и анализа функционала разнородных компонентов. Подход реализуется в проекте QuasiType (<https://www.tytip.com/ru/quasitype>), выполняемом на языке Scala.

Перспективным видится дальнейшее развитие подхода в различных направлениях. В частности, планируются разработка реализаций на других языках программирования, а также интеграция с такими системами, как Coq и Agda, позволяющими осуществлять логический вывод производных спецификаций. Планируется применение разработанного подхода для автоматизации верификации типизированных оберток (фасадов) над нетипизированными компонентами, такими как библиотеки JavaScript, средства работы с контейнеризированными приложениями и их API. Ограничением к применению подхода на данный момент является требование к наличию первичного фасада, на основе которого будет построена спецификация, способная затем итеративно верифицироваться и уточняться. Однако данное ограничение может ослабляться путем разработки анализаторов программного кода, автоматизирующих построение начальных спецификаций.

### Список литературы

1. Shapkin P. Automation of configuration, initialization and deployment of applications based on an algebraic approach. *Procedia Comput. Sci.*, 2022, vol. 213, pp. 785–792. doi: 10.1016/j.procs.2022.11.135.

2. Pierce B.C., de Amorim A.A., Casinghino Ch., Gaboardi M. et al. Logical foundations. *Software Foundations Ser.*, 2018, vol. 1. URL: <https://www.seas.upenn.edu/~cis5000/current/sf/lf-current/index.html> (дата обращения: 11.05.2024).
3. Methni A., Lemerre M., Hedia B.B. et al. An approach for verifying concurrent C programs. *Proc. 8th Junior Researcher Workshop on Real-Time Computing*, 2014, pp. 33–36.
4. Lau H., Nestmann U. Java goes TLA+. *Proc. V Int. Conf. on Theoretical Aspects of Software Eng.*, 2011, pp. 117–124. doi: 10.1109/TASE.2011.44.
5. Filliâtre J.-C., Pasutto C. Ortac: Runtime assertion checking for OCaml (tool paper). In: *LNPSE. Proc. RV*, 2021, vol. 12974, pp. 244–253. doi: 10.1007/978-3-030-88494-9\_13.
6. Барендрегт Х. Лямбда-исчисление. Его синтаксис и семантика. М.: Мир, 1985. 606 с.
7. Вольфенгаген В.Э. Комбинаторная логика в программировании. Вычисления с объектами в примерах и задачах. М.: изд-во ЮрИнфоР МГУ, 2008. 204 с.
8. Barendregt H. Introduction to generalized type systems. *J. of Functional Programming*, 1991, vol. 1, no. 2, pp. 125–154. doi: 10.1017/S0956796800020025.
9. Sørensen M.H., Urzyczyn P. Lectures on the Curry-Howard isomorphism. In: *Studies in Logic and the Foundations of Mathematics*, 2006, vol. 149, pp. 1–442.
10. Heyting A. Die intuitionistische Grundlegung der Mathematik. *Erkenntnis*, 1931, vol. 2, pp. 106–115. doi: 10.1007/BF02028143.
11. Колмогоров А.Н. Избранные труды. Математика и механика. М.: Наука, 1985. 476 с.
12. Claessen K., Hughes J. QuickCheck: A lightweight tool for random testing of Haskell programs. *ACM SIGPLAN Notices*, 2011, vol. 64, no. 4, pp. 53–64. doi: 10.1145/1988042.1988046.
13. Dénès M., Hritcu C., Lampropoulos L., Paraskevopoulou Z., Pierce B.C. QuickChick: Property-based testing for Coq. *Proc. the Coq Workshop*, 2014, vol. 125, 126 p.
14. Морозов А.В., Панамарев Г.Е. Исследование технологий повышения доверия к специальному программному обеспечению с применением инструментальных средств, реализующих фаззинг-тестирование // *Изв. РАН. 2022. № 3. С. 129–136.*
15. Маклейн С. Категории для работающего математика; [пер. с англ.]. М.: Физматлит, 2004. 352 с.
16. Pierce B.C. Basic category theory for computer scientists. The Mit Press Publ., 1991, 113 p.
17. de Bruijn N.G. Lambda calculus notation with nameless dummies, a tool for automatic formula manipulation, with application to the Church-Rosser theorem. *Proc. Indagationes Mathematicae*, 1972, vol. 75, no. 5, pp. 381–392.
18. Kiselyov O. Typed tagless final interpreters. In: *LNTCS. Generic and Indexed Programming*, 2012, vol. 7470, pp. 130–174. doi: 10.1007/978-3-642-32202-0\_3.
19. Carette J., Kiselyov O., Shan Ch.-Ch. Finally tagless, partially evaluated: Tagless staged interpreters for simpler typed languages. *J. of Functional Programming*, 2009, vol. 19, no. 5, pp. 509–543. doi: 10.1017/S0956796809007205.

Software &amp; Systems

doi: 10.15827/0236-235X.149.065-076

2025, 38(1), pp. 65–76

### A system of verifiable software component specifications with embedding and extraction

Pavel A. Shapkin <sup>1</sup>✉

<sup>1</sup> National Research Nuclear University “MEPhI”,  
Moscow, 115409, Russian Federation

#### For citation

Shapkin, P.A. (2025) ‘A system of verifiable software component specifications with embedding and extraction’, *Software & Systems*, 38(1), pp. 65–76 (in Russ.). doi: 10.15827/0236-235X.149.065-076

#### Article info

Received: 06.03.2024

After revision: 12.07.2024

Accepted: 24.07.2024

**Abstract.** This paper focuses on specification and verification of software systems and their components. It researches a unified specification language that correlates with both random testing systems and static verification tools based on type systems. A variety of programming languages, configuration systems, deployment and other tools require developers to make efforts to integrate them. Verifiable component specifications help to simplify the task. The paper proposes an approach to a unified specification representation integrated with systems for both static type checking and dynamic testing. This solution relies on methods of applicative computing and type theory. It is a conceptual framework for building specifications embedded in various software environments. The lack of static verification capabilities due to limited type systems is eliminated by dynamic testing to some extent. The author implements testing by interpreting specifications into definitions for property-based random testing systems. The practical significance of the proposed approach is automation of the process of constructing typed wrappers, or facades, which are essential for using components from less typed environments

in programming languages with more expressive type systems. The approach automates both the verification of such wrappers and the methods of their construction by defining specification refinement operations. In practice, this allows detecting errors in typing of third-party components at early development stages. The paper gives examples of program specifications with side effects. A basis for specifications is category theory formalizations. The author also analyzes approaches to translating specifications into other representations and to iteratively improving specifications by transforming them.

**Keywords:** software systems, verifiable, verification, semantics, logic, typing

### References

1. Shapkin, P. (2022) ‘Automation of configuration, initialization and deployment of applications based on an algebraic approach’, *Procedia Comput. Sci.*, 213, pp. 785–792. doi: 10.1016/j.procs.2022.11.135.
2. Pierce, B.C., de Amorim, A.A., Casinghino, Ch., Gaboardi, M. et al. (2018) ‘Logical foundations’, *Software Foundations Ser.*, 1, available at: <https://www.seas.upenn.edu/~cis5000/current/sf1f-current/index.html> (accessed May 11, 2024).
3. Methni, A., Lemerre, M., Hedia, B.B. et al. (2014) ‘An approach for verifying concurrent C programs’, *Proc. 8th Junior Researcher Workshop on Real-Time Computing*, pp. 33–36.
4. Lau, H., Nestmann, U. (2011) ‘Java goes TLA+’, *Proc. V Int. Conf. on Theoretical Aspects of Software Eng.*, pp. 117–124. doi: 10.1109/TASE.2011.44.
5. Filliâtre, J.-C., Pascutto, C. (2021) ‘Ortac: Runtime assertion checking for OCaml (tool paper)’, in *LNPSE. Proc. RV*, 12974, pp. 244–253. doi: 10.1007/978-3-030-88494-9\_13.
6. Barendregt, H.P. (1981) *The Lambda Calculus. Its Syntax and Semantics*. NY: North-Holland Publ. Company, 654 p. (Russ. ed.: (1985) Moscow, 606 p.).
7. Wolfengagen, V.E. (2003) *Combinatory Logic in Programming*. 347 p. (Russ. ed.: (2008) Moscow, 204 p.).
8. Barendregt, H. (1991) ‘Introduction to generalized type systems’, *J. of Functional Programming*, 1(2), pp. 125–154. doi: 10.1017/S0956796800020025.
9. Sørensen, M.H., Urzyczyn, P. (2006) ‘Lectures on the Curry-Howard isomorphism’, in *Studies in Logic and the Foundations of Mathematics*, 149, pp. 1–442.
10. Heyting, A. (1931) ‘Die intuitionistische Grundlegung der Mathematik’, *Erkenntnis*, 2, pp. 106–115. doi: 10.1007/BF02028143.
11. Kolmogorov, A.N. (1985) *Selected Proc. Mathematics and Mechanics*. Moscow, 476 p. (in Russ.).
12. Claessen, K., Hughes, J. (2011) ‘QuickCheck: A lightweight tool for random testing of Haskell programs’, *ACM SIGPLAN Notices*, 64(4), pp. 53–64. doi: 10.1145/1988042.1988046.
13. Dénès, M., Hritcu, C., Lampropoulos, L., Paraskevopoulou, Z., Pierce, B.C. (2014) ‘QuickChick: Property-based testing for Coq’, *Proc. the Coq Workshop*, 125, 126 p.
14. Morozov, A.V., Panamarev, G.E. (2022) ‘Research of technologies for increasing confidence in special software using tools that implement fuzzing testing’, *Proc. RARAS*, (3), pp. 129–136 (in Russ.).
15. Mac Lane, S. (1998) *Categories for the Working Mathematician*. Springer Publ., 315 p. (Russ. ed.: (2004) Moscow, 352 p.).
16. Pierce, B.C. (1991) *Basic Category Theory for Computer Scientists*. The Mit Press Publ., 113 p.
17. de Bruijn, N.G. (1972) ‘Lambda calculus notation with nameless dummies, a tool for automatic formula manipulation, with application to the Church-Rosser theorem’, *Proc. Indagationes Mathematicae*, 75(5), pp. 381–392.
18. Kiselyov, O. (2012) ‘Typed tagless final interpreter’, in *LNTCS. Generic and Indexed Programming*, 7470, pp. 130–174. doi: 10.1007/978-3-642-32202-0\_3.
19. Carette, J., Kiselyov, O., Shan, Ch.-Ch. (2009) ‘Finally tagless, partially evaluated: Tagless staged interpreters for simpler typed languages’, *J. of Functional Programming*, 19(5), pp. 509–543. doi: 10.1017/S0956796809007205.

### Авторы

**Шапкин Павел Александрович**<sup>1</sup>,  
к.т.н., доцент, pashapkin@mephi.ru

### Authors

**Pavel A. Shapkin**<sup>1</sup>, Cand. of Sci. (Engineering),  
Associate Professor, pashapkin@mephi.ru

<sup>1</sup> Национальный исследовательский ядерный университет «МИФИ», г. Москва, 115409, Россия

<sup>1</sup> National Research Nuclear University “MEPhI”,  
Moscow, 115409, Russian Federation

## Имитационное моделирование систем физической защиты в среде АКИМ

**Ю.Б. Сениченков**<sup>1</sup>, **И.К. Шарков**<sup>2</sup>✉

<sup>1</sup> Санкт-Петербургский политехнический университет Петра Великого,  
г. Санкт-Петербург, 195251, Россия

<sup>2</sup> Компания ПЕНТАКОН, г. Санкт-Петербург, 197198, Россия

### Ссылка для цитирования

Сениченков Ю.Б., Шарков И.К. Имитационное моделирование систем физической защиты в среде АКИМ // Программные продукты и системы. 2025. Т. 38. № 1. С. 77–88. doi: 10.15827/0236-235X.149.077-088

### Информация о статье

Группа специальностей ВАК: 2.3.1

Поступила в редакцию: 02.05.2024

После доработки: 23.07.2024

Принята к публикации: 13.08.2024

**Аннотация.** В статье обсуждается методика построения имитационных моделей в отечественном программном комплексе АКИМ. Модели предназначены для решения задачи анализа защищенности существующих и проектируемых систем физической защиты объектов и для формирования оценок их эффективности с помощью статистического эксперимента. Приводится обзор существующих современных подходов к решению аналогичной задачи. Большинство из них применяют марковские цепи для поиска уязвимых путей и графы атак и защит для оценки эффективности системы. В качестве альтернативы предлагается строить имитационную модель без построения графа атак и защит, опираясь только на план системы физической защиты. Модель в среде АКИМ создается из экземпляров базовых классов, моделирующих реальные элементы системы физической защиты, в результате чего возникает план, по которому будут двигаться модели агентов и охранников, имитируя реальные атаки. Подход позволяет подробно описывать функции, реакции и возможности системы на уровне ее элементов и задавать фактические параметры нарушителей и охранников, что обеспечивает точность и полноту анализа без упрощений или исключения важных деталей. На демонстрационных примерах показано, что оценки эффективности моделей защиты систем, полученные с помощью программного комплекса АКИМ, близки к оценкам эффективности моделей систем, построенных с помощью марковских цепей. При этом рассматриваемый метод построения имитационных моделей позволяет преодолевать трудности, связанные с использованием марковских цепей: необходимость использовать экспертные оценки коэффициентов матрицы переходов, матрицы большого размера, сложности модификации модели.

**Ключевые слова:** имитационное моделирование, система физической защиты, агентное моделирование, цепи Маркова, оценка эффективности, атака нарушителя, алгоритмы поиска пути

**Благодарности.** Автор выражает глубокую благодарность и признательность своему научному руководителю доктору технических наук Юрию Борисовичу Сениченкову, соавтору, безмерно ушедшему в октябре 2024 года

**Введение.** Обеспечение физической безопасности на территориях объектов повышенной важности, таких как аэропорты и топливно-энергетические комплексы, является важной задачей. Для ее выполнения применяются системы физической защиты (СФЗ), включающие инженерно-технические и человеческие ресурсы для защиты от различных угроз. Эффективность СФЗ определяется такими критериями, как вероятность обнаружения и нейтрализация нарушителей.

Современным способом оценки эффективности систем является компьютерное моделирование. Различные методики построения компьютерных моделей СФЗ и проведения статистических экспериментов с ними позволяют количественно оценить защищенность объектов, описать сильные и слабые стороны системы с помощью различных измеряемых характеристик, что способствует принятию обоснованных решений при проектировании или модификации СФЗ.

Задачи создания моделей, учитывающих все значимые особенности защищаемых объектов, и построения реалистичных сценариев атак и защит являются ключевыми в области оценки эффективности существующих или проектируемых СФЗ.

### Методы построения моделей СФЗ

В основе многих современных методик оценки эффективности систем защиты периметра заложены требования МАГАТЭ, учтенные в методике EASI (США, 1977). Методика EASI предлагает рассматривать только один заданный сценарий внешнего воздействия на систему, в котором последовательность событий (атака, обнаружение, локализация нарушителя и его захват) сопровождается последовательностью вычислений, базирующихся на вероятностях обнаружения нарушителя, на правильной реакции и своевременном прибытии охраны.

Главные недостатки этого подхода – наличие только одного сценария атаки и заданные экспертом значения исходных параметров модели (вероятностей событий). Они не дают возможность достоверно оценить эффективность защиты всего объекта.

Подходы к преодолению указанных недостатков можно разбить на несколько групп по способу формирования моделей из плана охраняемого объекта. Эти модели позволяют создавать множество различных сценариев атак и защит и уменьшать влияние субъективного мнения эксперта на основе

- формирования графа атак и защит вручную экспертом;
- параметрического чертежа (карты) системы и соответствующего ему графа атак;
- параметрического чертежа, на котором разыгрываются различные варианты атак агентов и защиты, без использования явно заданного графа атак и защит.

Перечисленные подходы к формированию моделей используются в программных комплексах различного назначения, преимущественно для оценки существующих систем и реже для их проектирования.

**Модели на основе заданного экспертом графа.** Представление модели системы защиты на основе графа атак является классическим решением. За основу берется предложенный экспертом граф атак [1], построенный для упрощенной системы, и строятся все возможные пути атак. Для оценки эффективности системы чаще всего применяется методика EASI для каждого пути и автоматически строится оценка для всей системы.

Для повышения достоверности оценки необходимо также применять методики и алгоритмы анализа СФЗ, которые способны имитировать действия агентов-нарушителей, охранников и операторов при взаимодействии с системой и друг с другом.

Существуют примеры имитации взаимодействия связи между охранниками и операторами безопасности, в которых применяется метод [2] на основе модифицированной методики EASI. Этот метод, в свою очередь, не учитывает человеческий фактор, связанный с взаимодействием атакующих нарушителей и охранников. Попытка учесть человеческий фактор описана в работах [3, 4], авторы которых использовали генетические алгоритмы для имитации действий нарушителя и нечеткие множества описания допустимых параметров.

В работах [5, 6] предлагается формировать возможные пути атак с учетом базового типа нарушителя, созданного в соответствии с категорией объекта (ГОСТ Р 78.36.032-2013).

Первый (и главный) недостаток оценки эффективности системы по созданному вручную графу заключается в том, что приходится упрощать структуру моделируемой системы. Это приводит к появлению упрощенных сценариев атак, а упрощения непосредственно влияют на достоверность результатов оценки.

Второй недостаток – не всегда понятно, как на основе модели, предложенной экспертом, построить план реальной системы (чертеж), если речь идет о проектировании новых систем. В работах [7] предлагается строить план по уже существующему графу атак и оснащения модели инженерно-техническими средствами. Однако при данном подходе главный недостаток – синтез реальной сложной системы на основе упрощенной модели графа атак, не учитывающей все возможные реальные пути атак и зависящей от субъективного мнения эксперта, – остается. С организационной точки зрения требуются подготовленный специалист для построения модели (графа) и проектировщик, рисующий чертеж и проверяющий систему на соответствие нормативным требованиям, что разделяет всю работу по анализу, оценке эффективности и синтезу планируемой системы на два сложных этапа.

Третий недостаток – сложность описания крупномасштабных систем, где количество вероятностных переходов и состояний будет огромным, а следовательно, при создании такой модели потребуются значительные усилия.

**Модели на основе параметрического плана и графа атак.** В качестве первоначальной информации об объекте предлагается использовать его параметрический план. По этому плану создается граф атак: автоматически с использованием специальной объектно-ориентированной модели системы (*Building Information Model, BIM*) и специальных алгоритмов [8] или при помощи промежуточной модели атак в виде сетки перемещения [9, 10], наложенной на план объекта.

В обоих случаях при каждом значимом изменении структуры системы приходится заново строить граф, что повышает трудоемкость проектирования.

Недостатком моделей на основе BIM является зависимость от стороннего ПО, поэтому на практике чаще применяются подходы с наложением сетки. Они реализованы в отече-

ственных программных продуктах ВЕГА-2, ИТЕРАЦИЯ-СФЗ и зарубежном IPAD.

Применение модели на основе сетки перемещения требует использования специальных эвристических алгоритмов поиска пути, что затрудняет моделирование систем с большой площадью и сложностью. В таком случае опять же приходится упрощать структуру графа (например, увеличивать размер ячеек сетки) и искажать траектории движения нарушителя и охранника, что приводит к очередному упрощению сценариев атак и влияет на достоверность оценки эффективности системы защиты.

**Модели на основе карты (параметрического чертежа) и имитации атак и защиты агентов.** Публикаций, посвященных этому методу и его реализациям в современном ПО, довольно мало.

План реализуется в виде двухмерной или трехмерной карты с пространством для имитации перемещения агентов атаки и защиты. Система физической защиты является иерархической моделью, каждый компонент которой представлен в виде ее базовых элементов (датчиков, камер, ограждений и т.д.) со своими уникальными параметрами, а агенты своими уникальными моделями.

Вместо построения графа атак и анализа всех его путей разрабатываются и применяются алгоритмы, имитирующие движение нарушителей и охранников по карте системы. Эффективность системы защиты определяется с помощью статистических экспериментов.

В работе [11] имитационная модель строится на основе игровой модели с применением стороннего ПО (ARMA3), в котором осуществляется имитация атак на объекты ядерной энергетики. В [12] обсуждаются возможности построения имитационных моделей на основе теории игр.

**Предлагаемый подход.** Способ построения цифрового двойника системы защиты и графический язык моделирования, позволяющий создавать чертеж проектируемой системы и проводить статистические эксперименты, имитируя проникновение агентов-нарушителей и противодействие агентов-защитников, описаны в [13, 14].

Данный подход отличается от описанных выше прежде всего тем, что не использует заранее заданный или автоматически построенный граф атак, а требует только чертеж исследуемой системы, включающий существующие, библиотечные, компоненты: модели устройств, созданных на основе их паспортных данных,

модели охранников и операторов с алгоритмами имитации их движения.

В отличие от подхода, предлагаемого в [7], процесс построения модели системы начинается с проектирования чертежа и не требует участия специально обученного специалиста в области моделирования. Предлагаемый подход помогает найти уязвимые места системы и улучшить ее свойства. Для этого достаточно изменить чертеж (исходную модель) и повторить статистический эксперимент.

Подход на основе марковских цепей используется многими авторами [15–17], поэтому сравним его с методом, применяемым в среде АКИМ.

### Построение имитационной модели СФЗ на базе графов атак

Граф атак можно использовать как основу имитационной модели, в которой каждая возможная атака (сценарий атаки) реализуется с помощью случайного выбора ребра при движении от текущего узла до следующей локальной цели. Новые сценарии формируются и реализуются до тех пор, пока оценки вероятности задержания агента и уничтожения объекта не будут вычислены с необходимой точностью. Такой подход позволяет избежать решения уравнений, однако сами матрицы переходов приходится строить и хранить даже при использовании имитационной модели.

Приведем пример использования имитационной модели на базе графа атак, реализованной в среде AnyDynamics (<https://www.mvstudium.com/advan.htm>).

Моделируемая СФЗ объекта, окруженного стеной с одним КПП, представляется графом атак (<http://www.swsys.ru/uploaded/image/2025-1/5.jpg>), который реализуется в виде карты поведения (упрощенной машины состояний UML), формирующей очередной сценарий, с помощью матрицы переходов (рис. 1).

Строки матрицы соответствуют следующим состояниям: S1 – начальное состояние, где случайным образом осуществляется выбор атаки через КПП (S2) или нескольких вариантов преодоления стены (S3, S4, S5); S6, S7 – вспомогательные состояния, позволяющие сократить число путей в S8 (захват охраной агента) и S9 (достижение агентом цели (проигрыш защиты)).

Имитационная модель на языке карт поведения представлена циклом (<http://www.swsys.ru/uploaded/image/2025-1/6.jpg>), многократно повторяющимся (Nexp) отдельный эксперимент.

$$A = \begin{matrix} \begin{pmatrix} 0 & \frac{1}{20} & \frac{3}{20} & \frac{1}{5} & \frac{3}{5} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{10} & \frac{9}{10} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{3}{10} & \frac{7}{10} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} & \begin{matrix} -S1 \\ -S2 \\ -S3 \\ -S4 \\ -S5 \\ -S6 \\ -S7 \\ -S8 \\ -S9 \end{matrix} \end{matrix}$$

Рис. 1. Матрица переходов (параметр  $A$  в модели AnyDynamics), где состояния  $S2-S5$  имеют безусловные переходы к другим, а  $S8, S9$  конечные

Fig. 1. Transition matrix (parameter  $A$  in AnyDynamics model), where states  $S2-S5$  have unconditional transitions to others and  $S8, S9$  are finite

Можно непосредственно решить уравнения, одновременно реализовать имитационную модель и увидеть, что результаты вычислений, как и следовало ожидать, совпадают до третьего знака.

### Имитация движения нарушителя по плану СФЗ без заданного графа

Имитационную модель можно создать, не строя граф атак и защит, опираясь только на план СФЗ. Такой подход реализован в комплексе экспертного моделирования АКИМ. Подход позволяет подробно описывать функции, реакции и возможности системы на уровне ее элементов и задавать фактические параметры нарушителей и охранников, что обеспечивает точность и полноту анализа без упрощений или исключения важных нюансов. Это особенно полезно для проектирования систем, так как созданный цифровой двойник объекта служит одновременно графическим образом и параметрическим планом в виде чертежа для проектной документации.

Модель в АКИМ строится из экземпляров базовых классов, моделирующих реальные элементы СФЗ, в результате чего возникает план, по которому будут двигаться модели агентов и охранников, имитируя реальные атаки (<http://www.swsys.ru/uploaded/image/2025-1/7.jpg>). Классы могут использовать базовые элементы средств защиты, например, извещатели (датчик), видеокамеры, ограждения и т.д.

Модели средств защиты содержат параметры, соответствующие техническим паспортам устройств, и их геометрическую форму зон воздействия с привязкой к плану СФЗ.

Имитация атак осуществляется с помощью алгоритма, прокладывающего локальные отрезки пути по плану от текущей точки до следующей достижимой, и алгоритма, учитывающего влияние событий на форму выбранного локального пути.

Для прокладки локального отрезка пути используется эвристический алгоритм поиска пути между двумя точками на плане с учетом видимых препятствий. Для выбора варианта обхода отдельного непреодолимого препятствия на пути атаки используется локальный граф видимости. Под ним подразумевается граф путей обхода одного препятствия с учетом его видимых вершин. Такой граф автоматически строится для каждого встреченного препятствия, и с его помощью строится кратчайший маршрут обхода.

Алгоритм формирования событийно-управляемых траекторий расставляет случайные события на локальном отрезке пути, и возникает событийно-управляемая траектория, формируемая с помощью гибридных автоматов. Это позволяет автоматически строить динамические сценарии экспериментов и исключать необходимость привлечения эксперта на этапе подготовки модели СФЗ. Случайные события влияют на выбор агентом очередной достижимой точки (смена локального отрезка пути под воздействием событий), что вносит случайные изменения в сценарий.

Взаимодействие прокладчика пути и алгоритма формирования событийно-управляемых траекторий можно представить в виде схемы (рис. 2).

Сценарии разыгрываются многократно, что позволяет собирать подробную статистику обо всех событиях и моделируемых значениях в ходе каждого испытания. С помощью доверительных интервалов определяется критерий для ограничения количества испытаний в серии, что обеспечивает достоверную оценку эффективности СФЗ:

- вероятность обнаружения с помощью ТСО ( $P_{Обн}$ );
- вероятность нейтрализации нарушителя ( $P_{Нейтр}$ );
- наиболее уязвимые траектории (пути и координаты);
- время задержки нарушителя на ИСО ( $t$ ) и другие.

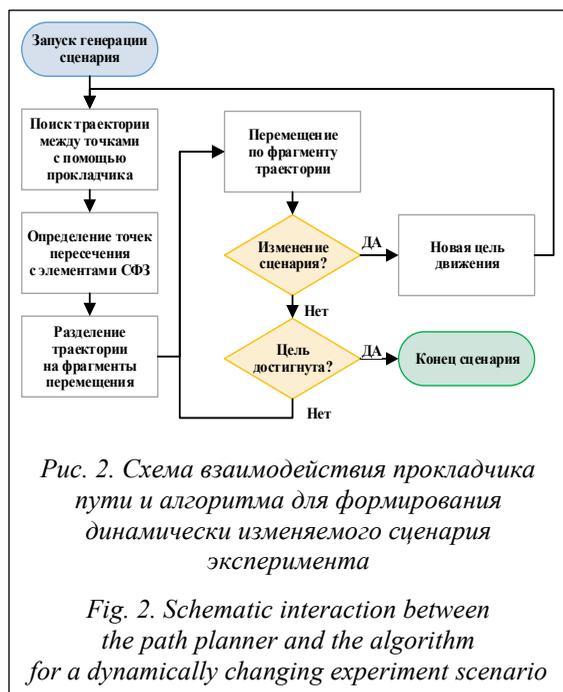


Рис. 2. Схема взаимодействия прокладчика пути и алгоритма для формирования динамически изменяемого сценария эксперимента

Fig. 2. Schematic interaction between the path planner and the algorithm for a dynamically changing experiment scenario

**Сравнение подходов:  
одномерная модель системы**

Сравним различные модели СФЗ на основе марковских цепей и моделей, реализованных в среде АКИМ. Рассмотрим одномерную модель СФЗ (последовательные события), показывающую путь нарушителя через зону досмотра в аэропорту при посадке на самолет.

Обычно одномерные модели не принято моделировать с использованием марковских цепей. Для них существует международная методика МАГАТЭ EASI. Эта методика оценивает на одном заданном пути атаки вероятность обнаружения и своевременного прибытия охранников (нейтрализации) к месту тревоги.

Одномерную модель можно реализовать в виде простой карты поведения (<http://www.swsys.ru/uploaded/image/2025-1/8.jpg>).

Результаты моделирования показали, что вероятность успешной защиты СФЗ составила 0.844, а вероятность провала – 0.155 (<http://www.swsys.ru/uploaded/image/2025-1/9.jpg>).

Помимо оценки вероятностей успешной защиты и ее провала, среда АКИМ позволяет – рисовать чертеж СФЗ, учитывающий реальную геометрию объекта (расположение всех его базовых элементов средств охраны в пространстве на пути атаки);

– прокладывать траекторию движения агента по созданному чертежу, представляющему собой цифровой двойник реального пространства СФЗ со всеми ее характеристиками;

– автоматически формировать характеристики для заданной траектории атаки (длину пути, список препятствий/датчиков/видеокамер на пути с учетом сложности их преодоления);

– осуществлять движение агента по сформированной траектории с учетом реальных характеристик пути, имитируя реальное движение агента; реальные характеристики движения агента (скорость, время преодоления препятствия и т.д.) формируются на основе выбираемых проектировщиком законов распределения (<http://www.swsys.ru/uploaded/image/2025-1/10.jpg>), причем случайные характеристики, связанные с людьми, могут определяться PERT-распределением (четырёхпараметрическим бета-распределением).

Рассматриваемая одномерная модель в АКИМ представлена на рисунке 3.

Модель атаки в среде АКИМ позволяет провести эксперимент, имитирующий реальное движение агента: агент движется со случайной скоростью на всех фрагментах пути атаки, затрачивая время не только на движение, но и на преодоление препятствий и взаимодействуя с зонами обнаружения технических средств охраны.

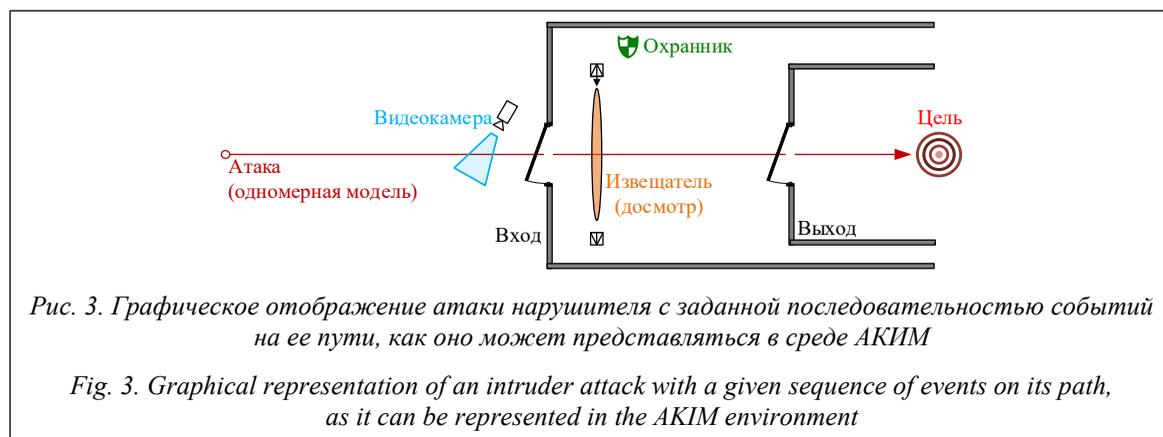


Рис. 3. Графическое отображение атаки нарушителя с заданной последовательностью событий на ее пути, как оно может представляться в среде АКИМ

Fig. 3. Graphical representation of an intruder attack with a given sequence of events on its path, as it can be represented in the AKIM environment

Аналогично моделируется реакция охранника, который, как и в реальной жизни, должен вовремя отреагировать на нарушение и приблизиться к атакующему агенту. Успешная защита объекта возможна только в случае, если время движения охранника к нарушителю ( $T_{Oxp}$ ) меньше времени движения агента ( $T_{Нар}$ ) до цели [11].

Параметры модели в среде АКИМ:

- скорость нарушителя (от 1 до 12 км/ч);
- путь нарушителя от точки старта до цели (244 м);
- вероятность обнаружения нарушения через видеокамеру ( $P = 0.4$ );
- вероятность обнаружения нарушения через извещатель ( $P = 0.95$ );
- время на преодоление входных и выходных ворот (от 30 до 300 сек.);
- вероятность правильной реакции охранника ( $P = 0.9$ );
- время принятия решения охранником (от 1 до 30 сек.);
- координаты охранника относительно пути атаки;
- скорость движения охранника по направлению к нарушителю (от 8 до 20 км/ч).

Результаты моделирования в среде АКИМ для 1 000 испытаний дали 957 успешных нейтрализаций нарушителя, что попадает в доверительный интервал от 0.9444 до 0.9696. Точечная вероятность составила 95.7 %, а обнаруживающая способность системы технических средств охраны – около 98.5 %.

Оценки вероятностей исходов для марковской цепи и имитационной модели среды АКИМ различаются на 11.3 %.

### Сравнение подходов: двухмерная модель системы

Рассмотрим двухмерную модель СФЗ охраняемого объекта с двумя рубежами охраны и тремя охраняемыми КПП и сравним результаты моделирования с помощью марковской цепи и имитационной модели АКИМ.

Граф атак для марковской модели с использованием карты состояний (<http://www.swsys.ru/uploaded/image/2025-1/11.jpg>) даже для такой простой системы оказывается трудновоспринимаемым. Матрица переходов (рис. 4) разреженная, размерности 10.

Как имитационная модель для марковской цепи в виде графа атак, так и решение уравнений дают следующие оценки: вероятность успешной защиты системы (нарушитель пойман)

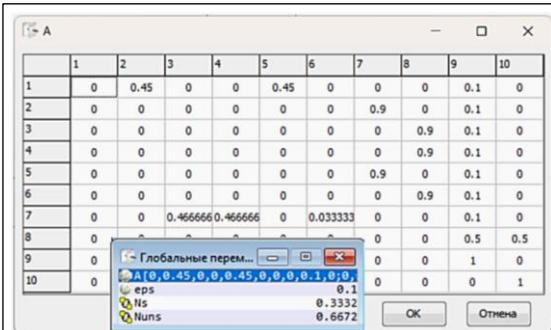


Рис. 4. Матрица переходов и результаты:  $N_s$  – провал защиты (нарушитель достиг цели),  $N_{uns}$  – успешная защита (атакующий агент пойман), в конце – поглощающие состояния, соответствующие удачной и неудачной атакам

Fig. 4. Transition matrix and results:  $N_s$  – defense failure (the intruder reached the target),  $N_{uns}$  – successful defense (the attacker is caught), at the end – absorbing states corresponding to successful and unsuccessful attacks

составила 0.667, вероятность провала защиты (нарушитель достиг цели) – 0.333.

Имитационная модель среды АКИМ для этого плана представлена на рисунке 5, где показан процесс построения СФЗ с помощью базовых компонентов.

Для проведения экспериментов в среде АКИМ необходимо указать место появления атакующих агентов (<http://www.swsys.ru/uploaded/image/2025-1/12.jpg>). Это можно сделать различными способами, например: линию вокруг объекта (окружность, многоугольник или отрезок) разбить на участки, где может появиться агент, и случайным образом выбрать точку появления.

*Первое отличие от марковской модели:* вместо ограниченного количества сценариев, порождаемых графом путей для марковской модели, имитационная модель АКИМ строит уникальные пути атак и защиты, порождаемые случайным образом ранее описанными алгоритмами (рис. 2).

Результат построения таких уникальных путей атак из одной точки появления агента в разных экспериментах проиллюстрирован на рисунке 6.

Результаты моделирования по участкам возникновения нарушителя представлены в таблице 1.

Оценки вероятностей исходов для марковских моделей и имитационной модели АКИМ отличаются на 11.4 %. При этом на участке 11

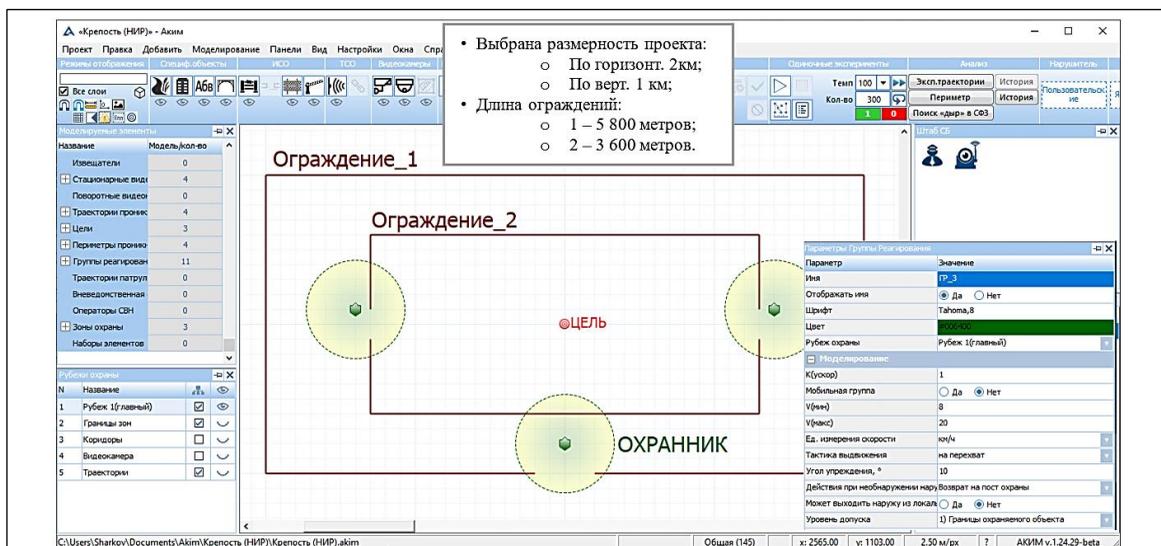


Рис. 5. Структура СФЗ в виде чертежа из базовых элементов, представленных отдельными моделями (ограждения, охранники, цель)

Fig. 5. Physical protection systems structure as a drawing of basic elements represented by separate models (fences, guards, target)

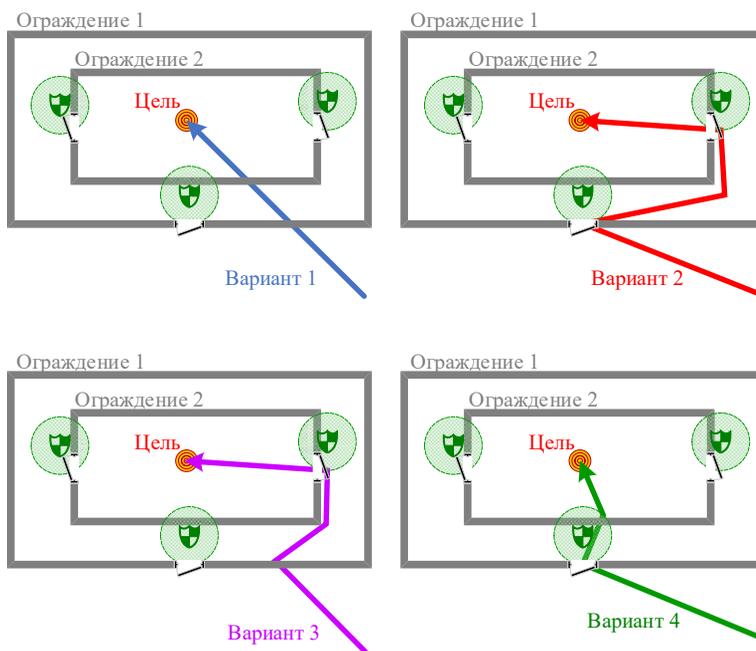


Рис. 6. Примеры построения случайных траекторий атак

Fig. 6. Examples of building random attack paths

оценка эффективности совпадает с марковской моделью (рис. 4).

Второе отличие от марковской модели: среда АКМ предлагает описывать агента-нарушителя набором реальных количественных характеристик, влияющих на имитацию движения и, следовательно, на результаты мо-

делирования. Такие характеристики может выбирать разработчик (рис. 7), создавая те или иные модели нарушителей, что недоступно для других методик.

Можно сравнить результаты моделирования с нарушителем «Дилетант» (не применяет специальные приспособления, облегчающие

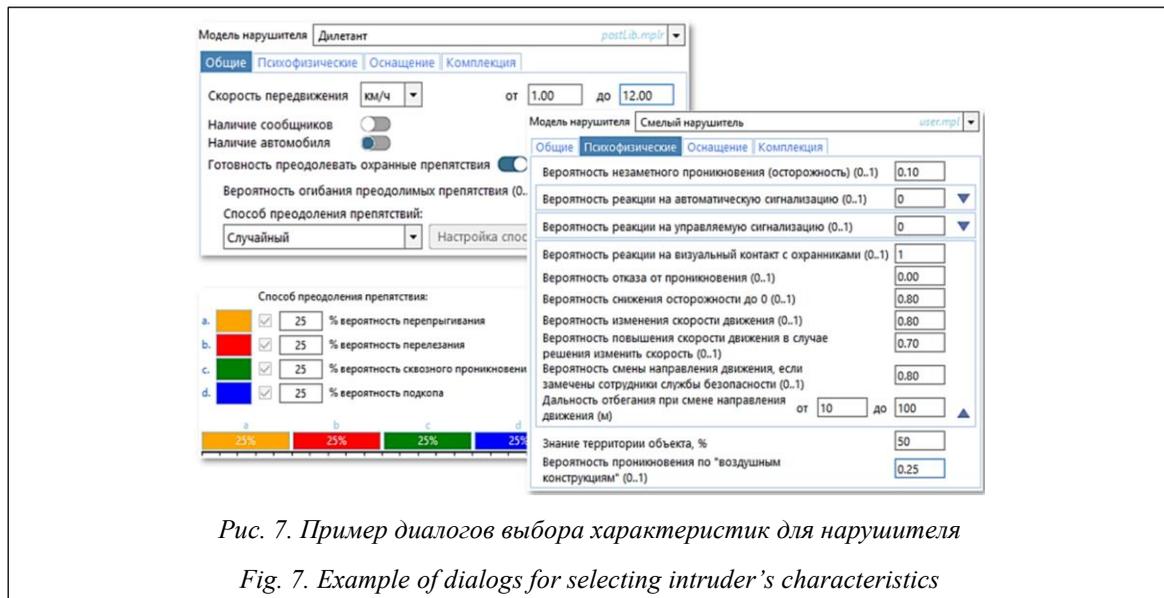


Рис. 7. Пример диалогов выбора характеристик для нарушителя

Fig. 7. Example of dialogs for selecting intruder's characteristics

ему проникновение, и готов отказаться от проникновения при виде охранника) (табл. 1) с результатами моделирования с «Нарушителем, применяющим силовой метод проникновения» (использует подручные средства и экипировку при атаке, не отказывается от проникновения в случае тревоги на объекте и обнаружения охранника) (табл. 2). Оценка вероятности нейтрализации для агента «Дилетант» (0.781) уменьшилась в полтора раза по сравнению с «Нарушителем, применяющим силовой метод проникновения» (0.473).

Для изменения модели поведения агента в марковской модели эксперту необходимо пе-

реопределять значения вероятностей в матрице переходов, опираясь на свой личный опыт, или прибегать к описанным в обзорном разделе методикам.

*Третье отличие от марковской модели:* среда АКИМ легко справляется с проблемой масштабирования размеров СФЗ (<http://www.swsys.ru/uploaded/image/2025-1/13.jpg>) и изменения ее структуры.

Чтобы оценить отличающийся по масштабам или пропорциям объект с помощью марковской цепи, потребуется переопределение значений вероятностей в матрице переходов. В среде АКИМ меняются только размеры эле-

Таблица 1

Результаты моделирования с моделью нарушителя «Дилетант»

Table 1

Simulation results with the “Amateur” intruder model

Участок	Количество испытаний	Количество успешных испытаний	Доверительный интервал
1	345	252	0.6836–0.7773
2	345	253	0.6867–0.7800
3	344	276	0.7602–0.8444
4	344	269	0.7383–0.8256
5	345	267	0.7298–0.8181
6	344	312	0.8763–0.9377
7	344	294	0.8174–0.8919
8	344	251	0.6827–0.7766
9	345	277	0.7609–0.8449
10	344	287	0.7950–0.8736
11	345	243	0.6562–0.7525
12	346	249	0.6723–0.7670

*Примечание:* минимальная вероятность поимки нарушителя составила 70.4 % (участок 1), общая вероятность поимки для всей СФЗ – 78.1 %.

ментов на карте, а все экспертные характеристики и данные из документации устройств остаются неизменными.

В таблицах 1 («Дилетант») и 2 («Нарушитель, применяющий силовой метод проникновения») приведены результаты оценки эффективности для территории СФЗ площадью 1.99 км<sup>2</sup>,

в таблице 3 – для территории площадью около 0.2 км<sup>2</sup> (агент «Дилетант»).

Вероятность нейтрализации нарушителя для СФЗ площадью 0.2 км<sup>2</sup> значительно выше, чем в предыдущих случаях, как и следовало ожидать.

Необходимо отметить, что реальная система будет во много раз сложнее, чем в пред-

Таблица 2

**Результаты моделирования с моделью «Нарушитель, применяющий силовой метод проникновения»**

Table 2

**Simulation results with the model "Intruder using forced penetration"**

Участок	Количество испытаний	Количество успешных испытаний	Доверительный интервал
1	1 000	485	0.4540–0.5160
2	1 000	454	0.4231–0.4849
3	1 000	528	0.4971–0.5589
4	1 000	495	0.4640–0.5260
5	1 000	456	0.4251–0.4869
6	1 000	433	0.4023–0.4637
7	1 000	420	0.3894–0.4506
8	1 000	451	0.4202–0.4818
9	1 000	478	0.4470–0.5090
10	1 000	542	0.5111–0.5729
11	1 000	463	0.4321–0.4939
12	1 000	466	0.4351–0.4969

*Примечание:* минимальная вероятность обезвреживания – 42.0 % (участок 7), общая вероятность обезвреживания – 47.3 %.

Таблица 3

**Результаты для моделирования с моделью нарушителя «Дилетант» на объекте площадью 0.2 кв. км**

Table 3

**Results for simulations with the "Amateur" intruder model at a 0.2 square kilometer site**

Участок	Количество испытаний	Количество успешных испытаний	Доверительный интервал
1	1 000	760	0.7335–0.7865
2	1 000	934	0.9186–0.9494
3	1 000	986	0.9787–0.9933
4	1 000	976	0.9665–0.9855
5	1 000	995	0.9906–0.9994
6	1 000	993	0.9878–0.9982
7	1 000	992	0.9865–0.9975
8	1 000	991	0.9851–0.9969
9	1 000	994	0.9892–0.9988
10	1 000	989	0.9825–0.9955
11	1 000	949	0.9354–0.9626
12	1 000	791	0.7658–0.8162

*Примечание:* минимальная вероятность обезвреживания составила 76.0 % (участок 1), общая вероятность обезвреживания – 94.6 %.

ставленном примере. Даже простейшая система может включать в себя десятки устройств обнаружения с разными характеристиками, множество охранников, сложную топографию пространства перемещения. Для моделирования такой системы с помощью марковских цепей неизбежно приходится упрощать модель, в противном случае усложнение модели приводит к экспоненциальному росту состояний и путей в графе только для одного атакующего агента (без рассмотрения множественных атак).

### Заключение

Предложенный в статье агентный подход имитационного моделирования атак и защиты представляет собой более гибкое и более правдоподобное решение задачи оценки эффективности СФЗ, позволяющее учитывать широкий спектр возможных сценариев атак и характеристик нарушителей. В отличие от подхода с использованием марковских цепей данная имитационная модель не требует явного построения

графа атак. Она дает возможность осуществлять детальную проработку модели, снижает субъективное влияние эксперта на входные данные, повышает объективность оценки и строит более реалистичные сценарии атак.

Таким образом, подход с имитацией движения агентов в цифровом двойнике системы, реализованном в виде реальных рабочих чертежей, является новым перспективным направлением в области планирования и оценки систем физической защиты.

Предлагаемая технология может применяться для проектирования новых и оценки существующих СФЗ, использует реальный план СФЗ без упрощений, легко справляется с масштабированием структуры СФЗ, генерирует сценарии атак автоматически в ходе испытаний. Она может использоваться для моделирования нескольких параллельных атак, для определения наиболее опасных направлений атак и уязвимых траекторий, а также менять тип угроз и модели нарушителей при исследовании уязвимости.

### Список литературы

1. Wely I.Ch.E., Chetaïne A. Analysis of physical protection system effectiveness of nuclear power plants based on performance approach. *Annals of Nuclear Energy*, 2021, vol. 152, art. 107980. doi: 10.1016/j.anucene.2020.107980.
2. Jiwei Z., Shunlong J., Jian L., Zhang L., Huaping C., Xiaofeng L. Optimization of communication probability in effectiveness evaluation of physical protection system. *IEEE Access*, 2020, vol. 8, pp. 228199–228205. doi: 10.1109/ACCESS.2020.3046037.
3. Тарасов А.Д., Шрейдер М.Ю., Боровский А.С. Метод проектирования систем физической защиты на основе генетического алгоритма // *Интеллект. Инновации. Инвестиции*. 2017. № 9. С. 70–74.
4. Chen S., Xiang D., Jin B., Jin H. Vulnerability assessment for physical protection systems of cave temples: A fuzzy petri net approach. *Heliyon*, 2024, vol. 10, no. 12, art. e33100. doi: 10.1016/j.heliyon.2024.e33100.
5. Kostin V., Borovsky A. Definition of basic violators for critically important objects using the information probability method and cluster analysis. *CEUR Workshop Proc. Proc. ITNT*, 2020, vol. 2667, pp. 343–347.
6. Шнякина Е.А., Костин В.Н. Разработка алгоритмического обеспечения принятия решений по идентификации типовых нарушителей категоризируемым объектам // *Изв. высших учебных заведений. Поволжский регион. Технич. науки*. 2023. № 3. С. 72–82. doi: 10.21685/2072-3059-2023-3-5.
7. Костин В.Н. Методика формирования элементов структуры организационного управления систем физической защиты на основе информационного подхода // *Тр. ИСА РАН*. 2020. Т. 70. № 1. С. 30–39. doi: 10.14357/20790279200104.
8. Čakija D., Van Ž., Golub M., Čakija D. Optimizing physical protection system using domain experienced exploration method. *Automatika*, 2020, vol. 61, no. 2, pp. 207–218. doi: 10.1080/00051144.2019.1698192.
9. Setiawan Y.A., Chirayath S.S., Kitcher E.D. MAPPs: A stochastic computational tool for multi-path analysis of physical protection systems. *Annals of Nuclear Energy*, 2020, vol. 137, art. 107074. doi: 10.1016/j.anucene.2019.107074.
10. Yang J., Huang L., Ma H. et al. A 2D-graph model-based heuristic approach to visual backtracking security vulnerabilities in physical protection systems. *IJCIP*, 2022, vol. 38, art. 100554. doi: 10.1016/j.ijcip.2022.100554.
11. Гаврилов П.М., Слюсарев А.В., Момот А.И. и др. Способ оценки эффективности системы физической защиты важного государственного объекта: Свид. о регистр. ПрЭВМ № 2724909С1. Рос. Федерация, 2020.
12. Жилин Р.А., Мельников А.В., Щербакова И.В. К вопросу о модели нарушителя системы безопасности объекта охраны // *Вестн. Воронежского института МВД России*. 2019. № 2. С. 57–69.
13. Шарков И.К. Имитационное моделирование средств физической защиты на охраняемом объекте в задачах оценки уязвимости // *ИМСВН-2020: тр. конф.* 2020. С. 254–264.
14. Шарков И.К., Желудков Е.А. Применимость эвристического алгоритма для задач поиска траекторий движения через систему физической защиты // *SEIM-2019: сб. тр. конф.* 2019. Т. 2. С. 34–40.
15. Костин В.Н. Оценка эффективности физической защиты информации критически важных объектов на основе марковских цепей // *Информационные технологии*. 2019. Т. 25. № 12. С. 757–765. doi: 10.17587/it.25.757-765.
16. Zou B., Li M., Yang M. Vulnerability learning of adversary paths in Physical Protection Systems using AMC/EASI. *Progress in Nuclear Energy*, 2021, vol. 134, art. 103666. doi: 10.1016/j.pnucene.2021.103666.

17. Zou B., Wang W., Liu J. et al. Development of a structure analytic hierarchy approach for the evaluation of the physical protection system effectiveness. *Nuclear Engineering and Technology*, 2020, vol. 52, no. 8, pp. 1661–1668. doi: 10.1016/j.net.2020.01.033.

Software &amp; Systems

doi: 10.15827/0236-235X.149.077-088

2025, 38(1), pp. 77–88

### Simulation modeling of physical protection systems in AKIM environment

Yury B. Senichenkov<sup>1</sup>, Ilya K. Sharkov<sup>1</sup>✉<sup>1</sup> Peter the Great SPbPU, St. Petersburg, 195251, Russian Federation<sup>2</sup> The PENTACON Company, St. Petersburg, 197198, Russian Federation**For citation**

Senichenkov, Yu.B., Sharkov, I.K. (2025) 'Simulation modeling of physical protection systems in AKIM environment', *Software & Systems*, 38(1), pp. 77–88 (in Russ.). doi: 10.15827/0236-235X.149.077-088

**Article info**

Received: 02.05.2024

After revision: 23.07.2024

Accepted: 13.08.2024

**Abstract.** The paper discusses the methodology of building simulation models in the domestic software package AKIM. The models focus on solving the problem of analyzing the security level of existing and designed systems of physical protection of objects. They also use statistical experiment to form estimates of such systems' effectiveness. The authors give a review of existing modern approaches to solving a similar problem. Most approaches apply Markov chains to search for vulnerable paths, as well as attack and defense graphs to assess the system effectiveness. Alternatively, it is suggested to build a simulation model without building an attack and defense graph, relying only on a physical defense system plan. The model in the AKIM environment consists of base class instances that model real elements of a physical protection system. As a result, there is a plan for models of agents and guards to move, simulating real attacks. The approach allows describing in detail the functions, reactions and capabilities of the system at the level of its elements and specifying the actual parameters of intruders and guards. This ensures accuracy and completeness of the analysis without simplification or exclusion of important details. Demonstration examples show that efficiency estimates of system protection models obtained by AKIM software package are close to efficiency estimates of system models built using Markov chains. In this case, the considered method of building simulation models allows overcoming the difficulties associated with using Markov chains: the need to use expert estimates of the coefficients of the transition matrix, large size matrices, the complexity of model modification.

**Keywords:** simulation modeling, physical protection system, agent-based modeling, Markov chains, performance evaluation, intruder attack, path finding algorithms

**Acknowledgements.** The author expresses his deep gratitude and appreciation to his supervisor Yury Borisovich Senichenkov, Doctor of Technical Sciences, co-author, who passed away in October 2024

#### References

1. Wely, I.Ch.E., Chetaine, A. (2021) 'Analysis of physical protection system effectiveness of nuclear power plants based on performance approach', *Annals of Nuclear Energy*, 152, art. 107980. doi: 10.1016/j.anucene.2020.107980.
2. Jiwei, Z., Shunlong, J., Jian, L., Zhang, L., Huaping, C., Xiaofeng, L. (2020) 'Optimization of communication probability in effectiveness evaluation of physical protection system', *IEEE Access*, 8, pp. 228199–228205. doi: 10.1109/ACCESS.2020.3046037.
3. Tarasov, A.D., Shrader, M.Yu., Borovskiy, A.S. (2017) 'Design method for physical protection systems based on the genetic algorithm', *Intellect. Innovations. Investments*, (9), pp. 70–74 (in Russ.).
4. Chen, S., Xiang, D., Jin, B., Jin, H. (2024) 'Vulnerability assessment for physical protection systems of cave temples: A fuzzy petri net approach', *Heliyon*, 10(12), art. e33100. doi: 10.1016/j.heliyon.2024.e33100.
5. Kostin, V., Borovskiy, A. (2020) 'Definition of basic violators for critically important objects using the information probability method and cluster analysis', *CEUR Workshop Proc. ITNT*, 2667, pp. 343–347.
6. Shnyakina, E.A., Kostin, V.N. (2023) 'Development of algorithmic support for decision-making to identify typical violators of categorized objects', *University Proc. Volga Region. Tech. Sci.*, (3), pp. 72–82 (in Russ.). doi: 10.21685/2072-3059-2023-3-5.

7. Kostin, V.N. (2020) 'Formation technique of elements of the structure of organizational management of physical protection systems based on the information approach', *Proc. ISA RAS*, 70(1), pp. 30–39 (in Russ.). doi: 10.14357/20790279200104.
8. Čakija, D., Ban, Ž., Golub, M., Čakija, D. (2020) 'Optimizing physical protection system using domain experienced exploration method', *Automatika*, 61(2), pp. 207–218. doi: 10.1080/00051144.2019.1698192.
9. Setiawan, Y.A., Chirayath, S.S., Kitcher, E.D. (2020) 'MAPPS: A stochastic computational tool for multi-path analysis of physical protection systems', *Annals of Nuclear Energy*, 137, art. 107074. doi: 10.1016/j.anucene.2019.107074.
10. Yang, J., Huang, L., Ma, H. et al. (2022) 'A 2D-graph model-based heuristic approach to visual backtracking security vulnerabilities in physical protection systems', *IJCIP*, 38, art. 100554. doi: 10.1016/j.ijcip.2022.100554.
11. Gavrillov, P.M., Slyusarev, A.V., Momot, A.I. et al. (2020) *Method for Assessing the Effectiveness of the Physical Protection System of an Important State Object*, Pat. RF, № 2724909C1.
12. Zhilin, R.A., Melnikov, A.V., Shcherbakova, I.V. (2019) 'To the question of the intruder model of the security system of the security object', *Bull. of the Voronezh Institute of the Ministry of Internal Affairs of Russia*, (2), pp. 57–69 (in Russ.).
13. Sharkov, I.K. (2020) 'Simulation modeling of physical protection systems in the tasks of vulnerability assessment', *Proc. IMSVN-2020*, pp. 254–264 (in Russ.).
14. Sharkov, I.K., Zheludkov, E.A. (2019) 'Applicability of heuristic algorithm for tasks of search of movement trajectories through physical protection system', *Proc. SEIM-2019*, 2, pp. 34–40 (in Russ.).
15. Kostin, V.N. (2019) 'Assessment of efficiency of physical security of information of crucial objects on the basis of Markov chains', *Information Tech.*, 25(12), pp. 757–765 (in Russ.). doi: 10.17587/it.25.757-765.
16. Zou, B., Li, M., Yang, M. (2021) 'Vulnerability learning of adversary paths in Physical Protection Systems using AMC/EASI', *Progress in Nuclear Energy*, 134, art. 103666. doi: 10.1016/j.pnucene.2021.103666.
17. Zou, B., Wang, W., Liu, J. et al. (2020) 'Development of a structure analytic hierarchy approach for the evaluation of the physical protection system effectiveness', *Nuclear Engineering and Technology*, 52(8), pp. 1661–1668. doi: 10.1016/j.net.2020.01.033.

**Авторы****Сениченков Юрий Борисович**<sup>1</sup>, д.т.н.,

доцент, профессор,

senyb@mail.ru

**Шарков Илья Кириллович**<sup>1</sup>, исследователь,

ведущий разработчик, shark2.1@mail.ru

**Authors****Yury B. Senichenkov**<sup>1</sup>, Dr.Sci. (Engineering),

Associate Professor, Professor,

senyb@mail.ru

**Ilya K. Sharkov**<sup>1</sup>, Researcher,

Leading Developer, shark2.1@mail.ru

<sup>1</sup> Санкт-Петербургский политехнический университет Петра Великого,  
г. Санкт-Петербург, 195251, Россия

<sup>2</sup> Компания ПЕНТАКОН,

г. Санкт-Петербург, 197198, Россия

<sup>1</sup> Peter the Great SPbPU,

St. Petersburg,

195251, Russian Federation

<sup>2</sup> The PENTACON Company, St. Petersburg,

197198, Russian Federation

## Авторская метрика оценки близости программ: приложение для поиска уязвимостей с помощью генетической дезволюции

М.В. Буйневич<sup>1</sup>, К.Е. Израилов<sup>2</sup>✉

<sup>1</sup> Санкт-Петербургский университет Государственной противопожарной службы МЧС России, г. Санкт-Петербург, 196105, Россия

<sup>2</sup> Санкт-Петербургский Федеральный исследовательский центр РАН, г. Санкт-Петербург, 199178, Россия

### Ссылка для цитирования

Буйневич М.В., Израилов К.Е. Авторская метрика оценки близости программ: приложение для поиска уязвимостей с помощью генетической дезволюции // Программные продукты и системы. 2025. Т. 38. № 1. С. 89–99. doi: 10.15827/0236-235X.149.089-099

### Информация о статье

Группа специальностей ВАК: 2.3.6

Поступила в редакцию: 10.06.2024

После доработки: 25.06.2024

Принята к публикации: 28.06.2024

**Аннотация.** Актуальность темы статьи обусловлена наличием в сфере информационной безопасности задач, требующих сравнения программ в их различных представлениях, таких, как текстовый ассемблерный код (например, для поиска уязвимостей или подтверждения авторства). В работе представлена метрика близости двух текстов в виде списка строк из символов, являющаяся развитием ее предыдущей версии. Основным результатом текущего исследования (как части главного, направленного на генетическую дезволюцию программ) являются сама метрика, а также ее характеристики и особенности, выявленные с помощью проведенных экспериментов. Метрика представлена в аналитическом виде, программно реализована на языке Python, принимает на вход два списка символьных строк для сравнения и коэффициенты учета позиции ее элементов от начала списка и последовательности символов. Результатом ее вычисления является числовое значение в диапазоне от 0 до 1. Новизна метрики заключается в достаточно точной и чувствительной оценке близости двух текстов независимо от форматов представления данных; текущая версия метрики отличается от предыдущей учетом указанных коэффициентов. Теоретическая значимость заключается в развитии способов сравнения произвольных текстов, представляющих собой список символьных строк, содержащих информацию, последовательно излагаемую согласно определенной логике (что требует учета позиции). Помимо общего назначения сравнительных инструментов такого рода, практическая значимость метрики состоит в возможности определения близости двух программ, имеющих бинарное представление машинного кода, предварительно преобразованное в текстовое представление ассемблерного кода.

**Ключевые слова:** информационная безопасность, метрика близости, программное обеспечение, поиск уязвимостей, базовый принцип, генетическая дезволюция

**Благодарности.** Работа выполнена при частичной финансовой поддержке бюджетной темы FFZF-2025-0016

**Введение.** Небезопасность ПО является существенной угрозой для сферы информационных технологий. Для ее минимизации (а в идеале ликвидации) требуется решение большого количества частных, но важных научно-практических задач. Одной из таких задач является необходимость сравнения двух программ по некоторой шкале – нахождение их близости (или степени подобия). Для этого требуется создание соответствующей метрики близости (далее – метрика) как некоторого расстояния в условном пространстве программного кода [1]. Актуальность задачи определяется способами применения и областями приложения ее решений.

Во-первых, близость машинного кода используемой программы и содержащегося в базе вредоносного ПО позволяет оценивать небезопасность первой (например, с применением частотного распределения байтов или нечетких хэшей [2]). Однако у данного способа обнару-

жения вредоносного ПО имеется ряд проблем, связанных с существенной зависимостью точной последовательности байтов инструкций от множества факторов, включая даже ключи компиляции исходного кода программ. Применение же злоумышленником техник обфускации [3], предназначенных для существенного изменения кода программы без потери ее функциональности, вообще сделает бесполезной попытку поиска машинного кода исследуемой программы по базе вредоносного ПО.

Во-вторых, применение степени близости программ заключается в поиске дубликатов кода, содержащих и уязвимости [4]. Для этого, в частности, может применяться анализ последовательностей инструкций машинного кода с построением графа потока управления [5].

В-третьих, расширение способа поиска дубликатов путем учета более сложных синтаксических и семантических подобий даст возможность определения авторства программы

как для всего исходного кода, так и для его частей [6].

Предполагаемым применением решения задачи является ее использование в исследованиях [7, 8], направленных на получение более ранних и высокоуровневых представлений программы (например, исходного кода, алгоритмов и архитектуры) по ее низкоуровневым представлениям (например, по машинному коду). Для этого предлагается применять так называемый генетический реверс-инжиниринг, состоящий из типовых генетических деэволюций ближайших представлений программы [7, 8]. Данный подход является развитием генетических алгоритмов, представляющих собой один из способов решения оптимизационной задачи в виде максимизации (или минимизации) некоторой функции путем итеративного подбора ее аргументов, которыми в данном случае определяют некоторый экземпляр исходного кода. Одно из центральных вычислений в генетическом алгоритме – функция приспособленности, которая заключается во взаимной близости двух программ в одном представлении (например, двух экземпляров машинного кода) – то есть метрику. Она и является искомым решением задачи и предметом настоящего научного исследования.

### Обзор исследований

Приведем обзор научных публикаций, в которых предлагаются различные способы сравнения и оценки близости массивных символьных сущностей (данных), не ограничиваясь только выполняемыми программами, их исходным и машинным кодами.

Классической мерой сходства двух множеств считается индекс Жаккара, вычисляемый как отношение мощностей (то есть размеров) пересечения множеств к их объединению [9]; впрочем, для формулы индекса есть и другие вариации. Так, для идентичных множеств индекс равен 1, а для содержащих абсолютно разные элементы – 0.

Другим подходом к оценке близости элементов множеств является их кластеризация различными алгоритмами (например,  $k$ -средних, семейство FOREL и др.). Соответственно, элементы одного кластера считаются близкими по некоторым признакам [10].

Оценивание близости программ к случайным последовательностям байт позволяет идентифицировать данные как зашифрованные. Для этого, помимо вычисления энтропии дан-

ных, предлагается применять вейвлет-преобразования.

Близость машинного кода программ для процессорных платформ x86, ARM и MIPS в исследовании [11] оценивается через семантику машинных инструкций, обрабатываемых в процессе выполнения или эмуляции.

В работе [12] оценивается близость текстов, представленных в виде структурированных документов, с применением смыслового анализатора. Для этого, в частности, выделяются основные понятия в тексте и их онтологические связи, задаются параметры анализа (сила и вес связи), проверяется совпадение терминологий сравниваемых документов, строится семантическая сеть документов и оценивается их соответствующая (семантическая) близость.

Уже классикой считается оценка близости документов на основании их тематической сепарации с применением кластеризации.

Описание различных представлений матриц близости на основе пространственной, теоретико-множественной и графовой моделей, а также их объединения дано в работе [13]. В качестве основного применения моделей указывается анализ социологических данных.

Исследование [14] посвящено поиску дубликатов графов. Для этого предлагается оригинальный алгоритм, основанный на индексации графов, а также на различных мерах их схожести.

Как показал краткий обзор опубликованных результатов научных исследований, все они или являются слишком общими и прогнозируемо имеющими низкую чувствительность для сравнения программ (например, применение индекса Жаккара или тематическая сепарация), или же, наоборот, учитывают частные аспекты данных и обладают слабой инвариантностью ко всему множеству процессоров выполнения машинного кода, языков программирования исходного кода и т.п. (например, использование семантики машинных инструкций). Предлагается авторская метрика, гипотетически применимая практически для любых текстовых и бинарных представлений двух программ и при этом обладающая высокой чувствительностью к их различиям.

### Описание метрики

*Требования.* Основное предназначение метрики, помимо применения для решения различных частных задач информационной безопасности, обусловлено необходимостью создания

авторского способа генетической дэволюции представлений программ. При этом наиболее классической дэволюцией считается преобразование программы из машинного кода (из бинарного представления) в исходный (в текстовое представление). Последнее может быть проанализировано экспертом по информационной безопасности программы на предмет наличия уязвимостей. Процесс получения текстового представления инструкций машинного кода, то есть ассемблерного кода программы, является хорошо отработанным и реализуемым с помощью специальных утилит – дизассемблеров [15]. Таким образом, метрика может оперировать именно текстами ассемблерного кода.

*Принцип вычисления* метрики для текстов (то есть списков из последовательностей символов)  $X$  и  $Y$  заключается в оценке близости между всеми строками из  $X$  и  $Y$ . Затем выбирается максимальное значение близости между каждой строкой из  $X$  и всеми строками из  $Y$ , которое корректируется с учетом расстояния между позициями этих строк в каждом тексте. Тем самым учитывается не только близость строк, но и их отдаление от одинаковых позиций. После этого производится средняя оценка близости всех строк из текстов. Аналогичным образом принцип вычисления метрики применяется и для двух строк только на основании совпадения и нахождения на одинаковых позициях пар символов. При этом близость дополнительно корректируется с учетом дальности расположения символов и строк от начала их последовательностей и текстов. Влияние дальности определяется соответствующими коэффициентами, учитываемыми метрикой. Отсутствие сравниваемой строки в одном из списков, как и отсутствие сравниваемого символа в одной из последовательностей, может трактоваться как их нахождение на большом расстоянии, равном максимальному размеру одного из списков и максимальной длине одной из строк соответственно. Для удобства интерпретации значения метрики целесообразно отнормировать его к диапазону  $[0,1]$ , например, путем суммирования всех близостей пар строк из  $X$  и  $Y$  и их деления на количество строк в списке. Естественно, близость строк должна нормироваться к такому же диапазону. В этом случае при полном несовпадении строк из  $X$  и  $Y$  вплоть до отсутствия общих символов их попарные оценки близостей будут равны 0, что при суммировании даст значение метрики, также равное 0. Для обратной ситуации, когда все строки в  $X$

и  $Y$  полностью совпадают и находятся на тождественных позициях, попарные близости дадут значение 1, а их сумма будет равна размеру списков, на который и необходимо будет поделить итоговый результат для его нормировки. Приведенный принцип может считаться базовым, поскольку он отражает суть вычисления метрики и может использоваться для ее качественного сравнения с аналогами.

*Аналитическая модель.* Для формализации метрики предлагается аналитическая модель ее вычисления, соответствующая описанному принципу.

Основная формула близости двух строк:

$$\left\{ \begin{aligned} &Metric_{str1, str2}^{Strings} = \frac{1}{length^{Strings}} \times \\ &\times \sum_{i1 \in [1..length(str1)]} \max \left( \bigcup_{i2 \in [1..length(str2)]} metric_{i1, i2}^{Symbols} \right), \quad (1) \\ &length(str1) \geq length(str2) \end{aligned} \right.$$

где  $Metric_{str1, str2}^{Strings}$  – близость двух строк  $str1$  и  $str2$ , притом что 1-я строка выбирается более длинной или такой же, как 2-я строка;  $length^{Strings}$  – максимальная длина (в символах) из двух строк;  $i1$  и  $i2$  – индекс символа каждой из строк в диапазоне от 1 до последнего, равного длине строки;  $length(\dots)$  – операция получения длины (в данном случае – строки из символов) и таким образом  $length^{Strings} = length(str1)$ ;  $\max(\dots)$  – операция нахождения максимального элемента в множестве;  $metric_{i1, i2}^{Symbols}$  – частная метрика близости двух символов с индексами  $i1$  и  $i2$ .

Таким образом, метрика близости двух строк вычисляется как сумма частных метрик максимальных близостей символа 1-й строки и всех символов 2-й строки (такое усложнение расчета обусловлено тем, что некоторый символ в 1-й строке может присутствовать в нескольких позициях 2-й строки и необходимо выбрать наиболее близкий). Указанная частная метрика близости двух символов определяется следующим образом:

$$metric_{i1, i2}^{Symbols} = \begin{cases} 1 - \frac{diff_{i1, i2}^{SymbolCorr}}{length}, \\ eclu\ str1[i1] = str2[i2], \\ 0, eclu\ str1[i1] \neq str2[i2], \end{cases} \quad (2)$$

где  $diff_{i1, i2}^{SymbolCorr}$  – скорректированное (с учетом позиции или индекса относительно начала строк) расстояние между позициями некоторого сим-

вола в разных строках;  $str[i]$  – символ в строке  $str$  на  $i$ -й позиции.

Таким образом, если символы в  $i1$ -й позиции 1-й строки и  $i2$ -й позиции 2-й строки различны, то метрика будет минимальной – равной 0; а, например, если в одинаковых позициях (то есть  $i1 = i2$ ) находится один и тот же символ, то расстояние между ними равно 0 ( $diff_{i1,i2}^{SymbolCorr} = 0$ ) и, следовательно, метрика будет максимальной – равной 1. В других случаях указанное скорректированное расстояние определяется по следующей формуле:

$$diff_{i1,i2}^{SymbolCorr} = diff_{i1,i2}^{Symbol} \times \left( 1 + p^{SymbolDist} \left( 1 - \frac{diff_{i1,i2}^{Symbol}}{length} \right) \right), \quad (3)$$

где  $diff_{i1,i2}^{Symbol} = |i1 - i2|$  – расстояние (нескорректированное по позиции) между индексами символа в разных строках;  $p^{SymbolDist}$  – вычисляемый параметр корректировки расстояния между позициями символом, определяемым в диапазоне  $[0 \dots 1]$ ; больший параметр соответствует большему влиянию расстояния, а 0 – отсутствию такового. Значит, при наименьшей корректировке ( $p^{SymbolDist} = 0$ ) эти расстояния будут идентичны –  $diff_{i1,i2}^{SymbolCorr} = diff_{i1,i2}^{Symbol}$ . Параметр корректировки определяется следующим образом:

$$p^{SymbolDist} = \begin{cases} \frac{(avg_{i1,i2} - 1)k^{String}}{length - 1}, & \text{если } length \geq 2, \\ 0, & \text{если } length = 1, \end{cases} \quad (4)$$

где  $avg_{i1,i2} = \frac{i1 + i2}{2}$  – средняя позиция символа, определяемая по позициям в двух строках ( $i1$  и  $i2$ );  $k^{String}$  – задаваемый коэффициент дополнительной корректировки близости согласно расстоянию символов от начала строк, определяемый в диапазоне  $[0 \dots 1]$ ; больший параметр соответствует большему влиянию расстояния, а 0 – отсутствию такового.

В случае строк единичной длины ( $length = 1$ ) согласно формуле (4) было бы деление на 0, и во избежание такой ситуации добавлено дополнительное условие, определяющее для таких строк отсутствие влияния расстояния на метрику ( $p^{SymbolDist} = 0$ ). Если дополнительная корректировка близости отсутствует ( $k^{String} = 0$ ), корректировка расстояния между позициями символов также будет отсутствовать ( $p^{SymbolDist} = 0$ ).

Аналогична формула близости двух текстов (вычисляемая через близость их символов):

$$\left\{ \begin{aligned} &Metric_{txt1,txt2}^{Texts} = \frac{1}{length^{Texts}} \times \\ &\times \sum_{i1 \in [1..length(txt1)]} \max \left( \bigcup_{i2 \in [1..length(txt2)]} metric_{i1,i2}^{StringsCorr} \right), \quad (5) \\ &length(txt1) \geq length(txt2) \end{aligned} \right.$$

где  $Metric_{txt1,txt2}^{Texts}$  – близость двух текстов  $txt1$  и  $txt2$  (как и для  $Metric_{str1, str2}^{Strings}$ , размер первого объекта сравнения выбирается больше второго, но уже в виде количества строк);  $length = (\dots)$  – указанная ранее операция получения длины (в данном случае – текста из строк), таким образом,  $length^{Texts} = length(txt1)$ ;  $metric_{i1,i2}^{StringsCorr}$  – частная метрика близости  $i1$ -й строки из 1-го текста и  $i2$ -й строки из 2-го текста с учетом корректировок, которая определяется следующим образом:

$$metric_{i1,i2}^{StringsCorr} = \begin{cases} Metric_{str1, str2}^{Strings} \times \left( 1 - \frac{diff_{i1,i2}^{StringCorr}}{length^{Texts}} \right), \\ \text{если } Metric_{str1, str2}^{Strings} = \max (metrics_{i1}^{String}) \\ 0, \text{ если } Metric_{str1, str2}^{Strings} \neq \max (metrics_{i1}^{String}) \end{cases}, \quad (6)$$

где  $Metric_{str1, str2}^{Strings}$  – метрика близости двух строк, описанная ранее;  $diff_{i1,i2}^{StringCorr}$  – скорректированное расстояние между позициями некоторой строки в разных текстах, вычисляемое полностью аналогично  $diff_{i1,i2}^{SymbolCorr}$ , но с использованием вводимого коэффициента дополнительной корректировки близости согласно расстоянию строк от начала текста  $k^{Text}$ , подставляемого в формулу (6) вместо  $k^{String}$ ;  $metrics_{i1}^{String}$  – множество метрик близости  $i1$ -й строки из 1-го текста и всех строк из 2-го текста:

$$\left\{ \begin{aligned} &metrics_{i1}^{String} = \bigcup_{i2 \in [length(txt1)]} Metric_{str1, str2}^{Strings} \\ &str1 = txt[i1] \\ &str2 = txt[i2] \end{aligned} \right., \quad (7)$$

где  $str1$  и  $str2$  – строки в текстах  $txt1$  и  $txt2$ , расположенные на позициях  $i1$  и  $i2$ .

Согласно базовому принципу, формулы для вычисления метрики близости текстов практически полностью идентичны формулам метрики близости строк (1)–(7) с тем дополнением, что первые в своем расчете используют вторые через формулу (1) как дополнительный параметр, способный ослаблять основную метрику.

Алгоритм вычисления, реализован на языке Python 3.11. Наиболее сложной для понимания является формула (3) для скорректированного расстояния символов (и аналогичная ей для строк) в зависимости от их начала, что можно пояснить на следующем примере. Предположим, что необходимо оценить близость символа «x», присутствующего в единственном экземпляре в двух строках длиной в 20 символов; при этом символ «x» в первой строке расположен на 10-й позиции. Тогда зависимость значения  $diff_{i1,i2}^{SymbolCorr}$  при различных корректировочных коэффициентах ( $k^{String}$ ) и позициях второго символа «x» будет такой, как представлено в таблице 1. В нее также добавлена тепловая маркировка ячеек таблицы – зеленый цвет соответствует меньшим значениям расстояния, а красный – большим.

Согласно тепловой карте и значениям  $diff_{i1,i2}^{SymbolCorr}$  в ячейках таблицы 1, при нахождении 2-го символа «x» в той же позиции, что и первый (то есть равной 10), расстояние при любых корректировочных коэффициентах равно 0. По мере увеличения расстояния между этими символами в двух строках  $diff_{i1,i2}^{SymbolCorr}$  постепенно увеличивается (максимум достигается в позициях 0 и 20), при этом при более высоких  $k^{String}$  (максимум достигается при значении 1.0) данное увеличение будет более существенным. Выявленная закономерность доказывает хорошую чувствительность метрики как к расстоянию между одинаковыми символами/

строками, так и к их отступу от начала последовательности/списка.

### Эксперимент

Для демонстрации работоспособности и применимости предлагаемой метрики был проведен соответствующий эксперимент. Ее первая версия (далее – метрика\_1), не учитывающая удаленность символов и строк от начала последовательностей и списков, уже была базово протестирована. В данном исследовании основные тестируемые аспекты были расширены для оценивания именно ее новой версии (далее – метрика\_2). Ранее значение метрики\_1 сопоставлялось с индексом Жаккарда и показало ее преимущество над классическим инструментом сравнения множеств. Сравним метрику\_2 с метрикой\_1, что позволит увидеть ее новизну, особенности и преимущества.

Все тесты были поделены на четыре группы. Поскольку алгоритмы вычисления близости строк алгоритмически подобны таким же алгоритмам для близости текстов, тесты будут приведены только для первых – они полностью соответствуют результатам тестирования вторых. Вводимый коэффициент дополнительной корректировки близости строк ( $k^{String}$ ) соответствовал незначительному влиянию позиции символов относительно начала строки (равнялся 0.1), а значение коэффициента дополнительной корректировки текстов ( $k^{Text}$ ) не учитывалось (равнялось 0.0).

Таблица 1

Зависимость  $diff_{i1,i2}^{SymbolCorr}$  от  $k^{String}$  и позиции символа «x» во второй строке

Table 1

Dependence of  $diff_{i1,i2}^{SymbolCorr}$  on  $k^{String}$  and the character “x” position in the 2nd line

Коэффициент корректировки ( $k^{String}$ )	Позиция символа «x» во второй строке ( $diff_{i1,i2}^{SymbolCorr}$ )																				
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
0	10.0	9.0	8.0	7.0	6.0	5.0	4.0	3.0	2.0	1.0	0.0	1.0	2.0	3.0	4.0	5.0	6.0	7.0	8.0	9.0	10.0
0.1	10.1	9.1	8.2	7.2	6.2	5.2	4.1	3.1	2.1	1.1	0.0	1.1	2.1	3.2	4.2	5.3	6.3	7.3	8.4	9.4	10.4
0.2	10.3	9.3	8.3	7.3	6.3	5.3	4.3	3.2	2.2	1.1	0.0	1.1	2.2	3.3	4.4	5.5	6.6	7.7	8.7	9.8	10.8
0.3	10.4	9.4	8.5	7.5	6.5	5.4	4.4	3.3	2.3	1.1	0.0	1.2	2.3	3.5	4.6	5.7	6.9	8.0	9.1	10.1	11.2
0.4	10.5	9.6	8.6	7.6	6.6	5.6	4.5	3.5	2.3	1.2	0.0	1.2	2.4	3.6	4.8	6.0	7.2	8.3	9.4	10.5	11.6
0.5	10.7	9.7	8.8	7.8	6.8	5.7	4.7	3.6	2.4	1.2	0.0	1.3	2.5	3.8	5.0	6.2	7.4	8.6	9.8	10.9	12.0
0.6	10.8	9.9	8.9	7.9	6.9	5.9	4.8	3.7	2.5	1.3	0.0	1.3	2.6	3.9	5.2	6.5	7.7	8.9	10.1	11.3	12.4
0.7	10.9	10.0	9.1	8.1	7.1	6.0	4.9	3.8	2.6	1.3	0.0	1.4	2.7	4.1	5.4	6.7	8.0	9.3	10.5	11.6	12.8
0.8	11.1	10.2	9.2	8.3	7.2	6.2	5.1	3.9	2.7	1.4	0.0	1.4	2.8	4.2	5.6	7.0	8.3	9.6	10.8	12.0	13.2
0.9	11.2	10.3	9.4	8.4	7.4	6.3	5.2	4.0	2.8	1.4	0.0	1.5	2.9	4.4	5.8	7.2	8.6	9.9	11.2	12.4	13.6
1.0	11.3	10.4	9.5	8.6	7.6	6.5	5.4	4.1	2.9	1.5	0.0	1.5	3.0	4.5	6.0	7.5	8.9	10.2	11.5	12.8	14.0

Рассмотрим результаты, полученные при прохождении метриками всех четырех групп тестов.

*Группа 1. Отсутствие одного символа в строке (или строки в тексте).*

В данной группе тестов производится вычисление двух метрик – ранней базовой и текущей расширенной при сравнении строки «abcd» с набором строк, составленных из 1-й строки путем замены ее символов на «\_». Количество всех таких комбинаций равно 16.

Результаты тестирования представлены в таблице 2.

Таблица 2

**Результаты вычисления метрики\_1 и метрики\_2 для группы тестов 1 (при сравнении с «abcd»)**

Table 2

**Results of metrica\_1 and metrica\_2 calculation for test group 1 (when compared with “abcd”)**

№ теста	2-я строка	Метрика_1	Метрика_2
1	abcd	1.0000	1.0000
2	abc	0.7500	0.7500
3	ab_d	0.7500	0.7500
4	a_cd	0.7500	0.7500
5	_bcd	0.7500	0.7500
6	ab__	0.5000	0.5000
7	a_c_	0.5000	0.5000
8	a_d_	0.5000	0.5000
9	_bc_	0.5000	0.5000
10	_b_d	0.5000	0.5000
11	_cd	0.5000	0.5000
12	a__	0.2500	0.2500
13	_b__	0.2500	0.2500
14	_c__	0.2500	0.2500
15	_d__	0.2500	0.2500
16	____	0.0000	0.0000

Результаты вычисления метрик согласно таблице 2 в виде гистограмм представлены на рисунке 1.

Анализ полученных результатов позволяет сделать следующие выводы.

Во-первых, значения метрик достаточно корректно отображают близость строк, поскольку тесты с большим номером (соответствующим, в том числе и степени изменения первоначальной строки) имеют существенные отличия от строки «abcd».

Во-вторых, значения метрик идентичны, что вполне закономерно, поскольку в строках все символы или совпадают и находятся в одинаковых позициях, или отсутствуют (то есть заменены на «\_»), что не требует учета коэффициента коррективы  $k^{String}$ .

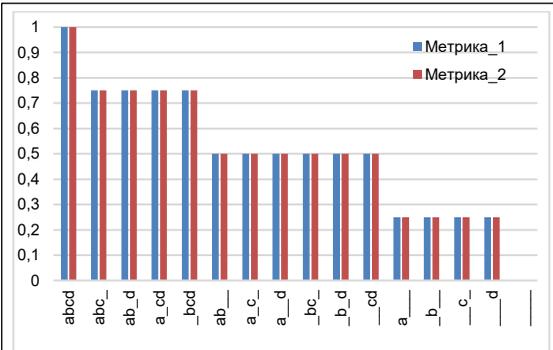


Рис. 1. Гистограмма результатов вычисления метрики\_1 и метрики\_2 для группы тестов 1 (при сравнении с «abcd»)

Fig. 1. Histogram of the calculation results of metrica\_1 and metrica\_2 for test group 1 (when compared with “abcd”)

*Группа 2. Присутствие одного символа в строке (или строки в тексте) в разных позициях.*

В данной группе тестов производится вычисление метрик при сравнении 1-й строки «abcd» с набором строк, составленных из одного символа 1-й строки на разных позициях при заполнении остальных позиций символом «\_». Количество таких комбинаций равно 16.

Результаты тестирования представлены в таблице 3.

Таблица 3

**Результаты вычисления метрики\_1 и метрики\_2 для группы тестов 2 (при сравнении с «abcd»)**

Table 3

**Results of metrica\_1 and metrica\_2 calculation for test group 2 (when compared with “abcd”)**

№ теста	2-я строка	Метрика_1	Метрика_2
1	a	0.2500	0.2500
2	a	0.1875	0.1867
3	a	0.1250	0.1229
4	a	0.0625	0.0602
5	b	0.1875	0.1867
6	b	0.2500	0.2500
7	b	0.1875	0.1852
8	b	0.1250	0.1208
9	c	0.1250	0.1229
10	c	0.1875	0.1852
11	c	0.2500	0.2500
12	c	0.1875	0.1836
13	d	0.0625	0.0602
14	d	0.1250	0.1208
15	d	0.1875	0.1836
16	d	0.2500	0.2500

Результаты вычисления метрик согласно таблице 3 представлены в виде гистограмм на рисунке 2.

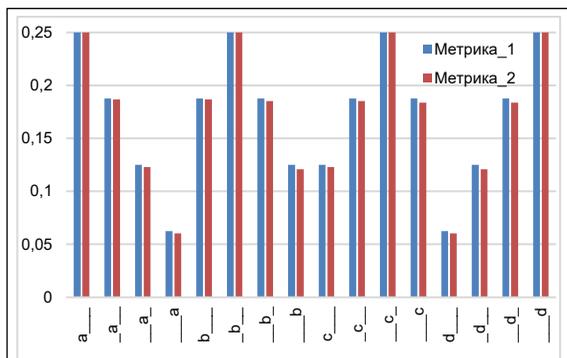


Рис. 2. Гистограмма результатов вычисления метрики\_1 и метрики\_2 для группы тестов 2 (для сравнения строк с «abcd»)

Fig. 2. Histogram of the calculation results of metrica\_1 and metrica\_2 for test group 2 (for comparing lines with "abcd")

Анализ полученных результатов позволяет сделать следующие выводы.

Во-первых, значения метрик достаточно корректно отображают близость строк, поскольку перемещение символа из 1-й строки (то есть кроме заполнителя « ») от корректной позиции к удаленным приводит к существенному уменьшению близости строк, что для метрики\_1 соответствует значениям 0.25, 0.1875, 0.1250 и 0.0625.

Во-вторых, значения метрики\_2 незначительно отличаются от метрики\_1 именно из-за учета позиции символа 2-й строки от ее начала. Так, смещение символа «а» вправо на 1 позицию дает значение 0.1867 вместо 0.1875, аналогичное смещение «b» – 0.1852, а «с» – 0.1836.

Группа 3. Комбинации символов строк (или строк текста).

В данной группе тестов производится вычисление метрик при сравнении 1-й строки «abcd» с набором строк, составленных из всех комбинаций ее символов (то есть без их дублирования и использования заполнителя « »). Количество таких комбинаций равно 24, а их порядок такой, что вначале перебирается 4-й символ, а затем 3-й, 2-й и 1-й.

Результаты тестирования представлены в таблице 4.

Результаты вычисления метрик согласно таблице 4 представлены в виде гистограмм на рисунке 3.

Анализ полученных результатов позволяет сделать следующие выводы.

Таблица 4

Результаты вычисления метрики\_1 и метрики\_2 для группы тестов 3 (при сравнении с «abcd»)

Table 4

Results of metrica\_1 and metrica\_2 calculation for test group 3 (when compared with "abcd")

№ теста	2-я строка	Метрика_1	Метрика_2
1	abcd	1.0000	1.0000
2	abdc	0.8750	0.8672
3	acbd	0.8750	0.8703
4	acdb	0.7500	0.7396
5	adbc	0.7500	0.7396
6	adcb	0.7500	0.7417
7	bacd	0.8750	0.8734
8	badc	0.7500	0.7406
9	bcad	0.7500	0.7448
10	bcda	0.6250	0.6156
11	bdac	0.6250	0.6141
12	bdca	0.6250	0.6177
13	cabd	0.7500	0.7448
14	cadb	0.6250	0.6141
15	cbad	0.7500	0.7458
16	cbda	0.6250	0.6167
17	cdab	0.5000	0.4875
18	cdba	0.5000	0.4891
19	dabc	0.6250	0.6156
20	dacb	0.6250	0.6177
21	dbac	0.6250	0.6167
22	dbca	0.6250	0.6203
23	dcab	0.5000	0.4891
24	dcba	0.5000	0.4906

Во-первых, метрики достаточно адекватно учитывают близость строк, что индицируется постепенным уменьшением ее значения для

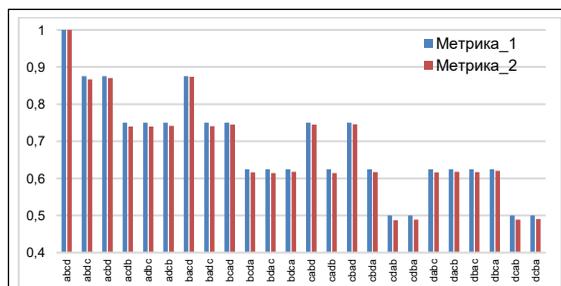


Рис. 3. Гистограмма результатов вычисления метрики\_1 и метрики\_2 для группы тестов 3 (для сравнения строк с «abcd»)

Fig. 3. Histogram of calculation results of metrica\_1 and metrica\_2 for test group 3 (for comparing lines with "abcd")

последующих комбинаций символов во 2-й строке.

Во-вторых, зависимость между значением метрик и комбинациями символов 2-й строки существенно сложнее, чем простой снижающийся тренд, что также закономерно. Так, 7-й тест («bacd») обладает большей близостью, чем 6-й («adcb»), имеющий совпадение 1-го символа «а» (с 1-й строкой «abcd»). Это можно обосновать интегральностью вычисления метрик, складывающейся из близости всех четырех символов строк.

В-третьих, незначительное отличие значений метрики\_2 от метрики\_1, как и ранее, обосновано учетом позиции символов (точнее их середины). Так, например, перестановка двух ближайших символов во 2-й строке дает одинаковое значение метрики\_1 – 0.8750, хотя метрика\_2 учитывает их позицию, уменьшая свое значение для более отдаленных: «bacd» – 0.8734 (тест 7), «acbd» – 0.8703 (тест 3), «abdc» – 0.8672 (тест 2).

*Группа 4. Изменение количества символов строки (или строк текста).*

В данной группе тестов производится вычисление двух метрик при сравнении 1-й строки «12345» с набором строк, составленных последовательным увеличением ее длины сначала совпадающими символами из 1-й строки, то есть до 5, а затем, добавляя новые символы, до 20. Количество таких комбинаций равно 21, поскольку 1-й тест содержит пустую строку.

Итоги тестирования представлены в таблице 5.

Результаты вычисления метрик согласно таблице 5 представлены в виде гистограмм на рисунке 4.

Анализ полученных результатов позволяет сделать следующие выводы:

- по мере дополнения 2-й строки до 1-й (тесты с 1-го по 6-й) значение метрик линейно растет от 0 до 1 с шагом 0.2, что вполне корректно и точно отражает их близость;

- по мере дополнения 2-й строки новыми символами (тесты с 7-го по 21-й) значение метрик снижается (с околологарифмической, но не являющейся таковой, закономерностью), что позволяет с достаточной точностью как отражать совпадение начальной части 2-й строки с 1-й, так и учитывать наличие в ней новых символов, отсутствующих в 1-й;

- значения метрик идентичны, что вполне закономерно, поскольку 2-я строка содержит часть символов 1-й строки на корректных позициях, тождественна ей или же дополнена новыми символами.

Таблица 5

**Результаты вычисления метрики\_1 и метрики\_2 для группы тестов 4 (при сравнении с «12345»)**

Table 5

**Results of metrica\_1 and metrica\_2 calculation for test group 4 (when compared with “12345”)**

№ теста	2-я строка	Метрика_1	Метрика_2
1		0.0000	0.0000
2	1	0.2000	0.2000
3	12	0.4000	0.4000
4	123	0.6000	0.6000
5	1234	0.8000	0.8000
6	12345	1.0000	1.0000
7	123456	0.8333	0.8333
8	1234567	0.7143	0.7143
9	12345678	0.6250	0.6250
10	123456789	0.5556	0.5556
11	1234567890	0.5000	0.5000
12	1234567890a	0.4545	0.4545
13	1234567890ab	0.4167	0.4167
14	1234567890abc	0.3846	0.3846
15	1234567890abcd	0.3571	0.3571
16	1234567890abcde	0.3333	0.3333
17	1234567890abcdef	0.3125	0.3125
18	1234567890abcdefg	0.2941	0.2941
19	1234567890abcdefgh	0.2778	0.2778
20	1234567890abcdefghi	0.2632	0.2632
21	1234567890abcdefghik	0.2500	0.2500

**Обсуждение результатов**

Укажем ряд особенностей текущей версии метрики, которые, хотя в ряде случаев и могут считаться недостатками, не являются критичными и позволяют применять ее в интересах генетической дэволюции.

Во-первых, вводимые коэффициенты дополнительной корректировки близости (то есть  $k^{Text}$  и  $k^{String}$ ) при различных позициях символов (и строк) оказывают не такое существенное влияние, как предполагалось изначально (табл. 1). Следовательно, требуется дополнительное исследование. Впрочем, для генетической дэволюции даже незначительный учет дальности символов и строк может иметь существенное значение для решения задачи подбора исходного кода, поскольку оно качественно повлияет на время и успешность селекции.

Во-вторых, несмотря на работоспособность метрики и ее применимость в алгоритмах гене-

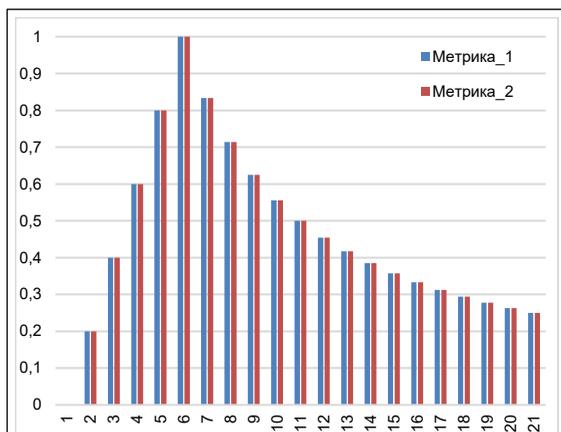


Рис. 4. Гистограмма результатов вычисления метрики\_1 и метрики\_2 для группы тестов 4 (при сравнении с «12345»)

Fig. 4. Histogram of the calculation results of metrica\_1 and metrica\_2 for test group 4 (when compared with "12345")

тической дэволюции, она не позволяет в полной мере учесть конкретный формальный синтаксис ассемблерного кода программы, описывая его как список строк.

В-третьих, были рассмотрены лишь достаточно общие случаи сравнения текстов и строк, что не позволяет оценивать в полной мере работоспособность метрики для более сложных случаев. Тем не менее такая ситуация в прин-

ципе не имеет однозначного решения, что не позволяет требовать его и от метрики.

### Заключение

Результатом исследования стала новая авторская версия метрики близости двух программ, представленных в виде текстов ассемблерного кода программы, которая отличается от предыдущей версии большей чувствительностью к различиям как в текстовых строках, так и в их символах (путем учета позиции конструкций относительно начала). Аналитическая запись метрики, а также ряд преимуществ (например, инвариантность к синтаксису текстов, оценочная шкала на отрезке  $[0 \dots 1]$  и др.) позволяют применять ее как часть комплексных математических инструментов без дополнительных доработок. Проведенные над метрикой эксперименты говорят не только о ее работоспособности в интересах генетической дэволюции, но и об общей востребованности в сфере информационных технологий и их безопасности.

Продолжением исследования должны стать устранение указанных недостатков метрики, а также синтез ее следующей версии, еще более инвариантной к сравниваемым представлениям данных за счет использования формальных синтаксисов этих представлений, передаваемых в метрику как один из параметров.

### Список литературы

1. Гусева Н.И. Пространства над алгебрами с евклидовой метрикой // Итоги науки и техники. Сер. Современная математика и ее приложения. Тематические обзоры. 2020. Т. 179. С. 10–15. doi: 10.36535/0233-6723-2020-179-10-15.
2. Израйлов К.Е., Гололобов Н.В., Краскин Г.А. Метод анализа вредоносного программного обеспечения на базе Fuzzy Hash // Информатизация и связь. 2019. № 2. С. 36–44.
3. Luo L., Ming J., Wu D., Liu P., Zhu S. Semantics-based obfuscation-resilient binary code similarity comparison with applications to software and algorithm plagiarism detection. IEEE Transactions on Software Engineering, 2017, vol. 43, no. 12, pp. 1157–1177. doi: 10.1109/TSE.2017.2655046.
4. Ашуров А.Я. Реализация переполнения стекового буфера программ и методы противодействия уязвимости // Системы синхронизации, формирования и обработки сигналов. 2023. Т. 14. № 1. С. 10–17.
5. Liu S., Yu C., Wang L., Wang C., Xu G. FF-BCSD: A Binary code similarity detection method based on feature fusion. Proc. ICCS, 2023, pp. 757–761. doi: 10.1109/ICCS59590.2023.10507703.
6. Ghosh K., Otero D. Discovering authorship of vulnerabilities in open source software. Proc. APSEC Workshops, 2021, pp. 41–46. doi: 10.1109/APSECW53869.2021.00018.
7. Израйлов К.Е. Концепция генетической дэволюции представлений программы. Ч. 1 // Вопросы кибербезопасности. 2024. № 1. С. 61–66. doi: 10.21681/2311-3456-2024-1-61-66.
8. Израйлов К.Е. Концепция генетической дэволюции представлений программы. Ч. 2 // Вопросы кибербезопасности. 2024. № 2. С. 81–86. doi: 10.21681/2311-3456-2024-2-81-86.
9. Fletcher S., Islam M.Z. Comparing sets of patterns with the Jaccard index. Australasian J. of Information Systems, 2018, vol. 22. doi: 10.3127/ajis.v22i0.1538.
10. Татарникова Т.М., Богданов П.Ю. Построение психологического портрета человека с применением технологий обработки естественного языка // Науч.-технич. вестн. информационных технологий, механики и оптики. 2021. Т. 21. № 1. С. 85–91. doi: 10.17586/2226-1494-2021-21-1-85-91.
11. Hu Y., Wang H., Zhang Y., Li B., Gu D. A semantics-based hybrid approach on binary code similarity comparison. IEEE Transactions on Software Engineering, 2021, vol. 47, no. 6, pp. 1241–1258. doi: 10.1109/TSE.2019.2918326.
12. Усачев Ю.Е. Вычисление степени семантической близости документов // XXI век: итоги прошлого и проблемы настоящего плюс. 2016. № 6. С. 96–103.

13. Ермолаев А.В. Методы анализа и визуализации структуры данных о близости // Социология: 4М. 2005. № 21. С. 150–171.

14. Naveena P., Rao P.K.S. Detection of near duplicates over graph datasets using pruning. Proc. INDISCON, 2020, pp. 309–313. doi: 10.1109/INDISCON50162.2020.00068.

15. Аникин И.В., Исяндавлетова Я.М. Реверсивный анализ вредоносного программного обеспечения Raccoon Stealer // Инженерный вестник Дона. 2023. № 4. С. 238–251.

Software &amp; Systems

doi: 10.15827/0236-235X.149.089-099

2025, 38(1), pp. 89–99

**Author's metric for assessing proximity of programs:  
Application for vulnerability search using genetic de-evolution**

Mikhail V. Buynovich <sup>1</sup>, Konstantin E. Izrailov <sup>2</sup>✉

<sup>1</sup> St. Petersburg University of State Fire Service of EMERCOM of Russia,  
St. Petersburg, 196105, Russian Federation

<sup>2</sup> St. Petersburg Federal Research Center of RAS,  
St. Petersburg, 199178, Russian Federation

**For citation**

Buynovich, M.V., Izrailov, K.E. (2025) 'Author's metric for assessing proximity of programs: Application for vulnerability search using genetic de-evolution', *Software & Systems*, 38(1), pp. 89–99 (in Russ.). doi: 10.15827/0236-235X.149.089-099

**Article info**

Received: 10.06.2024

After revision: 25.06.2024

Accepted: 28.06.2024

**Abstract.** The paper is relevant due to the tasks in the field of information security that require comparison of programs in their different representations, for example, in text assembly code (e.g., for vulnerability search or authorship verification). The paper presents a proximity metric for two texts in the form of a character string list, which is a development of its previous author's version. The main result of the current study (as a part of the main study aimed at genetic de-evolution of programs) is the metric itself, as well as its characteristics and peculiarities revealed through experiments. The paper presents the metric in analytical form implemented in Python. The metric takes at the input two lists of character lines for comparison, and the coefficients of taking into account the element position from the beginning of the list and the character sequence. The calculation result is a numeric value in the range from 0 to 1. Metric's novelty is in a sufficiently accurate and sensitive assessment of two texts' proximity regardless of data representation formats. The current metric version differs from the previous one by taking into account the mentioned coefficients. Theoretical significance lies in the development of comparing methods for arbitrary texts that are a list of character lines containing information, which appears sequentially according to a certain logic (requires position consideration). Besides the general purpose of comparative tools like this, the metric is practically relevant due to the possibility of determining the proximity of two programs. These programs have a binary representation of the machine code. It is pre-transformed into a textual representation of an assembly code.

**Keywords:** information security, proximity metric, software, vulnerability search, basic principle, genetic de-evolution

**Acknowledgements.** The work was partially financially supported from budget topic FFZF-2025-0016

**References**

1. Guseva, N.I. (2020) 'Spaces over algebras with Euclidean metric', *Progress in Sci. and Tech. Ser. Contemporary Mathematics and Its Applications. Thematic Surveys*, 179, pp. 10–15 (in Russ.). doi: 10.36535/0233-6723-2020-179-10-15.

2. Izrailov, K.E., Gololobov, N.V., Kraskin, G.A. (2019) 'Method of malware analysis based on Fuzzy Hash', *Informatization and Communication*, (2), pp. 36–44 (in Russ.).

3. Luo, L., Ming, J., Wu, D., Liu, P., Zhu, S. (2017) 'Semantics-based obfuscation-resilient binary code similarity comparison with applications to software and algorithm plagiarism detection', *IEEE Transactions on Software Engineering*, 43(12), pp. 1157–1177. doi: 10.1109/TSE.2017.2655046.

4. Ashurov, A.Ya. (2023) 'Stack buffer overflow of programs implementation and methods of counteracting the vulnerability', *Systems of Signal Synchronization, Generating and Processing*, 14(1), pp. 10–17 (in Russ.).

5. Liu, S., Yu, C., Wang, L., Wang, C., Xu, G. (2023) 'FF-BCSD: A Binary code similarity detection method based on feature fusion', *Proc. ICCS*, pp. 757–761. doi: 10.1109/ICCS59590.2023.10507703.

6. Ghosh, K., Otero, D. (2021) 'Discovering authorship of vulnerabilities in open source software', *Proc. APSEC Workshops*, pp. 41–46. doi: 10.1109/APSECW53869.2021.00018.

7. Izrailov, K.E. (2024) 'The genetic de-evolution concept of program representations. Pt 1', *Cybersecurity Issues*, (1), pp. 61–66 (in Russ.). doi: 10.21681/2311-3456-2024-1-61-66.
8. Izrailov, K.E. (2024) 'The genetic de-evolution concept of program representations. Pt 2', *Cybersecurity Issues*, (2), pp. 81–86 (in Russ.). doi: 10.21681/2311-3456-2024-2-81-86.
9. Fletcher, S., Islam, M.Z. (2018) 'Comparing sets of patterns with the Jaccard index', *Australasian J. of Information Systems*, 22. doi: 10.3127/ajis.v22i0.1538.
10. Tatarnikova, T.M., Bogdanov, P.Yu. (2021) 'Human psyche creation by application of natural language processing technologies', *Sci. Tech. J. Inf. Technol. Mech. Opt.*, 21(1), pp. 85–91 (in Russ.). doi: 10.17586/2226-1494-2021-21-1-85-91.
11. Hu, Y., Wang, H., Zhang, Y., Li, B., Gu, D. (2021) 'A semantics-based hybrid approach on binary code similarity comparison', *IEEE Transactions on Software Engineering*, 47(6), pp. 1241–1258. doi: 10.1109/TSE.2019.2918326.
12. Usachev, Yu.E. (2016) 'The calculation of the degree of semantic proximity of documents', *XXI Century: Resumes of the Past and Challenges of the Present Plus*, (6), pp. 96–103 (in Russ.).
13. Ermolaev, A.V. (2005) 'Methods for analyzing and visualizing the structure of proximity data', *Sociology: 4M*, (21), pp. 150–171 (in Russ.).
14. Naveena, P., Rao, P.K.S. (2020) 'Detection of near duplicates over graph datasets using pruning', *Proc. INDISCON*, pp. 309–313. doi: 10.1109/INDISCON50162.2020.00068.
15. Anikin, I.V., Isyandavletova, Ya.M. (2023) 'Reverse analysis of malware Raccoon Stealer', *Ing. J. of Don*, (4), pp. 238–251 (in Russ.).

**Авторы****Буйневич Михаил Викторович**<sup>1</sup>,д.т.н., профессор, профессор кафедры,  
bmv1958@yandex.ru**Израилов Константин Евгеньевич**<sup>2</sup>,к.т.н., доцент, старший научный сотрудник,  
konstantin.izrailov@mail.ru**Authors****Mikhail V. Buynevich**<sup>1</sup>, Dr.Sci. (Engineering),Professor, Professor of Department,  
bmv1958@yandex.ru**Konstantin E. Izrailov**<sup>2</sup>, Cand. of Sci. (Engineering),Associate Professor, Senior Researcher,  
konstantin.izrailov@mail.ru

<sup>1</sup> Санкт-Петербургский университет Государственной противопожарной службы МЧС России,  
г. Санкт-Петербург, 196105, Россия

<sup>2</sup> Санкт-Петербургский Федеральный исследовательский центр РАН,  
г. Санкт-Петербург, 199178, Россия

<sup>1</sup> St. Petersburg University of State Fire Service  
of EMERCOM of Russia, St. Petersburg,  
196105, Russian Federation

<sup>2</sup> St. Petersburg Federal Research Center of RAS,  
St. Petersburg, 199178,  
Russian Federation

## Фреймворк для автоматизации прогнозирования остаточного ресурса оборудования при построении проактивных систем поддержки принятия решений

К.С. Задиран<sup>1</sup>, Д.А. Волкова<sup>1</sup>✉, М.В. Щербаков<sup>1</sup>

<sup>1</sup> Волгоградский государственный технический университет, г. Волгоград, 400005, Россия

### Ссылка для цитирования

Задиран К.С., Волкова Д.А., Щербаков М.В. Фреймворк для автоматизации прогнозирования остаточного ресурса оборудования при построении проактивных систем поддержки принятия решений // Программные продукты и системы. 2025. Т. 38. № 1. С. 100–107. doi: 10.15827/0236-235X.149.100-107

### Информация о статье

Группа специальностей ВАК: 1.2.1

Поступила в редакцию: 08.05.2024

После доработки: 24.06.2024

Принята к публикации: 23.07.2024

**Аннотация.** В статье описан фреймворк, разработанный для автоматизации исследований в области проектирования проактивных систем поддержки принятия решений. В частности, исследуется задача анализа и прогнозирования временных рядов с целью создания инструментов автоматизации прогнозирования различных процессов в системах управления активами, в том числе технического обслуживания и ремонта. Определена роль процессов автоматизации в управлении активами в этих системах. В работе выделены основные факторы, влияющие на выбор программы для реализации системы предиктивной аналитики. Предложен алгоритм решения задачи прогнозирования оставшегося срока службы на основе анализа данных о производственных активах с использованием компонентов искусственного интеллекта. Предлагаемое программное решение, основанное на CRISP-DM, не является отдельным программным продуктом и может быть встроено в существующее программное обеспечение, поддерживая возможность модификации методов. Фреймворк выполняет следующие функции: загрузка и предобработка данных, построение моделей прогнозирования, прогнозирование временных рядов, оценка прогноза. Разработанный фреймворк обладает гибкой модульной архитектурой для добавления новых методов анализа и прогнозирования. Возможность на основе имеющихся базовых классов переопределить и реализовать собственные источники данных, стадии предобработки, модели прогнозирования и метрики расширяет вариативность и повышает эффективность его функционирования. В статье приведен пример использования фреймворка для решения задачи анализа временных рядов и определения оставшегося срока службы оборудования, демонстрирующий эффективность разработанного продукта в области исследования данных и искусственного интеллекта.

**Ключевые слова:** программный фреймворк, прогнозирование временных рядов, анализ временных рядов, системы поддержки принятия решений, RUL, модели прогнозирования

**Благодарности.** Исследование выполнено за счет гранта Российского научного фонда № 24-21-00483, <https://rscf.ru/project/24-21-00483/>

**Введение.** В современном мире эффективное управление *техническим обслуживанием, инженерной поддержкой и ремонтом* (ТОИР) является ключевым фактором для обеспечения бесперебойной работы промышленных и технических систем. Вместе с тем управление активами становится все более важным аспектом бизнеса, так как предприятия сталкиваются с растущей необходимостью оптимизации расходов, повышения производительности и увеличения надежности оборудования. Эффективность работы технических систем напрямую зависит от условий эксплуатации и поддержания работоспособного состояния.

Автоматизация процесса планирования технического обслуживания и ремонта оборудования на основе определения его остаточного ресурса дает возможность снизить риск аварийных ситуаций, минимизировать затраты на эксплуатационные расходы [1–3]. Современ-

ные подходы и методы анализа позволяют совершенствовать системы поддержки принятия решений путем внедрения механизмов прогнозирования развития событий и оценки риска неблагоприятного исхода, снижая количество отказов оборудования на основе оценки *остаточного ресурса оборудования (Remaining Useful Life, RUL)* [4].

Для управления активами в системах ТОИР RUL имеет решающее значение для оптимизации стратегий технического обслуживания, обеспечивая прогностическое планирование задач по ремонту [5]. Создание такого контроля способствует повышению маневренности действий предприятий ТОИР и своевременному реагированию на меняющееся состояние оборудования. Регулярное профилактическое обслуживание упрощается, так как система определяет начало процесса деградации, что приводит к оптимизации трудоза-

трат и снижению трудоемкости управления процессами.

Авторы исследовали работы, в которых описаны методики для анализа и прогнозирования временных рядов [6–8], и выделили четыре подхода к прогнозированию [9]: на основе моделей, данных, слияния и гибридных моделей. Стоит отметить, что все подходы, кроме моделирования, полагаются на собранные данные и включают методы, основанные на временных рядах, статистике и искусственном интеллекте. Наиболее общая классификация подходов [4] приведена на рисунке 1.

Модели прогнозирования, описывающие процесс деградации оборудования, построены на зависимости между показателями функционирования объекта и его состояниями. На основе анализа временных рядов строится алгоритм машинного обучения, способный для заданного набора параметров функционирования выдать ответ о состоянии устройства [10].

В последние годы все более актуальным становится направление по определению технического состояния с использованием машинного обучения [11]. Применение моделей прогнозирования временных рядов методами предиктивной аналитики [12, 13] с компонентами искусственного интеллекта лежит в основе проактивных систем принятия решений [14]. Для анализа временного ряда и построения прогноза широко применяются различные программные пакеты на основе предиктивной аналитики (<https://loginom.ru/>, <https://www.knime.com/about>, <https://in-dap.ru/>). Значимым критерием выбора, помимо наличия ограничений на бесплатное использование ПО, является тип лицензии, установленной разработчиком и отвечающей за возможность реализации дополнительных методов и интеграции предоставля-

емого пакета в собственное решение. Также стоит учитывать, что сборка, разворачивание и доработка свободного ПО требуют компетенций и значительных временных ресурсов.

Сложность методики построения моделей оценки RUL в проактивных системах принятия решений связана с ее долгосрочной реализацией. В качестве решения данной проблемы разработан фреймворк оценки и прогнозирования технического состояния оборудования (<https://github.com/zadiran/Dataworks>) для проведения исследований при анализе и прогнозировании временных рядов на основе анализа данных согласно методологии CRISP-DM.

### Описание фреймворка

Разработка фреймворка на языке Python началась с идеи создания интуитивно понятной программной платформы, взаимодействие с которой не требует больших временных затрат для аналитиков. Это достигается за счет применения единого алгоритма решения задачи прогнозирования RUL, представленного на рисунке 2, отдельных параметризуемых и перепределяемых шагов алгоритма, имеющих общий интерфейс, а также единого интерфейса хранения и передачи данных между шагами алгоритма. Структура программной системы приведена на рисунке 3. При получении на вход данных выполняется их подготовка: предобработка исходных данных, преобразование в выходной формат (в виде списка точек) и применение дополнительной обработки. Готовые данные сохраняются в кэше. Следующим шагом является построение модели: инициализация модели, оптимизация гиперпараметров и сохранение обученных моделей. Обученные модели с применением кросс-вали-

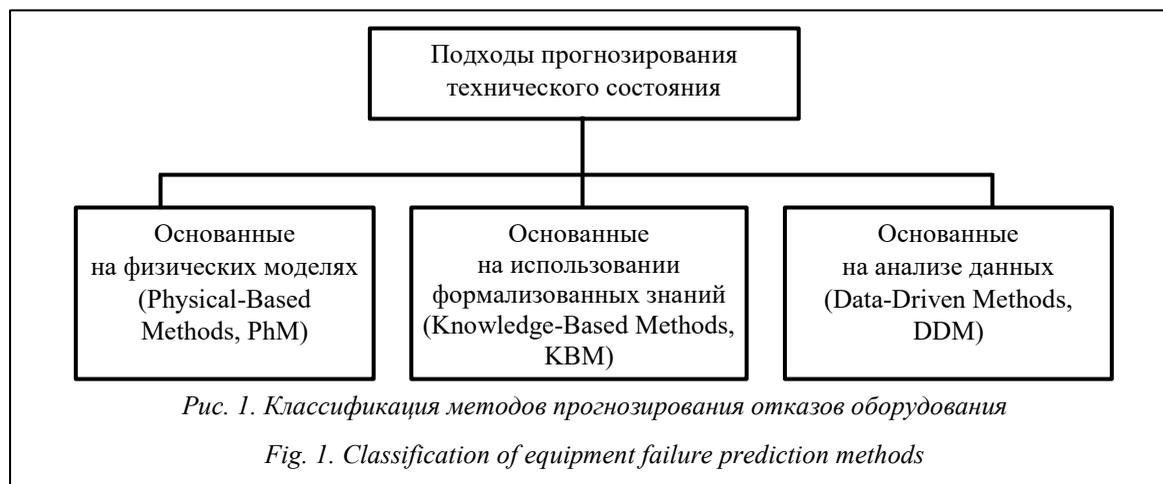


Рис. 1. Классификация методов прогнозирования отказов оборудования

Fig. 1. Classification of equipment failure prediction methods

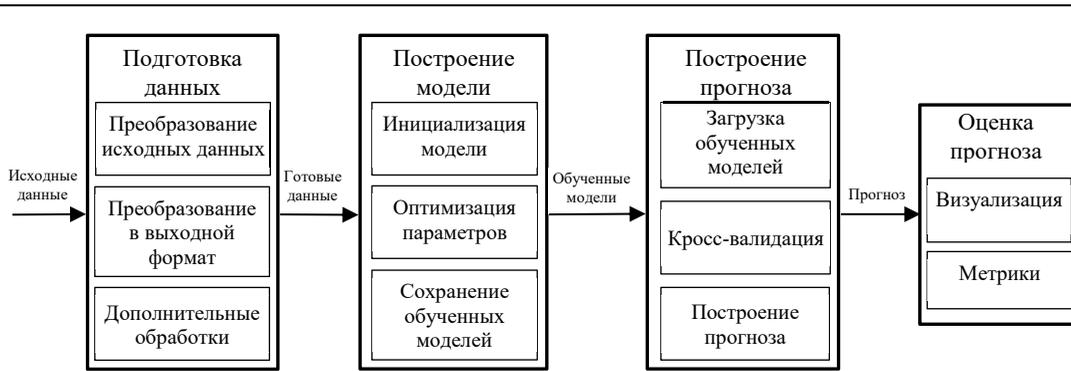


Рис. 2. Алгоритм построения прогноза

Fig. 2. Prediction algorithm

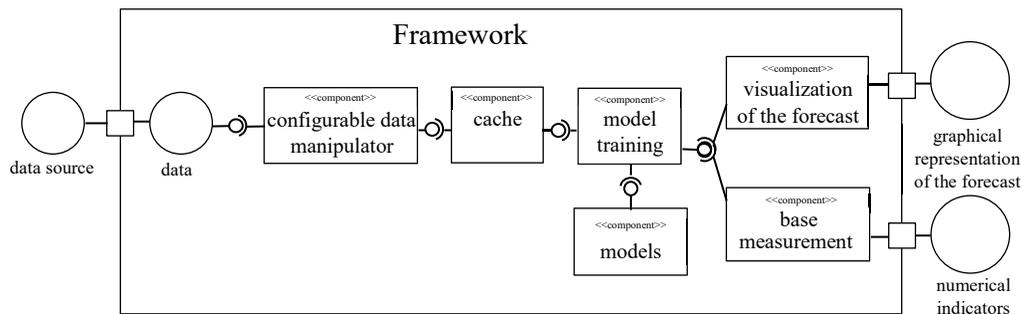


Рис. 3. Компонентная диаграмма фреймворка

Fig. 3. Component diagram of the framework

дании используются для построения прогноза и его оценки в виде расчета метрик и визуализации. Отметим, что можно переопределить каждый из базовых классов и реализовать свои собственные источники данных, стадии предобработки, модели прогнозирования и метрики.

Основная цель фреймворка – оценка технического состояния оборудования и его прогнозирование. Все функции фреймворка можно отнести к одной из следующих групп:

- загрузка и предобработка данных;
- построение моделей прогнозирования;
- прогнозирование временных рядов;
- оценка прогноза.

Рассмотрим их подробнее на примере кода по заданию параметров прогнозирования.

**Шаг 1. Загрузка данных и формирование внутреннего фрейма данных.**

Импортируем необходимые модули и функции из различных библиотек для обработки данных.

Для загрузки данных пользователь задает значения параметров *csv data source*:

‘filename’ – имя файла и путь к нему; ‘separator’ – разделитель.

Для удобства восприятия представим загружаемый исходный датасет в виде таблицы

(табл. 1), где *unit* – экземпляр оборудования, *time* – порядковый номер записи в рамках *unit*. Последняя запись в рамках *unit* – момент отказа, то есть 1 *unit* – 1 отказ. Остальные параметры – данные о работе оборудования.

**Шаг 2. Подбор параметров для методов предварительной обработки данных.**

*add\_pre\_point\_conversion\_stage()* – добавление нового этапа предобработки в список.

*set\_point\_conversion\_stage()* – этап преобразования точек.

*add\_post\_point\_conversion\_stage()* – добавление этапа постобработки в список.

Запускается метод обработки данных *get\_processed\_data* с установленными этапами.

Покажем выполнение шагов 1 и 2 на примере кода выполнения загрузки данных и настройки этапов их подготовки. Сначала устанавливаем ширину скользящего окна для последующего формирования точек типа *point* в процессе построения обучающей выборки [15].

Фрагмент кода:

```

window_size = WS
raw_data = csv_data_source().
get_data(.\data\DATA.csv', 'SEPARATOR')
cdm = configurable_data_manipulator(.\cache\
FILE_NAME.pickle')
    
```

Таблица 1

Представление исходного датасета

Table 1

Input dataset

Unit (идентификатор/ экземпляр оборудования)	Time (порядковый номер рабочего цикла)	COL 1 (номер параметра, снятого с оборудования)	...	COL N (номер параметра, снятого с оборудования)
1	1	value	...	value
1	...	value	...	value
1	192	value	...	value
2	1	value	...	value
2	...	...	...	...
2	287	value	...	value
3	1	value	...	value
...	...	...	...	...
100	200	value	...	value

```

cdm.add_pre_point_conversion_
stage(drop_columns(['COL_1', 'COL_2',]))
cdm.set_point_conversion_stage(CONVER-
TING_FORMAT(window_size))
data = cdm.get_processed_
data(raw_data)
    
```

Рассмотрим классы, используемые в приведенном фрагменте кода.

**data\_source** – абстрактный базовый класс, предоставляющий доступ к источнику данных: *get\_data(self,\*\*params) -> List[point]*: – метод принимает произвольное количество именованных параметров (**\*\*params**) и возвращает список объектов типа **point**.

**csv\_data\_source** – реализация класса **data source** для CSV-файлов. В качестве параметров принимает имя файла и разделитель данных в файле CSV. Если в данных не указаны столбцы **unit\_col** и **time\_col**, генерируется исключение, иначе столбцы переименовываются для дальнейшей обработки в датафрейме.

**configurable\_data\_manipulator** – базовый класс, инициализирующий имя файла кэша и списки этапов обработки данных:

*get\_processed\_data()->List[point]*: – основной метод, принимающий на вход датафрейм и выполняющий преобразование исходных данных в список объектов **point**.

**point** – базовый класс для хранения данных точки временного ряда, содержащей информацию о входных данных, спрогнозированных результатах и дополнительных параметрах.

**Шаг 3. Моделирование и прогнозирование.**

Фреймворк включает в себя единый базовый интерфейс для методов построения моделей прогнозирования временных рядов. На

данный момент реализованы три метода: линейная регрессия (*linear\_regression\_point\_forecast\_model*), градиентный бустинг XGBoost (*xgboost\_point\_forecast\_model*), сверточные нейронные сети (*cnn\_point\_forecast\_model*).

Пример выполнения шага 3 – код обучения модели и прогнозирования с использованием кросс-валидации:

```

def split_func(training_set, verification_set):
    bbpfm = LEARNING_MODEL()
    bbpfm.fit(training_set)
    bbpfm.predict_points(verification_set)
    splitter().run(split_func,
LIST[POINT], NUM_OF_PARTITIONS,
EXCLUSIONS)
    
```

**splitter** – базовый класс для применения кросс-валидации.

*run(func, data, num\_of\_partitions, exclusions)* – метод, принимающий на вход функцию (*func*), которую нужно выполнить на наборе данных (*data*), количество итераций и количество частей разбиения (*num\_of\_partitions*) и список исключений (*exclusions*), которые не должны участвовать в процессе кросс-валидации, возвращает список результатов после применения функции *func* на каждом разбиении данных.

*remove\_exclusions()* – метод для удаления объектов (**unit**) из списка исключений.

*split()* – метод, используемый для разделения данных на равные части, который возвращает список частей. Если разделение невозможно, возникает исключение.

*join\_df()* – метод для каждой части данных, объединяющий остальные части в список после разделения.

*hm hm()*: – метод, представляющий стратегию анализа временных рядов для выявления процесса деградации, используя критерий Краскела–Уоллиса.

**Шаг 4. Задание мер точности.**

Меры точности задаются списком измерений, фактическими и спрогнозированными значениями, которые передаются в *calculate\_measurements\_for\_points()*. На данный момент реализованы следующие меры точности: MAE, RMSE, MSE, MdAE, MAPE, MdAPE, RMSPE, RMdSPE, MASE, MdASE, RMSSE, SMAPE, SMdAPE. Пример:

```
measurements = calculate_measurements_for_points(LIST[BASE_MEASUREMENT], LIST[POINT])
for m in measurements:
    print(f '{m.name}: {m.value}')
```

**base\_measurement ()** – базовый абстрактный класс, используемый для вычисления метрик оценки качества прогнозирования между фактическими и прогнозными значениями временных рядов.

*calculate (self, actual\_set, forecasted\_set)*: – метод этого класса принимает два набора данных: *actual\_set* (фактические значения) и *forecasted\_set* (прогнозные значения).

За визуализацию данных во фреймворке отвечает метод *plot* из библиотеки *matplotlib.pyplot* – стандартная функция построения графика. Пример визуализации:

```
import matplotlib.pyplot as plt
plt.plot(list(map(lambda x: x.training_output, List[point])))
plt.plot(list(map(lambda x: x.forecasted_output, List[point])))
plt.show()
```

**Пример использования**

Настоящий фреймворк применялся для анализа временных рядов на примере задач NASA (<https://www.kaggle.com/datasets/behrad3d/nasa-staps>) для определения оставшегося срока службы (RUL) в наборе тестовых данных. В частности, для решения задачи на данных NASA были установлены значения, представленные в таблице 2, в соответствии с функциями фреймворка и выделенными ранее шагами. Сначала производятся импорт необходимых модулей и функций из различных библиотек и загрузка данных из файла *train\_FD001.csv* с агрегированными данными рабочих циклов [19]. Затем в качестве этапа предобработки данных удаляются малоинформативные для анализа столбцы и исходный датафрейм *raw\_data* преобразуется в список точек *data* с использованием метода скользящего окна, при этом размер окна задан значением параметра, равным 10. Обучается модель линейной регрессии с применением кросс-валидации (разделение имеющегося набора данных на пять частей). Для оценки качества прогноза использованы метрики точности и средняя величина ошибки – RMSE и MAE соответственно. Результаты оценки и график с прогнозными значениями данных NASA приведены на рисунке 4 (синим цветом отмечены реальные значения, оранжевым – полученные в результате прогнозирования).

Проверка работоспособности фреймворка для анализа временных рядов показала его высокую эффективность и функциональность.

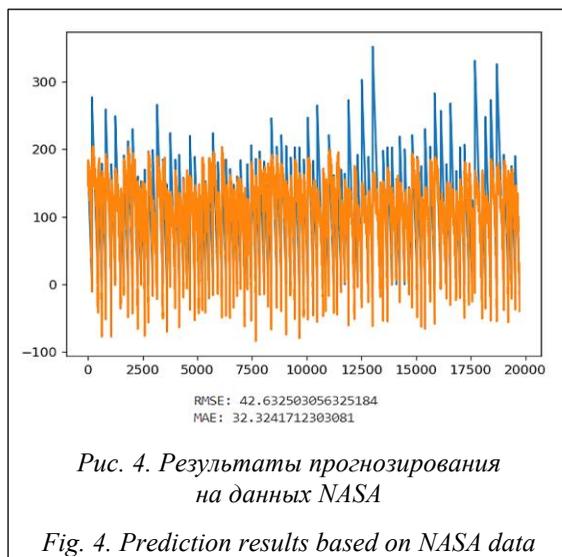
Таблица 2

**Пример параметров задачи прогнозирования**

Table 2

**Example of prediction task parameters**

Шаг	Параметр	Значение
1	<i>window size</i>	10
	<i>filename</i>	<i>train_FD001.csv</i>
	<i>separator</i>	;
2	<i>cdm</i>	<i>configurable data manipulator (.cache\all_input_continuous.pickle')</i>
	<i>cdm.add_pre_point_conversion_stage()</i>	<i>(drop_columns(['s3', 's4', 's8', 's9', 's13', 's19', 's21', 's22'])</i>
	<i>cdm.set_point_conversion_stage()</i>	<i>(convert to 1d input(window size))</i>
3	<i>bbpfm</i>	<i>linear_regression_point_forecast_model()</i>
	<i>List[point]</i>	<i>data</i>
	<i>num_of_partitions</i>	5
	<i>exclusions</i>	[]
4	<i>List[base_measurement]</i>	<i>[rmse(), mae()]</i>
	<i>List[point]</i>	<i>data</i>



Таким образом, фреймворк успешно загружает данные из источника, предоставляет широкий спектр инструментов для анализа и визуализации временных рядов, а также обладает гибкой архитектурой для добавления новых методов анализа и прогнозирования.

### Заключение

Предложено программное решение на Python для анализа временных рядов, где данные представлены в виде таблиц с разделителями. Данный фреймворк построен на модульной архитектуре, что позволяет расширять его возможности. Исследователи могут легко работать с различными форматами данных и добавлять новые методы анализа и прогнозирования временных рядов по своему усмотрению.

### Список литературы

1. Syreyshchikova N.V., Pimenov D.Yu., Mikolajczyk T., Moldovan L. Automation of production activities of an industrial enterprise based on the ERP system. *Procedia Manufacturing*, 2020, vol. 46, pp. 525–532. doi: 10.1016/j.promfg.2020.03.075.
2. Gómez-Pau Á., Riba J., Moreno-Eguilaz M. Time series RUL estimation of medium voltage connectors to ease predictive maintenance plans. *Appl. Sci.*, 2020, vol. 10, no. 24, art. 9041. doi: 10.3390/app10249041.
3. Safoklov B., Prokopenko D., Deniskin Y., Kostyshak M. Model of aircraft maintenance repair and overhaul using artificial neural networks. *Transportation Research Procedia*, 2022, vol. 63, pp. 1534–1543. doi: 10.1016/j.trpro.2022.06.165.
4. Cheng J.C., Chen W., Chen K., Wang Q. Data-driven predictive maintenance planning frame-work for MEP components based on BIM and IoT using machine learning algorithms. *Automation in Construction*, 2020, vol. 112, art. 103087.
5. Wahid A., Breslin J.G., Intizar M.A. TCRSCANet: Harnessing temporal convolutions and recurrent skip component for enhanced RUL estimation in mechanical systems. *Human-Centric Intelligent Systems*, 2024, vol. 4, pp. 1–24. doi: 10.1007/s44230-023-00060-0.
6. Berghout T., Benbouzid M. A systematic guide for predicting remaining useful life with machine learning. *Electronics*, 2022, vol. 11, no. 7, art. 1125. doi: 10.3390/electronics11071125.
7. Pérez-Chacón R., Asencio-Cortés G., Troncoso A., Martínez-Álvarez F. Pattern sequence-based algorithm for multivariate big data time series forecasting: Application to electricity consumption. *Future Generation Computer Systems*, 2024, vol. 154, pp. 397–412. doi: 10.1016/j.future.2023.12.021.
8. Cardoso F.C., Berri R.A., Borges E.N. et al. Echo state network and classical statistical techniques for time series forecasting: A review. *Knowledge-Based Systems*, 2024, vol. 293, art. 111639. doi: 10.1016/j.knsys.2024.111639.
9. Gómez-Pau Á., Riba J.-R., Moreno-Eguilaz M. Time series RUL estimation of medium voltage connectors to ease predictive maintenance plans. *Appl. Sci.*, 2020, vol. 10, no. 24, art. 9041. doi: 10.3390/app10249041.
10. Клячкин В.Н., Жуков Д.А. Прогнозирование состояния технического объекта с применением методов машинного обучения // Программные продукты и системы. 2019. Т. 32. № 2. С. 244–250. doi: 10.15827/0236-235X.126.244-250.
11. Sharma J., Mittal M.L., Soni G. Condition-based maintenance using machine learning and role of interpretability: A review. *IJSAEM*, 2024, vol. 15, pp. 1345–1360. doi: 10.1007/s13198-022-01843-7.
12. Соборнов Т.И., Ковалев И.О. Понятие и базовые методы предиктивной аналитики // Инновации. Наука. Образование. 2022. № 49. С. 1204–1207.
13. Терешина В.В. Применение систем предиктивной аналитики и предикативного моделирования // Инновационное развитие экономики. 2022. № 5. С. 243–246. doi: 10.51832/2223798420225243.
14. Щербаков М.В., Задиран К.С., Голубев А.В. и др. Фреймворк для анализа и прогнозирования временных рядов при разработке компонент проактивных систем поддержки принятия решений // Программные продукты и системы. 2017. Т. 30. № 3. С. 439–446. doi: 10.15827/0236-235X.119.439-446.
15. Задиран К.С. Новый метод прогнозирования остаточного ресурса оборудования для высокочастотных данных с неоднородной длительностью рабочих циклов // Изв. ЮФУ. Технические науки. 2023. № 4. С. 65–74.

## A framework for automating equipment remaining life prediction when building proactive decision support systems

Konstantin S. Zadiran<sup>1</sup>, Darya A. Volkova<sup>1</sup>✉, Maksim V. Shcherbakov<sup>1</sup>

<sup>1</sup> Volgograd State Technical University, Volgograd, 400005, Russian Federation

### For citation

Zadiran, K.S., Volkova, D.A., Shcherbakov, M.V. (2025) 'A framework for automating equipment remaining life prediction when building proactive decision support systems', *Software & Systems*, 38(1), pp. 100–107 (in Russ.). doi: 10.15827/0236-235X.149.100-107

### Article info

Received: 08.05.2024

After revision: 24.06.2024

Accepted: 23.07.2024

**Abstract.** The paper describes a framework for automating research in designing proactive decision support systems. In particular, it investigates the problem of time series analysis and prediction in order to create tools for automating prediction of various processes in asset management systems, including maintenance and repair. The authors identify the role of automation processes in asset management in these systems. They highlight the main factors influencing the choice of a program to implement a predictive analytics system. The authors propose an algorithm for solving the problem of predicting remaining useful life based on analyzing production asset data using artificial intelligence components. The proposed software solution is based on CRISP-DM. It is not a separate software product and can be embedded in existing software to support the possibility of modifying methods. The framework loads and preprocesses data, builds predicting models, predicts time series and evaluates the forecast. The developed framework has a flexible modular architecture for adding new methods of analysis and predicting. The possibility to redefine and implement own data sources, preprocessing stages, forecasting models and metrics on the basis of existing base classes extends the variability and increases the efficiency of its functioning. There is the example of using the framework to solve the problem of analyzing time series and determining equipment remaining useful life. It demonstrates the efficiency of the developed product in the field of data exploration and artificial intelligence.

**Keywords:** software framework, time series forecasting, time series analysis, decision support systems, RUL, predicting models

**Acknowledgements.** The research was funded by a grant from the Russian Science Foundation no. 24-21-00483, <https://rscf.ru/project/24-21-00483/>

### References

1. Syreyshchikova, N.V., Pimenov, D.Yu., Mikolajczyk, T., Moldovan, L. (2020) 'Automation of production activities of an industrial enterprise based on the ERP system', *Procedia Manufacturing*, 46, pp. 525–532. doi: 10.1016/j.promfg.2020.03.075.
2. Gómez-Pau, Á., Riba, J., Moreno-Eguilaz, M. (2020) 'Time series RUL estimation of medium voltage connectors to ease predictive maintenance plans', *Appl. Sci.*, 10(24), art. 9041. doi: 10.3390/app10249041.
3. Safoklov, B., Prokopenko, D., Deniskin, Y., Kostyshak, M. (2022) 'Model of aircraft maintenance repair and overhaul using artificial neural networks', *Transportation Research Procedia*, 63, pp. 1534–1543. doi: 10.1016/j.trpro.2022.06.165.
4. Cheng, J.C., Chen, W., Chen, K., Wang, Q. (2020) 'Data-driven predictive maintenance planning framework for MEP components based on BIM and IoT using machine learning algorithms', *Automation in Construction*, 112, art. 103087.
5. Wahid, A., Breslin, J.G., Intizar, M.A. (2024) 'TCRSCANet: Harnessing temporal convolutions and recurrent skip component for enhanced RUL estimation in mechanical systems', *Human-Centric Intelligent Systems*, 4, pp. 1–24. doi: 10.1007/s44230-023-00060-0.
6. Berghout, T., Benbouzid, M. (2022) 'A systematic guide for predicting remaining useful life with machine learning', *Electronics*, 11(7), art. 1125. doi: 10.3390/electronics11071125.
7. Pérez-Chacón, R., Asencio-Cortés, G., Troncoso, A., Martínez-Álvarez, F. (2024) 'Pattern sequence-based algorithm for multivariate big data time series forecasting: Application to electricity consumption', *Future Generation Computer Systems*, 154, pp. 397–412. doi: 10.1016/j.future.2023.12.021.
8. Cardoso, F.C., Berri, R.A., Borges, E.N. et al. (2024) 'Echo state network and classical statistical techniques for time series forecasting: A review', *Knowledge-Based Systems*, 293, art. 111639. doi: 10.1016/j.knosys.2024.111639.
9. Gómez-Pau, Á., Riba, J.-R., Moreno-Eguilaz, M. (2020) 'Time series RUL estimation of medium voltage connectors to ease predictive maintenance plans', *Appl. Sci.*, 10(24), art. 9041. doi: 10.3390/app10249041.
10. Klyachkin, V.N., Zhukov, D.A. (2019) 'Forecasting the state of a technical object using machine learning methods', *Software & Systems*, 32(2), pp. 244–250 (in Russ.). doi: 10.15827/0236-235X.126.244-250.
11. Sharma, J., Mittal, M.L., Soni, G. (2024) 'Condition-based maintenance using machine learning and role of interpretability: A review', *IJSAEM*, 15, pp. 1345–1360. doi: 10.1007/s13198-022-01843-7.
12. Sobornov, T.I., Kovalev, I.O. (2022) 'The concept and basic methods of predictive analytics', *Innovation. Science. Education*, (49), pp. 1204–1207 (in Russ.).

13. Tereshina, V.V. (2022) 'Application of predictive analytics and predicative modeling systems', *Innovative Economic Development*, (5), pp. 243–246 (in Russ.). doi: 10.51832/2223798420225243.

14. Shcherbakov, M.V., Zadiran, K.S., Golubev, A.V. et al. (2017) 'A framework for analysis and forecasting of time series in the development of proactive decision support system components', *Software & Systems*, 30(3), pp. 439–446 (in Russ.). doi: 10.15827/0236-235X.119.439-446.

15. Zadiran, K.S. (2023) 'A new method for predicting the remaining equipment life for high-frequency data with non-uniform duty cycles', *Izv. SFedU. Eng. Sci.*, (4), pp. 65–74 (in Russ.).

#### Авторы

**Задирин Константин Сергеевич**<sup>1</sup>,  
аспирант, konstantin.zadiran@gmail.com

**Волкова Дарья Александровна**<sup>1</sup>,  
студент, d\_volkova@vstu.ru

**Шербаков Максим Владимирович**<sup>1</sup>, д.т.н.,  
профессор, зав. кафедрой,  
maxim.shcherbakov@vstu.ru

#### Authors

**Konstantin S. Zadiran**<sup>1</sup>, Postgraduate Student,  
konstantin.zadiran@gmail.com

**Darya A. Volkova**<sup>1</sup>,  
Student, d\_volkova@vstu.ru

**Maksim V. Shcherbakov**<sup>1</sup>, Dr.Sci. (Engineering),  
Professor, Head of Chair,  
maxim.shcherbakov@vstu.ru

<sup>1</sup> Волгоградский государственный технический университет, г. Волгоград, 400005, Россия

<sup>1</sup> Volgograd State Technical University, Volgograd, 400005, Russian Federation

## Реализация модели нейро-нечеткой сети на основе байесовского логико-вероятностного подхода для решения задач аппроксимации

Г.А. Хамчичев <sup>1</sup>✉, Г.И. Кожомбердиева <sup>1</sup>

<sup>1</sup> Петербургский государственный университет путей сообщения Императора Александра I, г. Санкт-Петербург, 190031, Россия

### Ссылка для цитирования

Хамчичев Г.А., Кожомбердиева Г.И. Реализация модели нейро-нечеткой сети на основе байесовского логико-вероятностного подхода для решения задач аппроксимации // Программные продукты и системы. 2025. Т. 38. № 1. С. 108–121. doi: 10.15827/0236-235X.149.108-121

### Информация о статье

Группа специальностей ВАК: 1.2.1

Поступила в редакцию: 12.06.2024

После доработки: 01.08.2024

Принята к публикации: 12.08.2024

**Аннотация.** В статье описывается реализация как специализированного программного инструмента модели нейро-нечеткой сети (ННС), основанной на байесовской логико-вероятностной модели нечеткого вывода. Помимо собственно модели ННС, в нем реализованы адаптированные алгоритмы построения (алгоритм сеточного разбиения для генерации нечетких правил) и обучения (алгоритм обратного распространения ошибки и гибридный алгоритм) ННС, известные по использованию для сети ANFIS из пакета MATLAB. Программный инструмент разработан на языке Java, предназначен как для решения практических задач, так и для исследования эффективности и возможностей применения ННС на основе байесовской логико-вероятностной модели нечеткого вывода. Обсуждается опыт построения и обучения ННС предложенного типа с использованием разработанного авторами инструментария. Рассматриваются примеры создания и обучения ННС, ориентированных на решение конкретных задач аппроксимации функций нескольких переменных на основе реальных открытых и синтетических наборов данных. Сравниваются результаты, полученные с помощью разработанного инструмента и инструмента ANFIS из пакета MATLAB. Подтверждается способность предложенной модели ННС служить универсальным аппроксиматором сложных функциональных зависимостей, что позволяет судить об ее эффективности и возможностях применения в разных областях. Включение в программу различных метрик оценки качества аппроксимации позволяет комплексно оценивать качество обучения сети, ее точность, стабильность и адаптивность к новым данным. Ограничение доступа российских потребителей к коммерческому программному обеспечению зарубежных производителей повышает практическую значимость созданного на основе оригинальной модели ННС программного инструмента и делает разработку актуальной и полезной для широкого круга пользователей.

**Ключевые слова:** нейро-нечеткая сеть, байесовская логико-вероятностная модель нечеткого вывода, машинное обучение, обучение нейро-нечеткой сети, алгоритмы обучения, аппроксимация функций, ANFIS, Java

**Благодарности.** Авторы искренне признательны д.ф.-м.н. профессору М.М. Луценко за ценные консультации

**Введение.** Нейросетевой подход к решению задач анализа данных получил широкое распространение, а примеры использования нейронных сетей в системах искусственного интеллекта (включая так называемые генеративные модели) общеизвестны и выглядят впечатляюще. Эффективность применения нейронных сетей в подобных задачах обусловлена тем, что они являются универсальными аппроксиматорами, способными после обучения реализовать на заданном наборе данных любую непрерывную функциональную зависимость.

Менее известны среди непрофессионалов успехи применения в области искусственного интеллекта гибридных нейронных сетей, в которых нейросетевой подход распространяется на системы нечеткого вывода. В России одним из показательных примеров работ в этом направлении является монография [1]. Авторы данной статьи придерживаются классификации гибридных сетей, применяемой современными

зарубежными исследователями [2, 3] и признанной в нашей стране. Различают гибриды двух типов: нечеткие нейронные сети, структура которых сходна с классическими нейронными сетями, но сами нейроны являются нечеткими, то есть имеют нечеткие входы и выходы и/или нечеткие веса, и *нейро-нечеткие сети* (ННС).

Как нейронная сеть особого гибридного типа ННС, с одной стороны, представляет собой нечеткую систему, обладающую способностью формировать выходные сигналы на основе фаззификации значений входных сигналов, использования базы нечетких правил и механизма *нечеткого вывода* (НВ), а далее дефаззификации полученных результатов. С другой стороны, она организована в многослойную нейронную структуру, состоящую как из параметрических, так и из непараметрических слоев, нейроны в которых выполняют расчетные операции в соответствии с выбранной мо-

делью НВ. Благодаря адаптивной настройке значений параметров, задаваемых для нейронов параметрических слоев, сеть способна настраиваться таким образом, чтобы результирующие выходные значения соответствовали ожидаемым, что делает ее способной к обучению.

Известно, что объединение в гибридных системах нейросетевого и нечеткого подходов позволяет как привнести в системы нечеткого вывода способность к обучению и параллелизм вычислений, присущие нейронным сетям, так и усилить интеллектуальные возможности нейронных сетей использованием лингвистически интерпретируемых нечетких правил выработки решений и знаний экспертов. Однако, по мнению авторов, эти преимущества наиболее проявляются в нечетких нейронных сетях. В то же время в ННС аппарат НВ теряет свою семантическую природу, нечеткая терминология лишается своей смысловой значимости, а нечеткие правила упрощаются и приобретают, скорее, формальный характер: лингвистические переменные и функции принадлежности их термов используются просто как элементы модели нейронной сети. В результате такая нейросетевая структура становится самодовлеющей, способной самостоятельно (без создания базы правил экспертами) решать задачи, связанные с аппроксимацией функций нескольких переменных.

Количество новых гибридных сетей и их модификаций постоянно растет. Подробные актуальные обзоры, посвященные гибридным сетям различных типов, тенденциям, проблемам и перспективам их применения, представлены в [2, 3]. В [4] сделан обзор применяемых для таких сетей техник обучения. Имеются многочисленные примеры использования ННС, свидетельствующие об актуальности и интенсивности современных исследований и разработок в этой области искусственного интеллекта. ННС зарекомендовали себя как мощный инструмент, обеспечивающий эффективное решение задач прогнозирования, диагностики, автоматического управления и регулирования, а также других, связанных с анализом и обработкой данных. В ряде работ демонстрируются возможности решения задач прогнозирования временных рядов [5], состояния кабельных линий [6], энергопотребления зданий [7], поверхностного натяжения в материалах [8]. Пример использования ННС для решения задачи классификации показан в работе [9], а для решения задачи анализа и оценки проектных решений – в [10]. ННС успешно применяются в про-

мышленной сфере: в [11, 12] демонстрируется применение ННС в управлении и регулировании, а в [13] – для оптимизации параметров производственного процесса.

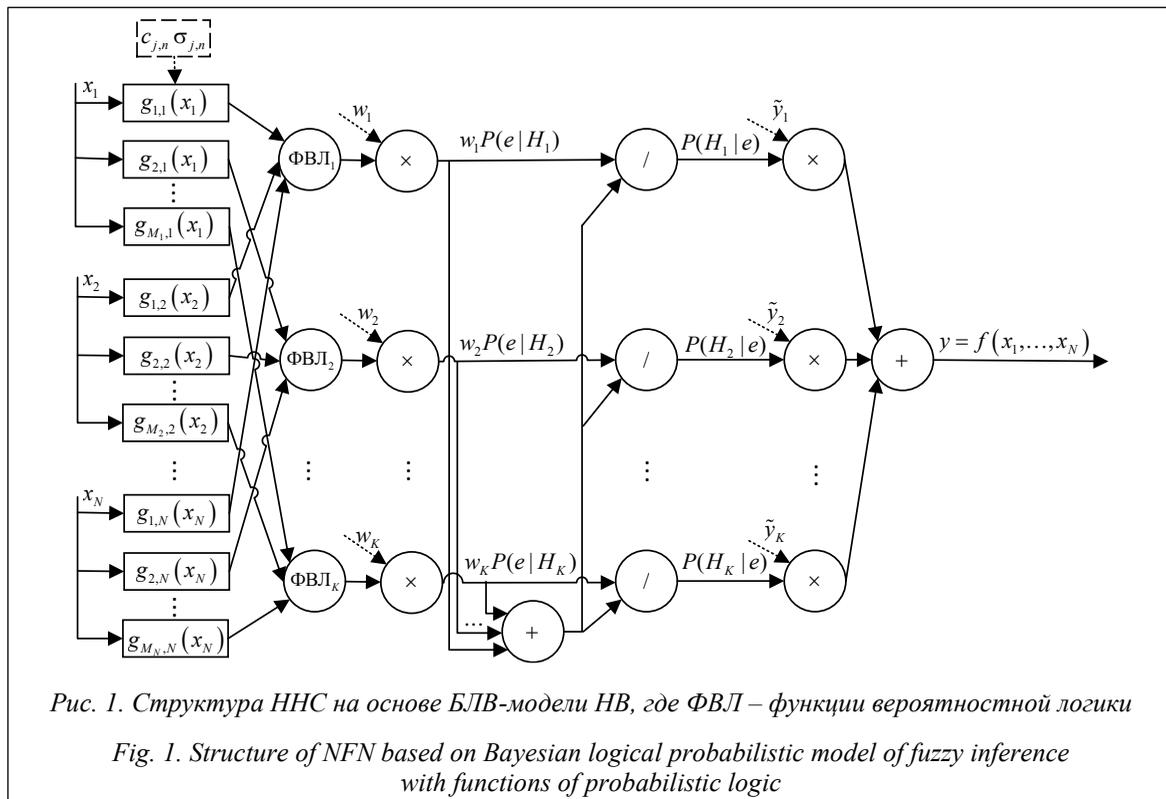
Структуре многослойной ННС на основе байесовской логико-вероятностной модели (БЛВ-модели) НВ и ее апробации посвящены работы [14–16]. Предложенная сетевая структура сопоставима с известными ННС Такаги–Сугено–Канга (TSK) и Ванга–Менделя [17], в частности, с широко применяемой моделью ANFIS (*Adaptive Neuro-Fuzzy Inference System*), также основанной на модели НВ TSK. Особую популярность ANFIS получила в связи с наличием ее готовой реализации в пакете MATLAB (<https://www.mathworks.com/help/fuzzy/neuro-adaptive-learning-and-anfis.html>).

В данной статье демонстрируется практическое применение разработанной программы для создания и обучения ННС на основе БЛВ-модели НВ, ориентированных на решение конкретных задач аппроксимации функций нескольких переменных на реальных и синтетических данных, что позволяет оценить их аппроксимационные способности и эффективность обучения в сравнении с инструментом для создания и обучения сетей ANFIS из MATLAB.

### ННС на основе БЛВ-модели

БЛВ-модель НВ предложена в [18]. Позднее она исследована и программно реализована авторами, получено Свидетельство о государственной регистрации программы для ЭВМ RU 2021662943. Опыт этой программной реализации использован при разработке программы для создания и обучения ННС на основе БЛВ-модели НВ. Нейросетевая интерпретация БЛВ-модели НВ и структурная схема ННС на ее основе представлены в [14, 15]. Остановимся кратко на функциональности нейронов, распределенных по слоям сети. Следует отметить, что в процессе программной реализации модели ННС учтены полученные предварительные результаты обучения и внесены полезные изменения в функциональное описание некоторых слоев сети.

Структура сети включает семь слоев, три из которых (первый, третий и шестой) являются параметрическими (рис. 1). Первый слой нейронов выполняет фазификацию каждой входной переменной  $x_n$ ,  $n = \overline{1, N}$ , то есть расчет значения функции принадлежности (ФП) для каждого  $j$ -го терма,  $j = \overline{1, M_n}$ , каждой  $n$ -й входной лингвистической переменной (ЛП), при-



чем, в отличие от представления структуры ННС в [14, 15], в программной реализации в качестве ФП используется частный случай ( $b = 1$ ) обобщенной гауссовой функции:

$$g_{j,n}(x_n; c_{j,n}, \sigma_{j,n}) = e^{-\frac{(x_n - c_{j,n})^{2b}}{\sigma_{j,n}^2}}, \quad (1)$$

которая при некоторых значениях показателя степени  $b$  может определять треугольную и трапецидальную ФП. Центры  $c = (c_{1,1}, \dots, c_{j,n}, \dots)$  и коэффициенты ширины  $\sigma = (\sigma_{1,1}, \dots, \sigma_{j,n}, \dots)$  гауссовых ФП являются параметрами первого слоя ННС.

Второй слой нейронов рассчитывает условные вероятности  $P(e|H_k)$ ,  $k = 1, \dots, K$ , на основе базы нечетких правил, трансформированной в набор функций вероятностной логики. Эти функции в качестве аргументов принимают значения ФП  $g_{j,n}$ , рассчитанные нейронами первого слоя сети. Отметим, что в качестве алгоритма генерации базы нечетких правил для программной реализации ННС выбран алгоритм сеточного разбиения, чем обусловлен простой вид формулы для расчета условной вероятности  $k$ -м нейроном второго слоя:

$$P(e|H_k) = \prod_{n=1}^N g_{j,n}(x_n; c_{j,n}, \sigma_{j,n}). \quad (2)$$

Нейроны третьего слоя выполняют взвешивание значений условных вероятностей параметрами-весами нечетких правил  $w = (w_1, \dots, w_k)$ .

Нейрон четвертого слоя суммирует взвешенные условные вероятности. Нейроны пятого слоя по формуле, основанной на формуле Байеса, формируют апостериорное распределение вероятностей:

$$P(H_k|e) = \frac{w_k P(e|H_k)}{\sum_{l=1}^K w_l P(e|H_l)}, \quad (3)$$

где  $K$  – число термов выходной ЛП (байесовских гипотез  $H_k$ ), соответствующих числу функций вероятностной логики (оценивающих степень истинности свидетельств в пользу каждой гипотезы);  $w_k$  – вес  $k$ -го правила,  $w_k \in [0; 1]$ .

Нейроны шестого слоя и нейрон-сумматор седьмого слоя отвечают за дефазификацию выходной переменной. Итоговая оценка  $\hat{y}$  значения выходной переменной  $y$  вычисляется по формуле

$$\hat{y} = \sum_{k=1}^K \tilde{y}_k P(H_k|e), \quad (4)$$

где  $P(H_k|e)$  – элемент апостериорного распределения вероятностей (3);  $\tilde{y}_k$  – характерное значение соответствующего терма выходной ЛП. При этом, в отличие от структуры ННС в [14, 15], в качестве параметров шестого слоя в программной реализации сети выступают не коэффициенты  $\alpha_k$ , предназначавшиеся для вычисления значений  $\tilde{y}_k$  как выпуклых комбинаций

граничных точек соответствующих термам интервалов на шкале выходной переменной, а непосредственно характерные значения  $\tilde{y} = (\tilde{y}_1, \dots, \tilde{y}_K)$  термов выходной ЛП. Результат работы сети – число  $\hat{y}$ , аппроксимирующее значение некоторой функции  $y = f(x_1, \dots, x_N)$ .

Таким образом, параметрами ННС на основе БЛВ-модели НВ, которые уточняются и обновляются в процессе обучения сети, являются:

центры  $c$  и коэффициенты ширины  $\sigma$  гауссовых ФП (первый слой нейронов);

веса нечетких правил  $w$  (третий слой нейронов);

характерные значения термов выходной ЛП  $\tilde{y}$  (шестой слой нейронов).

Компоненты векторов  $c$ ,  $\sigma$  и  $w$  отнесем к нелинейным параметрам, поскольку они используются в нелинейных функциях (1) и (3), а  $\tilde{y}$  – к линейным, так как они используются в линейной функции (4).

#### Алгоритмы, используемые при построении и обучении ННС на основе БЛВ-модели НВ

Для полноценной работы с ННС необходимы алгоритмы двух типов. Во-первых, алгоритмы начального формирования ее структуры, которые на основе имеющегося набора данных, выбранного вида ФП и указанного количества термов входных ЛП выполняют генерацию нечетких правил. Во-вторых, алгоритмы ее обучения, выполняющие обновление параметров выражений (1–4) таким образом, чтобы после каждой эпохи обучения сети (одного цикла полного прохода по обучающему набору данных) уменьшалось значение функции потерь:

$$L(y_i, \hat{y}_i; \mathbf{x}_i; \mathbf{c}, \mathbf{\sigma}, \mathbf{w}, \tilde{\mathbf{y}}) = \frac{1}{2} (y_i - \hat{y}_i)^2, \quad (5)$$

где  $y_i$  и  $\hat{y}_i$  представляют соответственно целевое и предсказанное значения для  $i$ -й строки данных  $\mathbf{x}_i = (x_1, \dots, x_N)$  из обучающего набора,  $i = \overline{1, Z}$ , где  $Z$  – число строк набора. Обучение заканчивается либо по завершении определенного числа эпох, либо по достижении заданной величины функции потерь  $L$ .

Для начального формирования структуры ННС был выбран алгоритм сеточного разбиения, поскольку он доступен в MATLAB и не-

сложен в реализации, а для обучения выбраны алгоритм обратного распространения ошибки (АОРО), основанный на градиентном спуске (ГС), и гибридный алгоритм (ГА) [16], адаптированные для учета особенностей ННС на основе БЛВ-модели НВ. Как показывает практика работы с ANFIS в MATLAB, ГА обеспечивает более быструю сходимость обучения, чем АОРО.

#### Алгоритм сеточного разбиения

Суть данного алгоритма состоит в равномерном распределении термов каждой  $n$ -й входной ЛП,  $n = \overline{1, N}$ , по шкале значений соответствующей входной переменной. Так, если ФП для термов входной ЛП определяются как гауссовы (1), то шкала делится на  $s_n + 1$  равных интервалов, где  $s_n$  – выбранное количество термов, и далее в качестве центров  $c_{j,n}$  выбираются значения точек соприкосновения этих интервалов, а в качестве коэффициентов ширины  $\sigma_{j,n}$  – длины интервалов. С помощью аналогичного разбиения шкалы выходной переменной на  $K + 1$  равных интервалов выбираются и  $\tilde{y}_k$  – характерные значения термов выходной ЛП. Количество характерных значений  $K$  соответствует количеству правил, посылка каждого из которых формируется как конъюнкция утверждений вида  $ЛП_n = \text{Терм}_j$ , где  $j$  – индекс термина  $n$ -й входной ЛП. При этом число  $K$  равно мощности  $N$ -кратного декартова произведения терм-множеств входных ЛП.

Алгоритм сеточного разбиения прост в реализации, имеет высокую скорость работы, однако создает избыточное количество правил, что увеличивает вычислительную сложность.

#### Алгоритм обратного распространения ошибки

Процесс обучения ННС направлен на минимизацию значения функции (5) путем итеративной коррекции параметров нейронов параметрических слоев ННС. В адаптации АОРО для его применения при обучении ННС на основе БЛВ-модели НВ ключевым элементом является выработка правила обновления параметров сети  $\mathbf{c}, \mathbf{\sigma}, \mathbf{w}, \tilde{\mathbf{y}}$  на основе ГС при минимизации функции  $L$ . Для обновления каждого параметра  $\theta$  ННС при ее обучении используется выражение

$$\theta^* = \theta - \eta_0 \frac{\partial L}{\partial \theta}, \quad (6)$$

где  $\theta^*$  – новое значение параметра;  $\theta$  – его текущее значение;  $\eta_0$  – темп изменения пара-

метра;  $\frac{\partial L}{\partial \theta}$  – частная производная функции (5)

по этому параметру, найденная с использованием цепного правила. В таблице 1 представлены конкретные варианты правила (6), используемые для обновления параметров ННС в ходе обучения.

Таблица 1

**Правила обновления параметров ННС**

Table 1

**Rules for updating NFN parameters**

Параметр ( $\theta$ )	Темп изменения параметра ( $\eta_\theta$ )	Ограничение на новое значение параметра ( $\theta^*$ )
$\tilde{y}_k$	$\eta_{\tilde{y}} = \frac{\lambda_{\tilde{y}}}{\sqrt{\sum_{k=1}^K \left(\frac{\partial L}{\partial \tilde{y}_k}\right)^2}}$	–
$w_k$	$\eta_w = \frac{\lambda_w}{\sqrt{\sum_{k=1}^K \left(\frac{\partial L}{\partial w_k}\right)^2}}$	$0 \leq w_k^* \leq 1$
$c_{j,n}$	$\eta_{c_{j,n}} = \frac{\lambda_c}{\sqrt{\sum_{n=1}^N \sum_{j=1}^{M_n} \left(\frac{\partial L}{\partial c_{j,n}}\right)^2}}$	–
$\sigma_{j,n}$	$\eta_{\sigma_{j,n}} = \frac{\lambda_\sigma}{\sqrt{\sum_{n=1}^N \sum_{j=1}^{M_n} \left(\frac{\partial L}{\partial \sigma_{j,n}}\right)^2}}$	$\sigma_{j,n}^* \geq 0$

Здесь  $\lambda_{\tilde{y}}, \lambda_w, \lambda_c, \lambda_\sigma$  – произвольно подбираемые размеры шага изменения соответствующего параметра сети в ходе обучения ННС. Наряду с числом используемых эпох обучения, пороговым значением функции потерь (5) и направлением градиента в ГС они рассматриваются как параметры обучения ННС.

Осуществляется контроль за тем, чтобы значения весов правил в процессе обучения не выходили за установленный диапазон [0, 1]. Для их изменения используется метод спроецированного градиентного спуска [19], а само новое значение веса определяется как  $w_k^* = \max(\min(w_k^*, 1), 0)$ .

Порядок обновления значений параметров  $\tilde{y}_k$  и  $w_k$  не имеет значения, так как производные  $\frac{\partial L}{\partial \tilde{y}_k}$  и  $\frac{\partial L}{\partial w_k}$  не зависят друг от друга. Это

позволяет выполнять их обновление параллельно и независимо, что обеспечивает гибкость в оптимизации функции потерь и упрощает реализацию алгоритма.

Преимуществами АОРО являются высокая эффективность для небольших сетей и простота реализации, однако при его применении возможны проблемы, связанные с попаданием в локальные минимумы, а также он довольно медленный при использовании на сетях с большим числом параметров.

**Гибридный алгоритм обучения**

Этот алгоритм сочетает в себе как ГС, так и метод наименьших квадратов. Алгоритм ГС применяется для тонкой настройки нелинейных параметров  $c, \sigma$  и  $w$ , обеспечивая эффективное снижение общей ошибки (5). В то же время метод наименьших квадратов идеально подходит для быстрой и точной корректировки линейных параметров  $\tilde{y}$  благодаря его способности к минимизации ошибок в линейно зависимых данных [17].

Для адаптации ГА часть уже реализованного алгоритма АОРО изменена таким образом, что при обратном проходе не выполняется изменение линейных параметров  $\tilde{y}$  с помощью ГС. Вместо этого в начале каждой эпохи обучения осуществляется прямой проход по всему обучающему набору данных и с помощью метода наименьших квадратов линейные параметры  $\tilde{y}$  обновляются путем решения системы линейных алгебраических уравнений

$$\tilde{y} = (P^T P)^{-1} P^T Y,$$

где вектор целевых значений  $Y$  содержит реальные выходные данные обучающего набора, а матрица  $P$  формируется на основе значений выходов пятого слоя сети – апостериорных вероятностей (3), рассчитанных для всех обучающих примеров.

Таким образом, метод наименьших квадратов направлен на нахождение оптимального вектора характерных значений  $\tilde{y}$ , минимизируя функцию потерь (5). После завершения этого шага происходит обновление нелинейных параметров  $c, \sigma$  и  $w$  в результате обратных проходов по каждой строке набора данных, как в АОРО.

Главными преимуществами ГА являются быстрая сходимость (особенно для набора данных с линейными зависимостями) и высокая точность обучения. Недостаток заключается в повышенной по сравнению с АОРО вычислительной сложности.

### Метрики оценки качества обученной сети

Для оценки точности и эффективности ННС в задачах аппроксимации в программе используются перечисленные далее известные метрики качества [19, 20].

– MSE (*Mean Squared Error* – среднеквадратическая ошибка). Измеряет среднюю квадратичную разницу между предсказанными  $\hat{y}_i$  и реальными значениями  $y_i$ , оценивая общую точность модели. Метрика особенно полезна для выявления больших ошибок, так как квадратичная функция штрафует их более сильно.

– RMSE (*Root Mean Squared Error* – корень из среднеквадратической ошибки). Обеспечивает оценку ошибки в тех же единицах, что и исходные данные. В дальнейшем будем ориентироваться на эту ошибку для сравнения предложенной модели ННС с ННС ANFIS из пакета MATLAB, поскольку в ней по умолчанию используется именно эта метрика.

– MAE (*Mean Absolute Error* – средняя абсолютная ошибка). Рассчитывает среднее абсолютное отклонение предсказанных значений  $\hat{y}_i$  от фактических  $y_i$ , являясь мерой линейной точности.

– MAPE (*Mean Absolute Percentage Error* – средняя абсолютная процентная ошибка). Выражает MAE в процентах, что полезно для оценки относительной точности без привязки к масштабу данных.

–  $R^2$  (коэффициент детерминации). Указывает на долю дисперсии зависимой переменной  $y$ , объясненную моделью, оценивая ее предсказательную силу.

Включение данных метрик в программу обусловлено их способностью к комплексной оценке точности, стабильности и адаптивности ННС к новым данным. Полезными для оценки качества обученной сети в контексте аппроксимации также являются скорректированный коэффициент детерминации  $R^2_{adj}$ , симметричная средняя абсолютная процентная ошибка SMAPE, взвешенная абсолютная процентная ошибка WAPE, средняя абсолютная масштабированная ошибка MASE. В дальнейшем планируется внедрить их в программу.

### Функциональные возможности программы

Разработанная программа обеспечивает построение и обучение ННС на основе БЛВ-модели НВ, которые позволяют эффективно решать задачи аппроксимации функций несколь-

ких переменных. Функциональные возможности программы и поддерживающий их реализацию графический интерфейс пользователя сопоставимы с инструментом ANFIS из пакета MATLAB.

Для разработки программы использован язык Java. Выбор обусловлен не только опытом работы авторов с ним, но и его кроссплатформенностью. К тому же последние версии среды исполнения значительно улучшили производительность (в том числе разработан высокопроизводительный вариант виртуальной машины Java – GraalVM), а язык предоставляет разработчику обширный API и множество библиотек, которые значительно упрощают разработку сложных приложений с продвинутым пользовательским интерфейсом.

### Построение сети

Программа поддерживает использование трех типов наборов данных: обучающий, валидационный и тестовый. Для построения сети используется обучающий набор. Пользователь указывает обучающий набор данных, количество термов входных ЛП и начальное значение весов  $w$ . На этой основе программа, используя алгоритм сеточного разбиения, автоматически создает нечеткие правила, необходимые для формирования структуры ННС, и определяет начальные значения параметров сети  $c$ ,  $\sigma$ ,  $\hat{y}$ .

В программе не поддерживается возможность редактирования правил, сгенерированных на этапе построения сети, причем правила содержат только просто реализуемую операцию конъюнкции. Однако на практике использования в сетях типа ANFIS лишь конъюнкции в нечетких правилах достаточно для получения точного результата в решаемых задачах, что подтверждается не только работой автора сети ANFIS [21], но и широким использованием этой модели в большом количестве приложений.

### Обучение сети

Для начала обучения нужно выбрать алгоритм обучения (АОРО или ГА), ввести целевое значение ошибки и количество эпох обучения сети. Во время обучения программа отображает динамические графики изменения ключевых метрик качества аппроксимации (<http://www.swsys.ru/uploaded/image/2025-1/1.jpg>), а также графики изменения значений параметров сети. Пользователь может в любой момент приостановить обучение сети, продолжить этот процесс или завершить его.

В программе реализована возможность настройки параметров обучения, позволяющая

отключать и включать обновление параметров для каждого параметрического слоя ННС, менять шаги их изменения, выбирать направление градиента и стратегию изменения значений параметров обучения (шагов изменения параметров сети) во время обучения, а также вручную настраивать параметры стратегий обучения (<http://www.swsys.ru/uploaded/image/2025-1/2.jpg>).

В программе поддерживаются две стратегии изменения значений параметров обучения. Одна из них – классическая, используемая в MATLAB. Суть ее в том, что при уменьшении ошибки на протяжении четырех эпох значение шага увеличивается, а если ошибка претерпевает две последовательные комбинации из одного увеличения и одного уменьшения, шаг уменьшается. Вторая стратегия – возврат значений параметров на определенное задаваемое число эпох назад при увеличении ошибки.

#### **Проверка сети**

После завершения обучения сеть подлежит проверке с использованием наборов данных для валидации и тестирования, что позволяет оценить ее способность к обобщению и точность предсказаний на новых данных. Валидационный набор данных, который не участвует в обучении сети, позволяет настраивать параметры обучения и предотвращает ее переобучение. Тестовый набор аналогично валидационному состоит из данных, не участвующих в обучении, и предназначен для окончательной проверки обобщающей способности обученной сети после тонкой настройки и валидации, имитируя ее реальное применение [20]. Для проверки сети нужно выбрать один из предварительно загруженных наборов данных. Если набор данных корректен, то результаты, включая значения ключевых метрик, отобразятся в специализированном текстовом поле графического интерфейса программы.

Пользователь может проверять работу сети на собственных данных, вводя значения входных переменных для получения дефазифицированного результата. Это позволяет проводить непосредственную проверку эффективности обученной ННС на специфичных для конкретного пользователя данных, упрощая процесс ее тестирования и адаптации под индивидуальные нужды. Результат расчета на основе текущих значений параметров сети отображается в специальном текстовом поле, включающем также и оценку ошибки вычисления в виде границ доверительного интервала  $\pm$  RMSE.

Пользователь имеет возможность сохранить текущее состояние сети со всеми значениями параметров, чтобы в дальнейшем можно было к нему вернуться. При экспорте и выборе опции «Совместимость» сеть будет экспортирована в формат, совместимый с авторской программой для решения задач нечеткого вывода на основе БЛВ-модели.

#### **Примеры создания сетей, ориентированных на решение практических задач аппроксимации**

Для исследования возможностей предложенной модели ННС выбрано несколько наборов данных: два набора (*Boston Housing Dataset* – ВН и *Auto MPG* – АМ), взятых с репозитория для исследователей в области машинного обучения и анализа данных Delve Datasets и UCI Machine Learning Repository, и два *синтетических набора данных* (СНД1 и СНД3), сформированных на основе вычисления заданной функции с последующим зашумлением.

#### **Характеристика наборов данных**

1. Набор данных ВН (<https://www.cs.toronto.edu/~delve/data/boston/bostonDetail.html>) содержит информацию о различных характеристиках жилых домов в окрестностях Бостона, включая среднее количество комнат в доме, уровень преступности, налоги на недвижимость и многое другое. Всего в наборе используются 14 характеристик, включая выходную переменную – сведения об их цене.

2. Набор данных АМ (<https://archive.ics.uci.edu/dataset/9/auto+mpg>) описывает различные характеристики автомобилей и содержит 9 переменных.

Наборы ВН и АМ имеют большую размерность, из-за чего их обработка проблематична даже в MATLAB: при попытке создать ННС на основе полного набора данных возникает нехватка памяти. Это связано с тем, что сложность ННС экспоненциально зависит от количества термов ЛП, построенных на основе входных переменных. Так как входные переменные в разной степени влияют на результат, было принято решение о сокращении размерности НД до двух входных переменных для АМ и трех входных для ВН, а также одной выходной переменной. Для этого была использована техника сокращения размерности пространства признаков на основе корреляционного анализа взаимосвязи между входными переменными и целевой переменной. Кроме того, были удалены строки с отсутствующими значениями переменных.

3. СНД1 сгенерирован на основе вычисления функции одной переменной  $f(x) = -4x^3 + 8x^2 + 5x - 1, x \in [-1,5; 3]$ . К результатам вычисления  $f(x)$  добавлен гауссов шум с распределением  $N(0;1)$  для имитации реальных условий измерений.

4. СНД3 сгенерирован на основе функции трех переменных  $f(x, y, z) = x^2 - 4y + 5z + 2, x, y, z \in [-4; 4]$ . К значениям функции  $f(x, y, z)$  добавлен гауссов шум с распределением  $N(0; 0,5)$  для имитации реальных условий измерений.

Итоговые характеристики используемых для создания и обучения ННС наборов данных приведены в таблице 2.

**Результаты обучения ННС**

Сравниваются результаты обучения ННС на основе БЛВ-модели НВ и ННС типа ANFIS с одинаковыми значениями количества термов входных ЛП, равными трем. Для описания термов ЛП используются гауссовы ФП (1). Для обучения применяется ГА, так как он демонстрирует наилучшую скорость обучения для обеих моделей.

Для сравнения качества обучения сетей используется метрика качества RMSE, поскольку, с одной стороны, она позволяет адекватно оценить шум в данных, а с другой, это единственная метрика, доступная как в MATLAB, так и в разработанной авторами программе. Зависимость RMSE от числа использованных эпох обучения на начальном периоде обучения, ко-

гда наблюдается активное уменьшение ошибки, иллюстрируется графиками. Итоговая сходимость, достигнутая на сравниваемых моделях, характеризуется минимальными изменениями RMSE в последующих эпохах и представлена в виде итоговых значений в соответствующих таблицах.

*Пример Boston Housing.* График на рисунке 2 и таблица 3 иллюстрируют изменение RMSE на первых 50 эпохах обучения ННС. Отмечается значительное улучшение RMSE после обучения обеих моделей ННС. График подтверждает их схожесть по эффективности и темпу обучения. Однако на тестовом наборе данных ННС на основе БЛВ-модели НВ демонстрирует большее значение RMSE, что указывает на более высокую способность ANFIS к обобщению на новых данных. Тем не менее близость значений RMSE на обучающем и валидационном наборах подчеркивает большой потенциал ННС на основе БЛВ-модели НВ в качестве инструмента аппроксимации сложных функциональных зависимостей.

На рисунке 3 приведены графики изменения значения RMSE по мере увеличения числа эпох обучения с использованием алгоритмов АОРО и ГА для обучения ННС ANFIS (а) и ННС на основе БЛВ-модели НВ (б). Графики показывают, что при использовании ГА для обучения ANFIS RMSE быстро уменьшается и стабилизируется на низком уровне, в то время

Таблица 2

**Характеристики наборов данных, используемых для создания и обучения ННС**

Table 2

**Characteristics of datasets for creating and training NFN**

Набор данных	Число входных переменных		Мощность набора данных			
	Исходное	Принятое	Всего строк	Обучающий	Валидационный	Тестовый
ВН	13	3	452	271	90	91
АМ	8	2	398	238	79	81
СНД1	1	1	200	120	40	40
СНД3	3	3	1000	600	200	200

Таблица 3

**Сравнение ННС по метрике RMSE для набора ВН**

Table 3

**Comparison of NFNs by RMSE metric for the Boston Housing dataset**

Тип ННС	ННС на основе БЛВ-модели НВ			ANFIS		
	обуч.	вал.	тест.	обуч.	вал.	тест.
До обучения	7,41	6,15	13,86	26,53	25,77	20,81
Лучшая эпоха	2,58	2,86	10,20	2,61	2,67	7,47

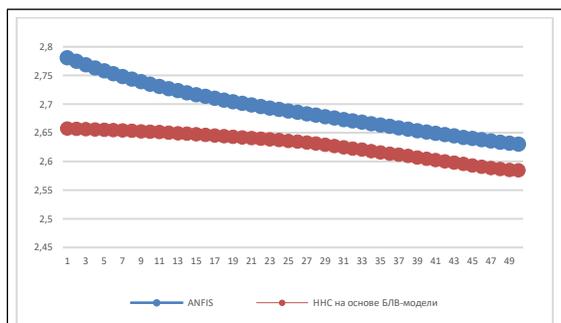


Рис. 2. График изменения RMSE для набора ВН на первых 50 эпохах обучения

Fig. 2. Diagram of RMSE variation for the Boston Housing dataset during the first 50 training epochs

как при использовании АОРО уменьшение происходит медленнее и не достигает таких низких значений. Аналогичная тенденция наблюдается и для ННС на основе БЛВ-модели НВ, где ГА также обеспечивает более быстрое и эффективное снижение RMSE по сравнению с АОРО. Это подтверждает эффективность применения ГА для обучения ННС, поскольку он объединяет преимущества ГС и метода наименьших квадратов, позволяющие более эффективно находить оптимальные значения нелинейных и линейных параметров нейронов сети в процессе ее обучения.

*Пример Auto MPG.* Результаты обучения ННС на этом наборе приведены в таблице 4.

Видно, что разница значений RMSE у обеих моделей ННС невелика как на начальном этапе, так и после обучения, что свидетельствует о сравнимом качестве аппроксимации данных обеими моделями. При этом ННС на основе БЛВ-модели НВ в отличие от ANFIS демонстрирует тенденцию уменьшения значения этого показателя в ходе обучения.

*Пример СНД1.* Результаты обучения сетей приведены в таблице 5.

Анализ результатов показывает, что обе модели ННС достигли значительного уменьшения RMSE по итогам обучения. График изменения RMSE (<http://www.swsys.ru/uploaded/image/2025-1/3.jpg>) демонстрирует стабильное улучшение качества аппроксимации для обеих моделей по мере их обучения. По завершении этапа обучения ННС на основе БЛВ-модели НВ достигает более высокого качества, чем ANFIS. Согласно данным таблицы 5, она показывает лучшие результаты на всех типах наборов данных, что подтверждает ее высокую адаптивность и способность к аппроксимации нелинейных зависимостей в условиях шума.

*Пример СНД3.* Результаты обучения сетей приведены в таблице 6.

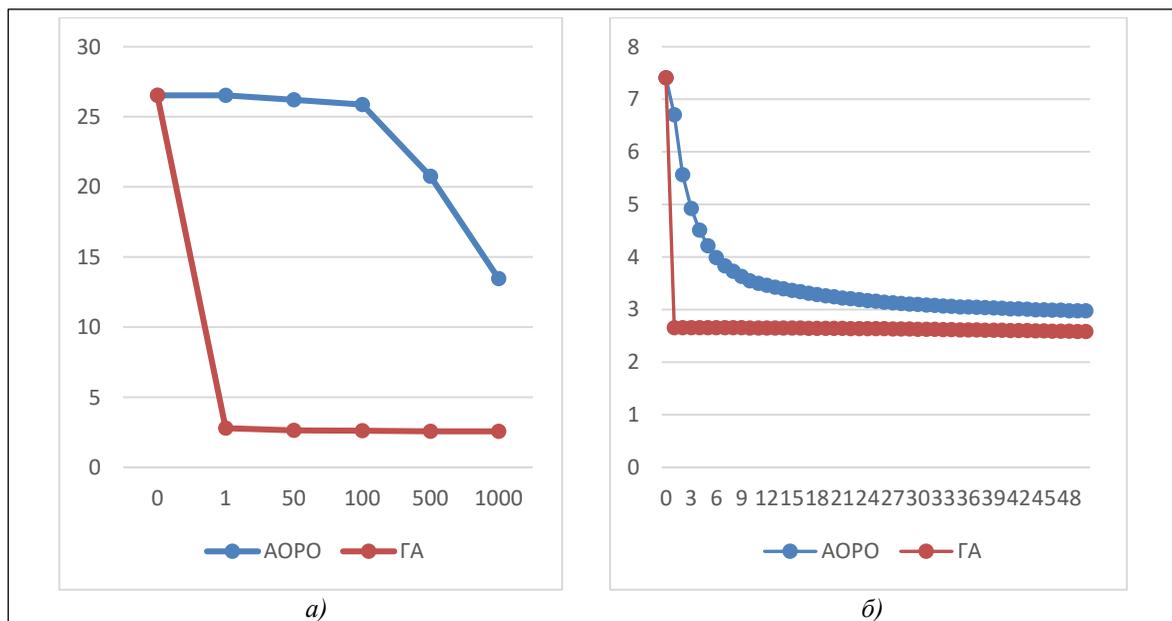


Рис. 3. Графики изменения RMSE на наборе ВН для алгоритмов обучения АОРО и ГА: а) для ННС ANFIS; б) для ННС на основе БЛВ-модели НВ

Fig. 3. Diagrams of RMSE variation on the Boston Housing dataset for error backpropagation and hybrid training algorithms: а) for ANFIS NFN; б) for NFN based on the Bayesian logical probabilistic model of fuzzy inference

На СНДЗ было проведено множество экспериментов с обучением ННС на основе БЛВ-модели НВ и подбора наилучших значений параметров обучения. В таблице 6 представлен результат с наименьшей полученной ошибкой, достигнутый при лимите эпох 2 500, начальных шагах для весов правил и параметров ФП 0,00005 и 0,0005 соответственно, при количестве термов входных ЛП, равном 3, и начальных весах правил 0,5. Параметры, установленные для обучения ННС ANFIS (количество термов входных ЛП – 3, количество эпох – 10 000), при этом не менялись.

График (<http://www.swsys.ru/uploaded/image/2025-1/4.jpg>) показывает, что ННС на основе БЛВ-модели НВ достигает стабильно низких значений ошибки быстрее, чем ANFIS, что свидетельствует в пользу предложенного подхода.

Из таблицы 6 видно, что ANFIS показывает лучшие результаты на тестовом наборе данных по сравнению с предложенной моделью ННС. Тем не менее результаты говорят и о высоком потенциале ННС на основе БЛВ-модели НВ как универсальном аппроксиматоре сложных функциональных зависимостей.

Несмотря на то, что для ННС на основе БЛВ-модели НВ при работе с СНДЗ не удалось получить значение RMSE, меньшее 0,7, скорее всего, это возможно при подборе подходящей стратегии обучения, поскольку ННС данного типа имеет больше параметрических слоев, чем ANFIS, и поэтому может быть настроена более гибко. Однако такая настройка ведет к возрастанию сложности обучения и требует тщательного подбора значений таких параметров обучения, как шаги изменения параметров сети  $\lambda_0$ . Это критически важно для обеспечения сходимости, следовательно, требует дополнительных исследований и экспериментов для достижения наилучших результатов.

По мнению авторов, ННС как гибридная система, объединяющая нейросетевой и нечеткий подходы, имеет особые перспективные достоинства в контексте исследований в области искусственного интеллекта. Поскольку в ходе обучения ННС меняются сгенерированные при ее создании параметры ФП термов ЛП, а также веса правил, по завершении ее обучения нейронная структура ННС реализует нечеткие правила, подобные правилам предметных экспертов и релевантные той задаче, на решение

Таблица 4

## Сравнение ННС по метрике RMSE для набора AM

Table 4

## Comparison of NFN using RMSE metric for the auto MPG

Тип ННС	ННС на основе БЛВ-модели НВ			ANFIS		
	обуч.	вал.	тест.	обуч.	вал.	тест.
Набор данных						
До обучения	13,915	14,234	13,428	24,95	24,426	24,609
Лучшая эпоха	4,04	4,2844	4,1762	4,0648	5,0002	4,2639

Таблица 5

## Сравнение ННС по метрике RMSE для СНД1

Table 5

## Comparison of NFN by RMSE metric for synthetic dataset 1

Тип ННС	ННС на основе БЛВ-модели НВ			ANFIS		
	обуч.	вал.	тест.	обуч.	вал.	тест.
Набор данных						
До обучения	9,914	12,569	9,367	8,238	10,238	7,599
Лучшая эпоха	3,726	4,986	4	3,941	4,313	3,758

Таблица 6

## Сравнение ННС по метрике RMSE для СНД3

Table 6

## Comparison of NFN by RMSE metric for synthetic dataset 3

Тип ННС	ННС на основе БЛВ-модели НВ			ANFIS		
	обуч.	вал.	тест.	обуч.	вал.	тест.
Набор данных						
До обучения	70,33	68,958	70,663	61,783	62,496	59,67
Лучшая эпоха	0,7153	0,7908	0,7220	0,5237	0,5203	0,5426

которой ориентирована данная ННС. После оптимизации эти правила могут быть извлечены из ННС и представлены в лингвистически интерпретируемом виде, что позволяет объяснить закономерности, выявленные ННС в предъявленном ей наборе данных. Это открывает возможности использования ННС в актуальном направлении исследований технологий искусственного интеллекта, известном как объяснимый искусственный интеллект и ставящем целью разработку интеллектуальных систем, решения которых могут быть объяснены человеком [22].

### Заключение

В статье рассмотрена модель ННС на основе БЛВ-модели НВ. Представлен разработанный авторами программный инструмент по созданию и обучению ННС этого типа, который предназначен для решения задач, сводимых к аппроксимации функциональных зависимостей, а также для исследования эффективности и возможностей использования ННС на основе БЛВ-модели НВ при решении таких задач. Эксперименты по построению ННС на основе БЛВ-модели НВ с использованием программы показали следующее.

– ННС, обученные с помощью алгоритма обучения ГА, достигают лучших результатов аппроксимации, чем при использовании АОРО, что связано с объединением в ГА преимуществ градиентного спуска и метода наименьших квадратов. При этом скорость обучения ННС зависит от степени линейности зависимости выходной переменной от входных. Если в данных присутствует нелинейность, то ННС обучается медленнее даже при использовании ГА.

– Шаги изменения нелинейных параметров ННС (весов нечетких правил и параметров ФП входных ЛП) должны быть сравнимы друг с другом и не превышать 0,01, а шаг для линейных параметров выходной ЛП должен изменяться в пределах 0,1–1, чтобы обновление параметров сети выполнялось согласованно и ошибка постоянно уменьшалась в процессе обучения.

В целом проведенные эксперименты позволяют сделать оптимистичные выводы о способности ННС этого типа к обучению и решению практических задач в областях, в которых обычно применяются ННС типа ANFIS, а также о сопоставимой эффективности этих сетей.

В дальнейшем планируется продолжить исследование способов согласования параметров обучения, в том числе использование стратегий с асинхронным изменением шагов обучения. Помимо этого, предполагается расширить набор доступных в программе типов ФП и алгоритмов генерации правил (например, основанных на методах кластеризации данных, а также на генетических алгоритмах) и обучения сети, метрик оценок качества обучения сети, а также дополнить список стратегий обучения. Кроме того, планируется задействовать методы оптимизации баз нечетких правил для улучшения производительности ННС и провести сравнение этих методов для оценки их эффективности при решении различных задач. Для этого могут быть задействованы генетические алгоритмы, методы роя частиц и другие подходы автоматической оптимизации нечетких правил.

Необходимо отметить, что у авторов вызывает интерес и такое возможное направление использования ННС на основе БЛВ-модели НВ, как объяснимый искусственный интеллект.

### Список литературы

1. Борисов В.В., Круглов В.В., Федулов А.С. Нечеткие модели и сети. М.: Горячая линия–Телеком, 2018. 285 с.
2. de Campos Souza P.V. Fuzzy neural networks and neuro-fuzzy networks: A review the main techniques and applications used in the literature. *Applied Soft Computing*, 2020, vol. 92, art. 106275. doi: 10.1016/j.asoc.2020.106275.
3. Talpur N., Abdulkadir S.J., Alhussian H., Hasan M.H., Aziz N., Bamhdi A. Deep Neuro-Fuzzy System application trends, challenges, and future perspectives: A systematic survey. *Artificial Intelligence Review*, 2023, vol. 56, pp. 865–913. doi: 10.1007/s10462-022-10188-3.
4. Karaboga D., Kaya E. Adaptive network based fuzzy inference system (ANFIS) training approaches: A comprehensive survey. *Artificial Intelligence Review*, 2019, vol. 52, pp. 2263–2293. doi: 10.1007/s10462-017-9610-2.
5. Ярушев С.А., Аверкин А.Н., Павлов В.Ю. Когнитивные гибридные системы поддержки принятия решений и прогнозирования // Программные продукты и системы. 2017. Т. 30. № 4. С. 632–642. doi: 10.15827/0236-235X.120.632-642.
6. Бирюлин В.И., Куделина Д.В., Серебровский В.И., Сафронов Р.И., Гнездилова Ю.П. Использование нечетких нейронных сетей для изучения процессов старения изоляции кабельных линий // Современные ресурсоэффективные технологии и технические средства в АПК: матер. науч.-практич. конф. 2021. С. 208–213.

7. Ghenai C., Al-Mufti O.A.A., Al-Isawi O.A.M., Amirah L.H.L., Merabet A. Short-term building electrical load forecasting using adaptive neuro-fuzzy inference system (ANFIS). *J. of Building Engineering*, 2022, vol. 52, art. 104323. doi: 10.1016/j.jobe.2022.104323.
8. Soleimani R., Saedi Dehaghani A.H. Insights into the estimation of surface tensions of mixtures based on designable green materials using an ensemble learning scheme. *Sci. Reports*, 2023, vol. 13, art. 14145. doi: 10.1038/s41598-023-41448-z.
9. Ravichandran, B.D., Keikhosrokiani, P. Classification of Covid-19 misinformation on social media based on neuro-fuzzy and neural network: A systematic review. *Neural Computing and Applications*, 2023, vol. 35, pp. 699–717. doi: 10.1007/s00521-022-07797-y.
10. Szafranko E., Srokosz P.E., Jurczak M., Śmieja M. Application of ANFIS in the preparation of expert opinions and evaluation of building design variants in the context of processing large amounts of data. *Automation in Construction*, 2022, vol. 133, art. 104045. doi: 10.1016/j.autcon.2021.104045.
11. Васильев С.Н., Кудинов Ю.И., Пашенко Ф.Ф., Дургарян И.С. и др. Интеллектуальные системы управления и нечеткие регуляторы. Часть II. Обучаемые нечеткие регуляторы, нечеткие ПИД-регуляторы // Датчики и системы. 2017. № 2. С. 3–12.
12. Дивеев А.И., Полтавский А.В., Алхатем А. Нейронечеткое управление процессом сушки пиломатериалов // Надежность и качество сложных систем. 2021. № 3. С. 93–97. doi: 10.21685/2307-4205-2021-3-12.
13. Mondal N., Banik S., Paul S., Sarkar S., Mandal S., Ghosh S. ANFIS-TLBO-based optimization of drilling parameters to minimize burr formation in aluminum 6061. *Multiscale and Multidisciplinary Modeling, Experiments and Design*, 2024, vol. 7, pp. 3607–3623. doi: 10.1007/s41939-024-00433-3.
14. Кожомбердиева Г.И., Бураков Д.П., Хамчичев Г.А. Структура нейро-нечеткой сети на основе байесовской логико-вероятностной модели // Мягкие измерения и вычисления. 2022. Т. 61. № 12. С. 52–64. doi: 10.36871/2618-9976.2022.12.004.
15. Kozhombardieva G.I., Burakov D.P., Khamchichev G.A. Neural network interpretation of Bayesian logical-probabilistic fuzzy inference model. *Proc. ISAIC*, vol. 1, pp. 50–56. doi: 10.5220/0011901700003612.
16. Хамчичев Г.А., Кожомбердиева Г.И. О настройке и возможностях обучения нейро-нечеткой сети на основе байесовской логико-вероятностной модели // Междунар. конф. по мягким вычислениям и измерениям. 2023. Т. 1. С. 29–33.
17. Осовский С. Нейронные сети для обработки информации; [пер. с польск.]. М.: Горячая линия–Телеком, 2018. 448 с.
18. Кожомбердиева Г.И. Байесовская логико-вероятностная модель нечеткого вывода // Междунар. конф. по мягким вычислениям и измерениям. 2019. Т. 1. С. 35–38.
19. Мэрфи К.П. Вероятностное машинное обучение: введение; [пер. с англ.]. М.: ДМК Пресс, 2022. 940 с.
20. Бишоп К.М. Распознавание образов и машинное обучение; [пер. с англ.]. СПб: ООО Диалектика. 2020. 960 с.
21. Jang J.-S.R. ANFIS: Adaptive-network-based fuzzy inference system. *IEEE Transactions on Systems, Man, and Cybernetics*, 1993, vol. 23, no. 3, pp. 665–685. doi: 10.1109/21.256541.
22. Аверкин А.Н., Ярушев С.А. Исследование развития систем объяснительного искусственного интеллекта // ИММВ-2022: сб. науч. тр. 2022. С. 127–134.

### Solving approximation problems: Implementing a neuro-fuzzy network model based on a Bayesian logical-probabilistic approach

Georgy A. Khamchichev <sup>1</sup>✉, Gulnara I. Kozhombardieva <sup>1</sup>

<sup>1</sup>Emperor Alexander I St. Petersburg State Transport University, St. Petersburg, 190031, Russian Federation

#### For citation

Khamchichev, G.A., Kozhombardieva (2025) ‘Solving approximation problems: Implementing a neuro-fuzzy network model based on a Bayesian logical-probabilistic approach’, *Software & Systems*, 38(1), pp. 108–121 (in Russ.). doi: 10.15827/0236-235X.149.108-121

#### Article info

Received: 12.06.2024

After revision: 01.08.2024

Accepted: 12.08.2024

**Abstract.** The paper describes the implementation of a neuro-fuzzy network (NFN) model based on Bayesian logical-probabilistic fuzzy inference model as a specialized software tool. In addition to the NFN model itself, it implements

adapted algorithms for constructing (grid partitioning algorithm for generating fuzzy rules) and training (error backpropagation algorithm and hybrid algorithm) NFN known from ANFIS network in MATLAB. The designed software tool is written in Java. It solves practical problems and investigates the effectiveness and applications of NFN based on the Bayesian logical-probabilistic fuzzy inference model. The authors discuss the experience of building and training the proposed NFN type using the developed toolkit. The authors also consider the creation and training examples of NFNs designed to solve specific approximation problems of multivariable functions based on real open and synthetic data sets. The paper compares the results obtained using the developed tool and the ANFIS tool from MATLAB. The authors confirmed that the proposed NFN model can be a universal approximator of complex functional dependencies. This confirms its efficiency and possibilities of using it in different fields. The inclusion of various approximation quality metrics in the program allows evaluating network training quality, accuracy, stability, and adaptability to new data comprehensively. Access restrictions for Russian users to foreign commercial software enhance the practical significance of the developed software tool based on the original NFN model; it makes the development relevant and useful for a wide range of users.

**Keywords:** neuro-fuzzy network, Bayesian logical-probabilistic model of fuzzy inference, machine learning, neuro-fuzzy network training, learning algorithms, function approximation, ANFIS, Java

**Acknowledgements.** The authors are sincerely grateful to Dr. Sci. (Phys. and Math.), Prof. M.M. Lutsenko for providing valuable consultations

## References

1. Borisov, V.V., Kruglov, V.V., Fedulov, A.S. (2018) *Fuzzy Models and Networks*. Moscow, 285 p. (in Russ.).
2. de Campos Souza, P.V. (2020) 'Fuzzy neural networks and neuro-fuzzy networks: A review of the main techniques and applications used in the literature', *Applied Soft Computing*, 92, art. 106275. doi: 10.1016/j.asoc.2020.106275.
3. Talpur, N., Abdulkadir, S.J., Alhussian, H., Hasan, M.H., Aziz, N., Bamhdi, A. (2023) 'Deep Neuro-Fuzzy System application trends, challenges, and future perspectives: A systematic survey', *Artificial Intelligence Review*, 56, pp. 865–913. doi: 10.1007/s10462-022-10188-3.
4. Karaboga, D., Kaya, E. (2019) 'Adaptive network based fuzzy inference system (ANFIS) training approaches: a comprehensive survey', *Artificial Intelligence Review*, 52, pp. 2263–2293. doi: 10.1007/s10462-017-9610-2.
5. Averkin, A.N., Yarushev, S.A., Pavlov, V.Yu. (2017) 'Cognitive hybrid systems for decision support and forecasting', *Software & Systems*, 30(4), pp. 632–642 (in Russ.). doi: 10.15827/0236-235X.120.632-642.
6. Biryulin, V.I., Kudelina, D.V., Serebrovsky, V.I., Safronov, R.I., Gnezdilova, Yu.P. (2021) 'The use of fuzzy neural networks for studying the aging processes of the cable lines insulation', *Proc. Sci.-Pract. Conf. Modern Resource-Efficient Technologies and Technical Means in Agriculture*, pp. 208–213 (in Russ.).
7. Ghenai, C., Al-Mufti, O.A.A., Al-Isawi, O.A.M., Amirah, L.H.L., Merabet, A. (2022) 'Short-term building electrical load forecasting using adaptive neuro-fuzzy inference system (ANFIS)', *J. of Building Engineering*, 52, art. 104323. doi: 10.1016/j.job.2022.104323.
8. Soleimani, R., Saeedi Dehaghani, A.H. (2023) 'Insights into the estimation of surface tensions of mixtures based on designable green materials using an ensemble learning scheme', *Sci. Reports*, 13, art. 14145. doi: 10.1038/s41598-023-41448-z.
9. Ravichandran, B.D., Keikhosrokiani, P. (2023) 'Classification of Covid-19 misinformation on social media based on neuro-fuzzy and neural network: A systematic review', *Neural Computing and Applications*, 35(1), pp. 699–717. doi: 10.1007/s00521-022-07797-y.
10. Szafranko, E., Srokosz, P.E., Jurezak, M., Śmieja, M. (2022) 'Application of ANFIS in the preparation of expert opinions and evaluation of building design variants in the context of processing large amounts of data', *Automation in Construction*, 133, art. 104045. doi: 10.1016/j.autcon.2021.104045.
11. Vassilyev, S.N., Kudinov, Yu.I., Pashchenko, F.F., Durgaryan, I.S. et al. (2017) 'Intelligent control systems and fuzzy controllers. Part II. Trained fuzzy controllers, fuzzy PID controllers', *Sensors and Systems*, (2), pp. 3–12 (in Russ.).
12. Diveev, A.I., Poltavskiy, A.V., Alhatem, A. (2021) 'Neuro-fuzzy control of the lumber drying process', *Reliability and Quality of Complex Systems*, (3), pp. 93–97 (in Russ.). doi: 10.21685/2307-4205-2021-3-12.
13. Mondal, N., Banik, S., Paul, S., Sarkar, S., Mandal, S., Ghosh, S. (2024) 'ANFIS-TLBO-based optimization of drilling parameters to minimize burr formation in aluminum 6061', *Multiscale and Multidisciplinary Modeling, Experiments and Design*, 7, pp. 3607–3623. doi: 10.1007/s41939-024-00433-3.
14. Kozhombierdieva, G.I., Burakov, D.P., Khamchichev, G.A. (2022) 'The structure of a neuro-fuzzy network based on Bayesian logical-probabilistic model', *Soft Measurements and Computing*, 61(12), pp. 52–64 (in Russ.). doi: 10.36871/2618-9976.2022.12.004.
15. Kozhombierdieva, G.I., Burakov, D.P., Khamchichev, G.A. (2022) 'Neural network interpretation of Bayesian logical-probabilistic fuzzy inference model', *Proc. ISAIC*, 1, pp. 50–56. doi: 10.5220/0011901700003612.
16. Khamchichev, G.A., Kozhombierdieva, G.I. (2023) 'On setting up and the training capabilities of a neuro-fuzzy network based on a Bayesian logical-probabilistic model', *Proc. Int. Conf. SCM*, 1, pp. 29–33 (in Russ.).
17. Osowski, S. (2000) *Sieci Neuronowe Do Przetwarzania Informacji*. 420 p. (Russ. ed.: (2018) Moscow, 448 p.).
18. Kozhombierdieva, G.I. (2019) 'Bayesian logical-probabilistic model of fuzzy inference', *Proc. Int. Conf. SCM*, 1, pp. 35–38 (in Russ.).
19. Murphy, K.P. (2022) *Probabilistic Machine Learning: An Introduction*. Mit Press, 864 p. (Russ. ed.: (2022) Moscow, 940 p.).

20. Bishop, C.M. (2006) *Pattern Recognition and Machine Learning*. Springer Publ, 738 p. (Russ. ed.: (2020) St. Petersburg, 940 p.).
21. Jang, J.-S.R. (1993) ‘ANFIS: Adaptive-network-based fuzzy inference system’, *IEEE Transactions on Systems, Man, and Cybernetics*, 23(3), pp. 665–685. doi: 10.1109/21.256541.
22. Averkin, A.N., Yarushev, S.A. (2022) ‘Study of the development of explainable artificial intelligence systems’, *Proc. IMMV-2022*, pp. 127–134 (in Russ.).

**Авторы**

**Хамчичев Георгий Алексеевич**<sup>1</sup>,  
аспирант, gyqn@yandex.ru  
**Кожомбердиева Гульнара Исмановна**<sup>1</sup>,  
к.т.н., доцент,  
kgi-liizht@yandex.ru

<sup>1</sup> Петербургский государственный университет  
путей сообщения Императора Александра I,  
г. Санкт-Петербург, 190031, Россия

**Authors**

**Georgy A. Khamchichev**<sup>1</sup>,  
Postgraduate Student, gyqn@yandex.ru  
**Gulnara I. Kozhomberdieva**<sup>1</sup>,  
Cand. of Sci. (Engineering),  
Associate Professor, kgi-liizht@yandex.ru

<sup>1</sup> Emperor Alexander I St. Petersburg State  
Transport University, St. Petersburg,  
190031, Russian Federation

## Определение принципов проектирования отечественной платформы обмена научными датасетами

К.В. Гарев<sup>1, 2</sup>✉

<sup>1</sup> Департамент информационных технологий Банка России, г. Москва, 107016, Россия

<sup>2</sup> Национальный исследовательский центр «Курчатовский институт» – НИИСИ, г. Москва, 117218, Россия

### Ссылка для цитирования

Гарев К.В. Определение принципов проектирования отечественной платформы обмена научными датасетами. 2025. Т. 38. № 1. С. 122–133. doi: 10.15827/0236-235X.149.122-133

### Информация о статье

Группа специальностей ВАК: 2.3.1

Поступила в редакцию: 01.07.2024

После доработки: 26.07.2024

Принята к публикации: 31.07.2024

**Аннотация.** В работе рассматриваются предпосылки и предложения по созданию отечественной платформы обмена научными датасетами в контексте глобальных тенденций, связанных с развитием открытой науки и распространением FAIR-принципов. На основе анализа зарубежных инициатив (EUDAT, EOSC, DataONE, Dryad, Zenodo) выявляются ключевые проблемы, затрудняющие эффективное использование, сохранение и повторное применение научных данных. К таким проблемам относятся отсутствие единых регламентов для описания датасетов, разрозненность инфраструктурных решений, недостаточная межплатформенная совместимость, а также сложность обеспечения воспроизводимости исследований. Особое внимание уделяется роли профессионального сообщества, важности формирования среды для обмена опытом, проведения междисциплинарных проектов и совершенствования навыков работы с большими массивами данных. При этом подчеркивается необходимость систематизации работы с научными данными и унификации требований к их сбору, хранению, обработке и представлению, что позволит повысить прозрачность исследовательских процессов. В статье обоснована целесообразность внедрения механизмов распределенного хранения, федеративной аутентификации и высокопроизводительных вычислительных ресурсов, способных удовлетворять потребности отечественного научного сообщества. Выдвигаются предложения по проектированию единой платформы обмена научными датасетами – от разработки методологических регламентов и стандартов взаимодействия с внешними системами до принципов интеграции инструментов аналитики и обеспечения надежной защиты данных.

**Ключевые слова:** датахабы, научные данные, научные датасеты, цифровые платформы обмена научными данными, открытая наука, FAIR, EUDAT, EOSC, DataONE, Dryad, Zenodo

**Введение.** В международном научном сообществе постепенно укрепляется и развивается концепция Open Science, призванная обеспечить более широкое и открытое распространение научных данных, результатов исследований и методологических разработок. Европейская комиссия подчеркивает, что открытое взаимодействие в научно-исследовательской среде стимулирует инновации и экономический рост, а также способствует решению крупных социально-экономических задач [1]. В свою очередь, Организация экономического сотрудничества и развития (OECD) отмечает, что доступность и повторное использование исследовательской информации являются важнейшими условиями повышения эффективности научного сотрудничества и ускорения научно-технического прогресса [2]. Основная идея данной концепции заключается в повышении прозрачности научных исследований и их воспроизводимости, что, как предполагается, может содействовать ускорению прогресса в различных обла-

стях знаний и укреплению междисциплинарных связей. Несмотря на то, что потенциальные эффекты от внедрения принципов Open Science выглядят многообещающими, в мировом сообществе все еще ведутся дискуссии о реальном масштабе и траекториях будущих изменений, обусловленных данной концепцией.

Неотъемлемой составляющей современных подходов к организации научных данных является концепция FAIR-принципов (*Findable, Accessible, Interoperable, Reusable*). Ее авторы акцентируют внимание на необходимости обеспечения удобного поиска, доступности, совместимости и возможности повторного использования данных в широком спектре научных дисциплин, что должно способствовать повышению эффективности обмена научными данными и прозрачности процесса получения исследовательских результатов [3].

Вышеупомянутые принципы реализуются путем создания различных цифровых инфраструктур, в том числе международных науч-

ных датахабов. К числу известных ресурсов, формирующих основу для открытого обмена данными и предоставляющих доступ к широкому спектру научных датасетов, относятся такие, как EUDAT (*European Data Infrastructure*), EOSC (*European Open Science Cloud*), DataONE (*Data Observation Network for Earth*), Dryad, Zenodo и др. Их создание и развитие во многом стимулируются как государственными, так и неправительственными инициативами, стремящимися упростить доступ к актуальным наборам данных и повысить эффективность научного взаимодействия.

В России данное направление находится в стадии формирования: несмотря на появление отечественных баз научных данных, до последнего времени отсутствовал системный подход к созданию универсальных платформ для обмена научными датасетами. Новая стратегия научно-технологического развития Российской Федерации декларирует необходимость разработки и внедрения таких платформ, указывая на их потенциальную значимость для обеспечения технологической независимости и повышения глобальной конкурентоспособности отечественных научных организаций. Вместе с тем реализация подобных проектов может потребовать существенного комплексного анализа потребностей различных научных направлений, обеспечения дополнительных ресурсов, а также принятия соответствующих нормативных актов.

Данная статья посвящена рассмотрению существующих инициатив и подходов в области создания научных датахабов и возможных направлений их создания и развития.

#### **Анализ зарубежного опыта построения и использования систем управления научными данными**

В мире продолжают создаваться и развиваться системы управления научными датасетами (научными датахабами), позволяющие осуществлять взаимообмен результатами экспериментов, а также однозначно ссылаться на них. Системы необходимы для консолидации и долгосрочного хранения данных, предоставления доступа к ним исследовательских сообществ.

Наличие подобных платформ дает возможность систематизировать подходы к управлению научными данными, что делает их доступными для поиска, легко интегрируемыми в разные системы и пригодными для повторного

использования. Реализация FAIR-принципов в научных датахабах способствует формированию глобальной инфраструктуры данных. Кроме того, в международном научном сообществе принято делиться научными датасетами в открытом доступе. Этот подход позволяет обеспечить воспроизводимость результатов, подтвердить их достоверность и исключить риск фальсификации данных. Как показано в [4], открытый доступ к датасетам является важным элементом современной научной практики, поскольку предоставляет возможность независимой проверки выводов и использования данных для новых исследований. Данные, размещенные на специализированных платформах для сбора, хранения, обработки и представления научных данных, сопровождаются описанием метаданных и стандартами цитирования, что гарантирует корректное использование и ссылку на источник.

Датахабы условно можно разделить на следующие группы:

- по областям наук: специализированные, профильные, междисциплинарные;
- по назначению: хранилища данных, аналитические платформы, каталоги справочников и стандартов;
- по организации инфраструктуры: облачные, локальные, гибридные.

Рассмотрим примеры таких систем.

EUDAT (<https://www.eudat.eu/>) – крупномасштабная европейская инфраструктура, предназначенная для управления научными данными и предоставления исследовательским сообществам инструментов для хранения, обработки и обмена данными. Ее основная цель заключается в создании единой унифицированной платформы для обеспечения FAIR-принципов и поддержки междисциплинарных исследований. EUDAT предоставляет широкий спектр функций и технологий для управления научными данными, обеспечивая их долгосрочное хранение, обработку, доступ и интеграцию. Предлагает комплекс взаимосвязанных сервисов для управления научными данными: B2SAFE обеспечивает их репликацию, повышая надежность и длительное сохранение; B2STAGE предоставляет высокопроизводительные средства (GridFTP, HTTP) для передачи больших объемов данных на внешние вычислительные ресурсы; B2SHARE, основанный на платформе Invenio, упрощает загрузку, аннотирование и обмен исследовательскими данными с присвоением постоянных идентификаторов; B2FIND обеспечивает единый ка-

талог метаданных и расширенный поиск; B2DROP служит для хранения, синхронизации и совместного использования динамических данных; B2ACCESS поддерживает федеративную аутентификацию и авторизацию через OpenID, SAML, x.509 и OAuth 2, а B2HANDLE, работающий на основе системы Handle, позволяет регистрировать постоянные идентификаторы и гарантирует доступность и цитируемость данных. EUDAT активно применяется в различных предметных областях науки. В исследовании [5] приведен пример использования данной платформы как основы для управления научными данными в биомедицине на базе геораспределенной высокопроизводительной инфраструктуры. Проект ESCAPE Dark Matter Science Project, реализуемый в рамках инициативы EOSC Future, использует инфраструктуру EUDAT для управления и обработки данных в исследовании темной материи, включая сервис для масштабируемого хранения и обработки данных, сервис для авторизации и управления доступом, а также платформу на базе JupyterHub для проведения воспроизводимого анализа данных и обеспечения соответствия принципам FAIR [6].

EOSC (<https://eosc.eu/>) – инициатива Европейской комиссии, направленная на создание федеративной и открытой мультидисциплинарной среды, где европейские исследователи, инновационные компании и граждане могут публиковать, находить и повторно использовать данные, инструменты и сервисы для научных, инновационных и образовательных целей. Техническая реализация EOSC основана на федеративной архитектуре, которая объединяет существующие и новые научные данные и сервисы в единую интероперабельную экосистему, устраняя фрагментацию и предоставляя пользователям единую точку доступа к ресурсам. Для обеспечения взаимодействия компонентов применяются EOSC Interoperability Framework [7] с рекомендациями по технической, семантической и организационной интероперабельности, а также федеративная система аутентификации и авторизации с интеграцией через eduGAIN (<https://edugain.org/>), что позволяет безопасно использовать институциональные учетные данные. Через EOSC Portal исследователям доступен каталог сервисов и данных со стандартизированными метаданными, упрощающий поиск и интеграцию различных ресурсов, включая ПО и вычислительные мощности. В качестве примера использования можно привести исследование, где в рамках Большой латиноамери-

канской обсерватории (LAGO) была проведена масштабная интеграция симуляционного фреймворка ARTI в облачные сервисы EOSC-Synergy (проект в контексте EOSC, направленный на увеличение международной сети обмена научными данными), что позволило обеспечить централизованное хранение, обработку и моделирование больших объемов данных о космических лучах, поступающих с распределенной сети датчиков в различных регионах и на высотах до 5 500 м [8].

DataONE (<https://www.dataone.org/>) – сеть репозитория данных, призванная сохранять и предоставлять доступ к данным в области наук о Земле и окружающей среде. DataONE функционирует на базе распределенной архитектуры с координирующими узлами и узлами-участниками, которые связаны стандартизированными RESTful API. Координирующие узлы отвечают за регистрацию репозитория, управление метаданными, предоставление сервисов поиска и обнаружения данных, а узлы-участники представляют собой репозитории, чьи коллекции индексируются и реплицируются для повышения доступности и надежности. Система устойчивых идентификаторов обеспечивает однозначную идентификацию объектов данных, а метаданные описываются унифицированными схемами, что упрощает их поиск и повышает интероперабельность. Механизмы репликации между узлами-участниками гарантируют долговременное хранение данных и устойчивость к сбоям, обеспечивая широкий доступ к информации. В статье [9] авторы описывают, как DataONE предоставляет инфраструктуру для хранения, обмена и управления биологическими и экологическими данными, что способствует инновациям в этих областях, акцентируя внимание на интеграции сервисов DataONE в исследовательские процессы.

Dryad (<https://datadryad.org/>) – международный открытый репозиторий, предоставляющий исследователям платформу для публикации и сохранения данных, лежащих в основе научных и медицинских публикаций. Dryad использует открытую платформу Stash (на базе Ruby on Rails) для управления процессом публикации научных данных, обеспечивая загрузку, описание и распространение наборов данных в различных форматах с применением адаптированной схемы метаданных Dublin Core. Долговременное хранение данных осуществляется в сотрудничестве с Merritt, репозиторием Калифорнийской цифровой библиотеки, который гарантирует надежное битовое хранение и управ-

ляет копиями файлов, размещенными у нескольких облачных провайдеров в двух географических регионах. Каждый опубликованный набор получает DOI от DataCite для постоянного доступа и цитирования, а поддержка Schema.org повышает видимость данных в поисковых системах. Команда кураторов Dryad проверяет загружаемые материалы на соответствие стандартам качества и этики, а партнерство с Zenodo, установленное в 2019 году, связывает опубликованное в Dryad исследовательское ПО с репозиторием CERN. Аутентификация через ORCID способствует точному учету авторства и вклада ученых [10].

Zenodo (<https://github.com/zenodo/zenodo>) – универсальный открытый репозиторий, разрабатываемый в рамках программы OpenAIRE и управляемый CERN, предоставляет исследователям возможности по хранению и распространению статей, наборов данных, ПО и других цифровых объектов. Техническая инфраструктура платформы основана на ПО Invenio, созданном в CERN, что обеспечивает масштабируемость и поддержку множества форматов данных. Размещаясь в высокопроизводительном вычислительном центре CERN в Швейцарии, Zenodo гарантирует надежное долгосрочное хранение материалов; каждому загружаемому объекту присваивается DOI через DataCite, способствуя его обнаружению и корректному цитированию. Гибкая политика лицензирования и интеграция с GitHub позволяют автоматически сохранять версии ПО и обеспечивать их цитируемость, а запуск репозитория в 2013 году сделал его широко востребованной платформой для международного научного сообщества.

В качестве иллюстрации одного из примеров использования можно привести долгосрочное археологическое исследование в Южной Румынии [11]. Платформа Zenodo была интегрирована в рабочий процесс для хранения и обмена данными, полученными с помощью дронов, включая ортофотоснимки, цифровые модели рельефа и 3D-модели археологических памятников. Использование Zenodo позволило обеспечить долгосрочное хранение больших объемов пространственных данных, предоставление уникальных DOI для удобного цитирования и соблюдение принципов открытого доступа.

Рассмотренные международные платформы демонстрируют ряд общих принципов.

Во-первых, каждая из них делает ставку на использование устойчивых идентификаторов

для обеспечения цитируемости и глобальной прослеживаемости данных (DOI, Handle), что упрощает их поиск и интеграцию.

Во-вторых, ключевым аспектом во всех системах выступает работа с качественными и стандартизированными метаданными (Dublin Core, Schema.org и др.), обеспечивающими высокую интероперабельность между различными областями исследований.

В-третьих, данные платформы ориентированы на соблюдение принципов FAIR.

Повсеместно применяется федеративный подход к аутентификации (SAML, OpenID, eduGAIN и т.д.), что повышает безопасность и упрощает управление доступом. Немаловажны и механизмы репликации и распределенного хранения (например, интеграция с облачными провайдерами и национальными репозиториями), которые обеспечивают устойчивость к сбоям и долговременное сохранение данных.

Таким образом, совокупность этих характеристик закладывает основу для эффективного функционирования датахабов.

#### **Отечественные решения для работы с научными данными**

По данным портала «Научно-технологическая инфраструктура Российской Федерации», в России функционируют более 400 уникальных научных установок, в рамках использования которых так или иначе могут генерироваться научные датасеты. Использование физических научных установок увеличивает издержки, связанные с закупкой расходных материалов и амортизацией инфраструктуры, а также влияет на время проведения исследования за счет физической очереди. Практика переиспользования научных датасетов, напротив, позволяет произвести необходимые вычисления зачастую без необходимости получения физического доступа к научной установке.

В России отсутствуют централизованные датахабы, аналогичные зарубежным. При этом существуют репозитории научных данных, созданные отдельными институтами. Например, Институт космических исследований РАН предоставляет доступ к данным через Центр коллективного пользования «ИКИ-Мониторинг», который обеспечивает хранение и обработку спутниковых наблюдений для изучения и мониторинга окружающей среды [12].

Одним из примечательных является проект «Инфраструктура научно-исследовательских данных» (ИНИД) (<https://data.rcsi.science/about/>),

разработанный Центром перспективных управленческих решений. Изначально проект ИНИД был создан для содействия цифровой трансформации науки и совершенствования взаимодействия между учеными и органами государственной власти. Однако реализованный функционал позволяет рассматривать данную платформу как сервисную основу для развития отечественного междисциплинарного научного датахаба на базе Российского центра научной информации, являющегося текущим оператором ИНИД.

Иная ситуация с платформами, направленными на работу не с научными датасетами, а с научно-технической информацией, являющейся ключевым ресурсом для организации и управления научной и инновационной деятельностью. Она включает в себя сведения о научных исследованиях, технологических разработках, патентах, публикациях, конференциях и других аспектах, отражающих динамику научно-технического прогресса. Эффективное управление этим видом информации является критически важным для обеспечения ее доступности, анализа и использования в процессе принятия решений. В России функционируют несколько таких систем, предназначенных для поддержки научных исследований, технологического развития и обеспечения обмена информацией.

Единая государственная информационная система учета результатов научно-исследовательских, опытно-конструкторских и технологических работ гражданского назначения – централизованная платформа для учета данных о научных исследованиях и разработках в России, охватывающая все этапы проекта от запуска до итоговых результатов. Для каждой работы формируется регистрационная карта с основными сведениями, а исполнители обязаны предоставлять информацию об итогах и созданных интеллектуальных продуктах, полученных за счет средств федерального бюджета. Такой подход обеспечивает наличие единого ресурса для анализа, планирования и мониторинга научной деятельности в стране.

Поисковая платформа Роспатента – цифровая система для доступа к патентной информации, позволяющая осуществлять поиск и анализ сведений об объектах интеллектуальной собственности (патенты, товарные знаки, полезные модели) по различным параметрам, включая текст, изображения и прочее. Кроме пользовательского интерфейса, платформа предлагает документированное API, возвращающее структурированные данные, что облегчает ин-

теграцию в аналитические задачи и разработку специализированных приложений.

eLIBRARY.RU – одна из крупнейших российских научных электронных библиотек с более чем 50 млн публикаций (статьи, монографии, диссертации, материалы конференций) и интеграцией с РИНЦ. Она обеспечивает доступ к ресурсам для проведения исследований, анализа публикационной активности и мониторинга научной деятельности.

Анализ международного опыта в сфере создания и развития инфраструктуры научных датахабов, а также рассмотрение существующих российских систем работы с научно-технической информацией позволяют выявить ряд проблем.

Во-первых, неоднородность форматов и стандартов метаданных затрудняет интеграцию и обработку крупных и разнородных массивов данных, а недостаток специализированной инфраструктуры не обеспечивает на должном уровне хранение и анализ больших объемов информации.

Во-вторых, отсутствие единого подхода к интероперабельности и стандартам взаимодействия между различными платформами порождает сложности доступа к данным и их повторного использования.

В-третьих, проблемы воспроизводимости научных результатов, а также неоднородная стандартизация правил по формированию метаданных отрицательно сказываются на возможности воспроизводимости исследований.

Кроме того, барьеры доступа к данным, связанные с авторским правом и правовыми ограничениями, этические аспекты использования данных и необходимость защиты конфиденциальности и интеллектуальной собственности также требуют учета при формировании эффективной экосистемы.

Важно отметить, что российские информационные системы в большей степени ориентированы на учет результатов научной деятельности, публикационную активность и патентную статистику, тогда как готовых решений, специализирующихся именно на совместном использовании структурированных научных датасетов, на сегодняшний день существенно меньше. Международная практика показывает, что успешное формирование платформ обмена научными данными подразумевает наличие целостной инфраструктуры, единой системы идентификаторов и аутентификации пользователей, а также строго прописанных регламентов обмена информацией.

### Общие принципы построения платформы обмена научными датасетами

Формирование единой междисциплинарной платформы для обмена научными датасетами в России предполагает комплексный подход, охватывающий несколько ключевых принципов (см. рисунок):

- создание и поддержка среды для обмена знаниями, совместных междисциплинарных исследований и обучения;
- разработка четкой методологии, включающей регламенты описания и оценки качества датасетов, а также механизмов защиты интеллектуальных прав;
- выбор и адаптация инструментов, позволяющих эффективно формировать, обрабатывать и распространять данные, включая решение задач по загрузке, выгрузке и визуализации.

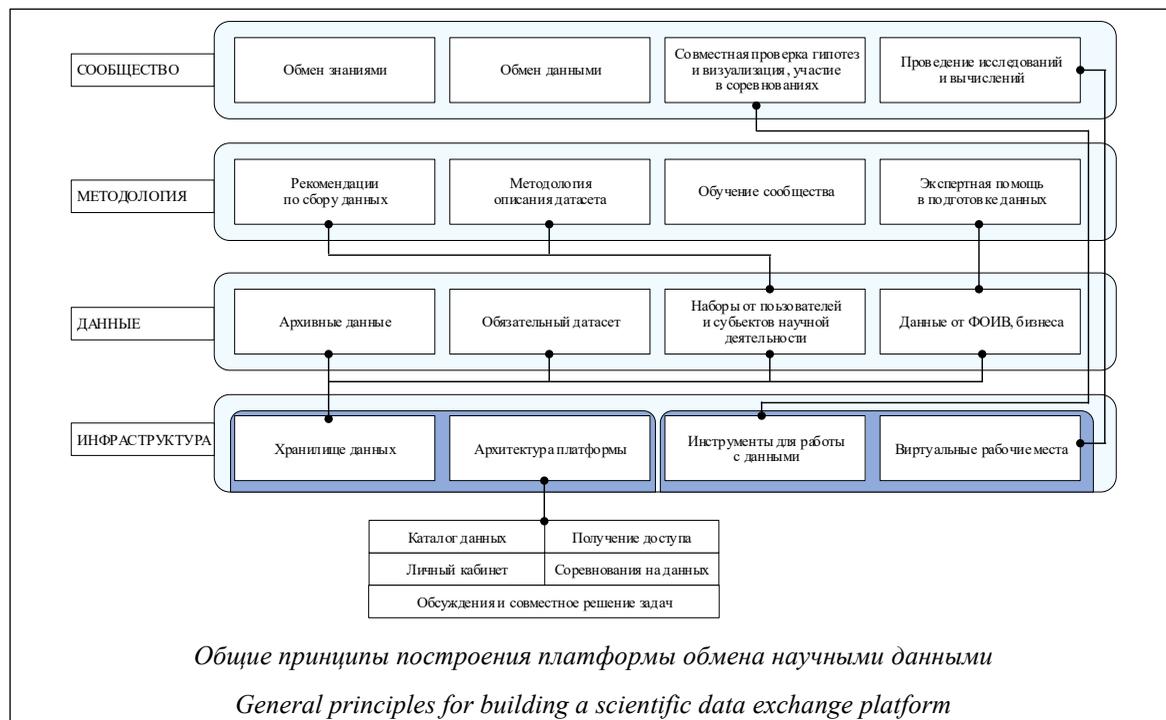
Еще одним определяющим фактором становится организация самих данных – создание единого каталога с детальным классификатором и реализация FAIR-принципов доступности и совместимости. Устойчивой основой для перечисленных элементов должна выступать соответствующая инфраструктура, отвечающая требованиям быстродействия, безопасности и интероперабельности.

Сообщество – это не только носитель культурных и научных традиций, но и движущая сила для непрерывного развития платформы

обмена датасетами. Международные проекты показывают важность комплексной поддержки взаимодействия между учеными. Так, в рамках EOSC созданы отдельные тематические рабочие группы, занимающиеся формированием рекомендаций и обменом опытом в области обработки, документирования и распространения данных. Аналогично платформа DataONE развивает программу обучения работе с научными данными, выпуская подробные руководства и проводя сетевые семинары, что способствует повышению цифровой грамотности и стимулирует сотрудничество участников. В экосистеме Kaggle (<https://www.kaggle.com/>) особую роль играют регулярные соревнования и форумы, где исследователи и инженеры обмениваются лучшими методами анализа и обработки датасетов. В результате создается культура кооперации, ускоряющая внедрение новых решений.

Следуя данному принципу, функционально необходимо реализовать:

- тематические рабочие группы (виртуальные исследовательские сообщества); создание специализированных секторов внутри платформы, где ученые по смежным дисциплинам могут общаться, совместно формировать библиотеки рабочих инструментов, обсуждать кейсы и получать экспертизу от более опытных коллег;
- форумы и систему «вопрос–ответ»; аналог Kaggle Discussions, где участники задают практические вопросы по обработке датасетов, об-



Общие принципы построения платформы обмена научными данными

General principles for building a scientific data exchange platform

ращаются за технической или методической поддержкой и получают обратную связь от членов сообщества с разным уровнем компетенций;

– регулярные конкурсы и хакатоны; организация соревнований по анализу предоставленных датасетов в различных предметных областях;

– инструменты для совместной работы; предоставление возможности вести коллективную разработку внутри платформы (совместное редактирование метаданных, настройка рабочих процессов для обработки данных и т.п.) с сохранением версии и истории изменений.

Преимуществом такой модели станет формирование сплоченного научного сообщества, где обмен датасетами сочетается с постоянной профессиональной поддержкой и обучением. В результате появится платформа, в которой исследователи не просто загружают и скачивают данные, а ведут активный диалог, совместно прорабатывают методологии анализа и решают прикладные научные проблемы.

Наличие методологии является фундаментальным принципом, определяющим структуру, стандарты и регламенты работы с данными в рамках платформы обмена научными датасетами. Международный опыт показывает, что выстраивание четких методологических основ позволяет обеспечить прозрачность, воспроизводимость и высокое качество научных исследований. Подобного рода подходы охватывают вопросы корректной структуры данных, единых правил описания и цитирования, а также правовой и этической сторон работы с научными результатами.

При разработке методологической основы необходимо учесть следующие основные аспекты.

• Регламенты описания датасетов (управление метаданными), фиксирующие:

– разработку единых руководств по созданию и ведению метаданных с учетом принятых в мире стандартов (Dublin Core, ISO 19115 для геоданных, Darwin Core для биологических данных и т.д.);

– интеграцию принципов FAIR в качестве ориентира для формирования структуры метаданных и описания данных;

– многоуровневое описание: от базовых полей (автор, название, ключевые слова) до расширенных (условия сбора, протокол экспериментов, единицы измерения), что повышает воспроизводимость научных исследований.

• Стандарты и процедуры оценки качества датасетов, включающие:

– установку критериев проверки достоверности, целостности и репрезентативности данных; например, Dryad и Zenodo внедряют базовый кураторский надзор для проверки формата, точности описания и соответствия лицензионным требованиям;

– разработку формализованных методик для оценки качества метаданных и самих наборов данных (ошибки пропущенных значений, корректность классификаций), а также определение ответственных за подобные проверки (кураторы, эксперты);

– использование показателей для мониторинга качества и востребованности датасетов: число загрузок, цитирований, перепубликаций, что помогает формировать репутацию и рейтинг отдельных наборов данных.

• Порядок обмена научными данными и защита авторских прав, предусматривающие:

– создание типовых лицензионных соглашений для регулирования использования данных и защиты авторских прав;

– описание процедур ознакомления и согласия всех участников научной деятельности с установленными правилами лицензирования, этическими нормами, а также механизмов контроля за соблюдением прав собственности;

– меры, направленные на соблюдение законов в сфере персональных данных и конфиденциальной информации.

• Юридические и этические регламенты, обеспечивающие:

– учет законодательства в области интеллектуальной собственности, авторских прав и защиты персональных данных;

– включение в методологические документы положений об этических аспектах: согласование протоколов исследования, анонимизация данных, обеспечение прозрачности исследования для участников экспериментов;

– регулярное обновление методических материалов с учетом изменения законодательства и появления новых международных и национальных актов.

Таким образом, методологический принцип формирует основу, на которой строится вся деятельность платформы от создания датасетов и их аннотирования до правового регулирования и обеспечения воспроизводимости результатов. Адаптируя проверенные мировые практики к отечественным условиям, можно разработать единый свод регламентов и процедур, отвечающих потребностям российских науч-

ных организаций. Эти методологические решения позволят наладить прозрачный и эффективный обмен научными данными.

Данные занимают центральное место в любой платформе обмена научными датасетами и определяют качество проводимых исследований и возможности для междисциплинарного сотрудничества. Международные проекты показывают, что единая точка доступа к разнородным наборам данных значительно упрощает их поиск, интеграцию и повторное использование. Особое внимание уделяется созданию подробных каталогов, каждый датасет сопровождается подробными метаданными, что позволяет отслеживать происхождение, качество и условия применения научного материала. Подход в формате единого окна позволит исследователям находить соответствующие наборы данных из различных направлений и оперативно выявлять возможности для междисциплинарных проектов.

Для эффективной работы каталога необходим определенный порядок наполнения платформы.

- Зафиксировать требования к загружаемым датасетам:
  - определить минимальный набор метаданных, соответствующий как международным стандартам, так и требованиям отечественных исследовательских сообществ;
  - назначить уникальный идентификатор (например DOI) для каждого набора данных, что упростит цитирование и отслеживание переиспользования;
  - обеспечить автоматизированную проверку загружаемого материала на корректность форматов (CSV, NetCDF, XML и т. д.) и целостность файлов, а также валидацию ключевых полей метаданных.
- Определить этапы и процедуры добавления новых наборов данных:
  - первичная загрузка: пользователь (исследователь или организация) регистрирует датасет, указывая основные характеристики, и прикрепляет файлы; на этом этапе может быть применена автоматизированная проверка структуры данных и соответствия базовым правилам метаданных;
  - кураторская проверка: специально назначенные эксперты (или кураторы) оценивают релевантность, правильность описания и соответствие установленным стандартам, включая возможные этические и правовые аспекты, в случае выявления несоответствий предоставляются рекомендации по доработке;

– публикация в каталоге: после успешного прохождения проверки датасет становится доступным для поиска и скачивания; ему присваивается уникальный постоянный идентификатор, формируется страница описания, при необходимости указываются условия лицензирования.

- Обеспечить долгосрочное хранение и управление версиями датасетов за счет:

– механизмов сохранения версий, отражающих изменения набора данных (добавление новых наблюдений, корректировки или уточнения), а также архивации старых версий в целях воспроизводимости экспериментов;

– средств интеграции с распределенными системами хранения (облачные репозитории, государственные ЦОД) для обеспечения устойчивости и резервирования данных.

- Обеспечить контроль качества и актуализацию информации:

– введение рейтинговой системы для оценки датасетов сообществом (по уровню полноты, надежности, востребованности);

– систематические обзоры качества с целью проверки востребованности и актуальности датасетов, а также определения устаревших или дублирующихся наборов, которые могут быть упразднены или объединены с другими.

- Принимать во внимание специфику отечественной науки и междисциплинарную ориентированность, для чего требуется:

– учитывать потребности различных российских научных организаций, а также специфику предметных областей;

– предусмотреть возможность интеграции с уже существующими базами данных и реестрами для упрощения импортирования и синхронизации научных данных;

– создать механизмы ссылки и совместного отображения метаданных (cross-linking), чтобы исследователи могли проследивать связь между статьями, патентами и первичными данными.

Инфраструктура является ключевым фактором, определяющим устойчивость, производительность и безопасность работы платформы обмена научными датасетами. Международные проекты наглядно демонстрируют важность комплексного подхода к организации вычислительных ресурсов, сетевых взаимодействий и механизмов хранения, без которых невозможно обеспечить надлежащий уровень обслуживания. Например, в EUDAT применяется распределенная архитектура, основанная на координирующих узлах и узлах-участниках,

что гарантирует репликацию и надежное хранение данных в нескольких центрах обработки [13]. Подобная модель дает высокую устойчивость к сбоям, повышает пропускную способность и сокращает риски единой точки отказа.

Целесообразно учитывать уже имеющиеся национальные центры обработки данных и мощные исследовательские сети, в частности, проект НИКС (Национальная исследовательская компьютерная сеть) как возможную основу для построения единой платформы.

Приведем основные аспекты требований к инфраструктуре.

- Репликация и распределенное хранение:

- размещение основных узлов (центров обработки данных) в нескольких регионах с учетом географического фактора и доступности высокоскоростных сетей; это обеспечивает минимизацию задержек и резервирование в случае отключений или аварий;

- использование модулей репликации и кластерного хранения, позволяющих гарантировать целостность и восстановление данных при сбоях оборудования.

- Высокопроизводительные вычислительные мощности:

- интеграция с отечественными суперкомпьютерными центрами и кластерами, что позволит проводить сложные расчеты и анализ больших объемов данных непосредственно в рамках платформы; опыт EOSC показывает эффективность такого подхода, когда пользователи могут занимать вычислительные ресурсы из федеративной экосистемы на основе сервисов B2STAGE или аналогичных решений;

- возможность быстрого масштабирования вычислительных и сетевых ресурсов под пиковые нагрузки, например, во время проведения хакатонов или крупных научных проектов.

- Интероперабельность и открытые стандарты:

- использование принципов совместимости на уровне протоколов и интерфейсов, что упрощает подключение к международным сервисам и репозиториям;

- реализация федеративной аутентификации и авторизации пользователей, повышающая удобство использования учетных записей и сокращающая необходимость многократной регистрации.

- Безопасность и защита данных:

- введение многоуровневой системы контроля доступа, позволяющей разграничивать права пользователей в соответствии с их ролью (исследователь, куратор, администратор);

- шифрование данных при передаче и хранении с учетом требований российского законодательства и международных норм по защите интеллектуальной собственности;

- регулярные аудиты и тесты на проникновение для мониторинга надежности инфраструктуры.

- Высокая доступность и скорость отклика:

- организация резервных каналов связи между основными хранилищами и вычислительными центрами; при сбое в одном сегменте трафик должен автоматически переключаться на другой;

- применение механизмов кеширования для ускорения доступа к часто запрашиваемым датасетам.

- Системная интеграция и сервисный уровень (SLA):

- предоставление четкого уровня сервиса, в котором фиксируются показатели доступности и сроков реакции на инциденты;

- наличие круглосуточной службы технической поддержки и мониторинга для оперативного решения проблем с доступом, передачей или обработкой данных;

- интеграция с уже существующими отечественными платформами учета научных результатов и внешними репозиториями через стандартные API, обеспечивающая единый информационный контур.

## Заключение

Таким образом, при проектировании инфраструктуры платформы стоит принимать во внимание принцип устойчивого масштабируемого и безопасного распределенного хранения данных, а также предусмотреть инструменты для межплатформенной интеграции и высокопроизводительной обработки. Опыт международных инициатив показывает, что без комплексного решения вопросов распределенного хранения, сетевой связности, федеративного управления доступом и резервирования ресурсов невозможно создать эффективный сервис для обмена научными датасетами.

Формирование единой платформы обмена научными датасетами требует целостного продуманного подхода, основанного на консолидации международного опыта, адаптации мировых стандартов и учете специфики отечественной научной инфраструктуры. Успешные зарубежные примеры демонстрируют важность единой среды взаимодействия специалистов,

детальной методологической базы, современных инструментов управления данными и соответствующей инфраструктурной поддержки. В российском контексте особую роль при этом играет объединение усилий различных исследовательских групп, вузов, отраслевых институтов и технологических партнеров, что позволит стимулировать междисциплинарные исследования и повысить качество научных результатов.

На уровне организации и методологии необходимо обеспечить единые регламенты описания, хранения и документирования данных, а также формализовать порядок обмена ими в соответствии с принципами FAIR. Внедрение систем идентификации и единых протоколов аутентификации дает возможность поддерживать прозрачность и воспроизводимость исследований, а также защищать авторские права и интеллектуальную собственность. Рациональная структура данных, основанная на классификаторах предметных областей и междисциплинарных рубрикаторах, позволит создавать хорошо организованный научный датахаб, доступный широкому кругу пользователей.

При проектировании инфраструктуры платформы следует ориентироваться на устойчивое масштабируемое и безопасное распределенное хранение данных, предусмотрев механизмы

межплатформенной интеграции и высокопроизводительной обработки больших объемов информации. Международный опыт наглядно показывает, что без комплексного решения вопросов сетевой связности, федеративного управления доступом и резервирования ресурсов невозможно создать эффективный сервис для обмена научными датасетами. Адаптируя эти подходы к отечественным условиям и опираясь на уже существующие сети, например НИКС, российская платформа сможет обеспечить непрерывный и надежный доступ к данным, а также их оперативную обработку, соответствующую требованиям современных научных исследований.

Таким образом, комплексное сочетание функциональных сервисов, регламентов взаимодействия и механизмов поддержки сообщества позволит сформировать полноценную экосистему открытой науки, способствующую повышению конкурентоспособности российской науки в глобальном пространстве. Внедрение описанных решений создаст условия для более тесного междисциплинарного сотрудничества, простимулирует инновационную активность и обеспечит доступ исследовательских групп к данным, необходимым для проведения передовых исследований и технологических разработок.

### Список литературы

1. European Commission: DG Research and Innovation. Open innovation, open science, open to the world: A vision for Europe. Publications Office of the European Union, 2015. URL: <https://data.europa.eu/doi/10.2777/061652> (дата обращения: 12.06.2024).
2. Making open science a reality. OECD Science, Technology and Industry Policy Papers, 2015, no. 25. URL: [https://www.oecd.org/en/publications/making-open-science-a-reality\\_5jrs2f963zsl-en.html](https://www.oecd.org/en/publications/making-open-science-a-reality_5jrs2f963zsl-en.html) (дата обращения: 12.06.2024). doi: 10.1787/5jrs2f963zsl-en.
3. Wilkinson M., Dumontier M., Aalbersberg I. et al. The FAIR Guiding Principles for scientific data management and stewardship. *Scientific Data*, 2016, vol. 3, art. 160018. doi: 10.1038/sdata.2016.18.
4. Tenopir C., Rice N.M., Allard S., Baird L., Borycz J. et al. Data sharing, management, use, and reuse: Practices and perceptions of scientists worldwide. *PLoS One*, 2020, vol. 15, no. 3, art. e0229003. doi: 10.1371/journal.pone.0229003.
5. Pribec I., Hachinger S., Hayek M., Pringle G.J. et al. Efficient and reliable data management for biomedical applications. In: *MIMB*, 2024, vol. 2716, pp. 383-403. doi: 10.1007/978-1-0716-3449-3\_18.
6. Bhattacharjee P., Bird I., Calore F., Doglioni C., Eckner Ch. et al. ESCAPE dark matter science project for EOSC future project (WP6.3). *ARPHA Preprints*, 2023. doi: 10.3897/arphapreprints.e116673.
7. Scardaci D.O., Sciacca E., Hériché J.-K., Van De Sanden M., Klaas W. et al. A landscape overview of the EOSC interoperability framework – capabilities and gaps. *Zenodo*, 2023. URL: <https://zenodo.org/records/8399710> (дата обращения: 12.06.2024).
8. Rubio-Montero A.J., Pagán-Muñoz R., Mayo-García R., Pardo-Díaz A., Sidelnik I., Asorey H. The EOSC-Synergy cloud services implementation for the Latin American Giant Observatory (LAGO). *Proc. ICRC2021*, 2021, vol. 395. doi: 10.22323/1.395.0261.
9. Michener W.K., Allard S., Budden A., Cook R.B., Douglass K. Participatory design of DataONE—Enabling cyberinfrastructure for the biological and environmental sciences. *Ecological Informatics*, 2012, vol. 11, pp. 5–15. doi: 10.1016/j.ecoinf.2011.08.007.
10. Isard M., Budiu M., Yu Y., Birrell A., Fetterly D. Dryad: Distributed data-parallel programs from sequential building blocks. *ACM SIGOPS Operating Systems Review*, 2007, vol. 41, no. 3, pp. 59–72. doi: 10.1145/1272998.1273005.
11. Stal C., Covataru C., Müller J., Parnic V. et al. Supporting long-term archaeological research in southern romania chalcolithic sites using multi-platform UAV mapping. *Drones*, 2022, vol. 6, no. 10, art. 277. doi: 10.3390/drones6100277.

12. Лупян Е.А., Прошин А.А., Бурцев М.А., Кашницкий А.В., Балашов И.В. и др. Опыт эксплуатации и развития центра коллективного пользования системами архивации, обработки и анализа спутниковых данных (ЦКП «ИКИ-Мониторинг») // *Современные проблемы дистанционного зондирования Земли из космоса*. 2019. Т. 16. № 3. С. 151–170. doi: 10.21046/2070-7401-2019-16-3-151-170.

13. Lecarpentier D., Wittenburg P., Elbers W., Michelini A., Kanso R., Coveney P., Baxter R. EUDAT: A new cross-disciplinary data infrastructure for science. *IJDC*, 2013, vol. 8, no. 1, pp. 279–287. doi: 10.2218/ijdc.v8i1.260.

Software &amp; Systems

doi: 10.15827/0236-235X.149.122-133

2025, 38(1), pp. 122–133

### Design principles of a domestic platform for scientific dataset exchange

Kirill V. Garev<sup>1,2</sup>✉<sup>1</sup> Information Technology Department of the Bank of Russia, Moscow, 107016, Russian Federation<sup>2</sup> National Research Centre “Kurchatov Institute”, Moscow, 117218, Russian Federation

#### For citation

Garev, K.V. (2025) ‘Design principles of a domestic platform for scientific dataset exchange’, *Software & Systems*, 38(1), pp. 122–133 (in Russ.). doi: 10.15827/0236-235X.149.122-133

#### Article info

Received: 01.07.2024

After revision: 26.07.2024

Accepted: 31.07.2024

**Abstract.** The paper considers prerequisites and proposals for establishing a domestic platform for a scientific dataset exchange in terms of global trends related to the development of open science and spread of FAIR-principles. The authors analyze foreign initiatives (EUDAT, EOSC, DataONE, Dryad, Zenodo) to identify key problems that hinder effective use, preservation and reuse of scientific data. These problems include the lack of uniform regulations for dataset description, disparate infrastructure solutions, insufficient cross-platform interoperability, and difficulty in ensuring reproducibility of research. The authors focus on the role of the professional community, the importance of creating an environment for sharing experience, conducting interdisciplinary projects, and improving skills in working with large datasets. The paper emphasizes the need to systematize the work with scientific data and to unify the requirements for their collection, storage, processing and presentation. This will increase the transparency of research processes. The paper substantiates the feasibility of implementing distributed storage mechanisms, federated authentication and high-performance computing resources capable of meeting the needs of the domestic scientific community. Finally, it outlines proposals for designing a unified platform for scientific dataset exchange: from developing methodological regulations and standards for interaction with external systems to principles of integrating analytical tools and ensuring reliable data protection.

**Keywords:** data hubs, scientific data, scientific datasets, digital platforms for scientific data exchange, open science, FAIR, EUDAT, EOSC, DataONE, Dryad, Zenodo

#### References

1. European Commission: DG Research and Innovation (2015) ‘Open innovation, open science, open to the world: A vision for Europe’, *Publications Office of the European Union*, available at: <https://data.europa.eu/doi/10.2777/061652> (accessed June 12, 2024).

2. (2015) ‘Making open science a reality’, *OECD Science, Technology and Industry Policy Papers*, (25), available at: [https://www.oecd.org/en/publications/making-open-science-a-reality\\_5jrs2f963zs1-en.html](https://www.oecd.org/en/publications/making-open-science-a-reality_5jrs2f963zs1-en.html) (accessed June 12, 2024). doi: 10.1787/5jrs2f963zs1-en.

3. Wilkinson, M., Dumontier, M., Aalbersberg, I. et al. (2016) ‘The FAIR Guiding Principles for scientific data management and stewardship’, *Scientific Data*, 3, art. 160018. doi: 10.1038/sdata.2016.18.

4. Tenopir, C., Rice, N.M., Allard, S., Baird, L., Borycz, J. et al. (2020) ‘Data sharing, management, use, and reuse: Practices and perceptions of scientists worldwide’, *PLoS One*, 15(3), art. e0229003. doi: 10.1371/journal.pone.0229003.

5. Pribec, I., Hachinger, S., Hayek, M., Pringle, G.J. et al. (2024) ‘Efficient and reliable data management for biomedical applications’, in *MIMB*, 2716, pp. 383–403. doi: 10.1007/978-1-0716-3449-3\_18.

6. Bhattacharjee, P., Bird, I., Calore, F., Doglioni, C., Eckner, Ch. et al. (2023) ‘ESCAPE dark matter science project for EOSC future project (WP6.3)’, *ARPHA Preprints*. doi: 10.3897/arphapreprints.e116673.

7. Scardaci, D.O., Sciacca, E., Hériché, J.-K., Van De Sanden, M., Klaas, W. et al. (2023) ‘A landscape overview of the EOSC interoperability framework – capabilities and gaps’, *Zenodo*, available at: <https://zenodo.org/records/8399710> (accessed June 12, 2024).

8. Rubio-Montero, A.J., Pagán-Muñoz, R., Mayo-García, R., Pardo-Díaz, A., Sidelnik, I., Asorey, H. (2021) ‘The EOSC-Synergy cloud services implementation for the Latin American Giant Observatory (LAGO)’, *Proc. ICRC2021*, 395. doi: 10.22323/1.395.0261.
9. Michener, W.K., Allard, S., Budden, A., Cook, R.B., Douglass, K. (2012) ‘Participatory design of DataONE—Enabling cyberinfrastructure for the biological and environmental sciences’, *Ecological Informatics*, 11, pp. 5–15. doi: 10.1016/j.ecoinf.2011.08.007.
10. Isard, M., Budiu, M., Yu, Y., Birrell, A., Fetterly, D. (2007) ‘Dryad: Distributed data-parallel programs from sequential building blocks’, *ACM SIGOPS Operating Systems Review*, 41(3), pp. 59–72. doi: 10.1145/1272998.1273005.
11. Stal, C., Covataru, C., Müller, J., Parnic, V. et al. (2022) ‘Supporting long-term archaeological research in southern Romania Chalcolithic sites using multi-platform UAV mapping’, *Drones*, 6(10), art. 277. doi: 10.3390/drones6100277.
12. Loupian, E.A., Proshin, A.A., Bourtsev, M.A., Kashnitskii, A.V., Balashov, I.V. et al. (2019). ‘Experience of development and operation of the IKI-Monitoring center for collective use of systems for archiving, processing and analyzing satellite data’, *Current Problems in Remote Sensing of the Earth from Space*, 16(3), pp. 151–170 (in Russ.). doi: 10.21046/2070-7401-2019-16-3-151-170.
13. Lecarpentier, D., Wittenburg, P., Elbers, W., Michelini, A., Kanso, R., Coveney, P., Baxter, R. (2013) ‘EUDAT: A new cross-disciplinary data infrastructure for science’, *IJDC*, 8(1), pp. 279–287. doi: 10.2218/ijdc.v8i1.260.

**Авторы**

**Гарев Кирилл Валерьевич**<sup>1,2</sup>,  
руководитель направления,  
аспирант, kv@garev.ru

<sup>1</sup> Департамент информационных технологий  
Банка России, г. Москва, 107016, Россия

<sup>2</sup> Национальный исследовательский центр  
«Курчатовский институт» – НИИСИ,  
г. Москва, 117218, Россия

**Authors**

**Kirill V. Garev**<sup>1,2</sup>, Team Lead,  
Postgraduate Student,  
kv@garev.ru

<sup>1</sup> Information Technology Department of the Bank  
of Russia, Moscow, 107016, Russian Federation

<sup>2</sup> National Research Centre  
“Kurchatov Institute”,  
Moscow, 117218, Russian Federation

## Применение специализированных программных комплексов для автоматизации инженерных расчетов оборудования

Э.В. Осипов <sup>1</sup>✉, А.А. Хоменко <sup>1</sup>, Л.Э. Осипова <sup>2</sup>

<sup>1</sup> Казанский национальный исследовательский технологический университет, г. Казань, 420015, Россия

<sup>2</sup> Казанский государственный архитектурно-строительный университет, г. Казань, 420043, Россия

### Ссылка для цитирования

Осипов Э.В., Хоменко А.А., Осипова Л.Э. Применение специализированных программных комплексов для автоматизации инженерных расчетов оборудования // Программные продукты и системы. 2025. Т. 38. № 1. С. 134–142. doi: 10.15827/0236-235X.149.134-142

### Информация о статье

Группа специальностей ВАК: 1.2.2, 2.3.7

Поступила в редакцию: 24.04.2024

После доработки: 26.08.2024

Принята к публикации: 27.08.2024

**Аннотация.** В данной статье описана разработка подхода, при котором инженерные методики расчета оборудования оформляются в виде отдельных программ, связанных с моделирующими САПР, а результаты являются исходными данными для САД-системы. Объект исследования – отстаивание толуола от водометанольной смеси, а его предмет – нестандартное технологическое оборудование процесса отстаивания (отстойник). Технологические расчеты для составления материального и энергетического балансов проведены в среде универсальной моделирующей программы, геометрические размеры нестандартного оборудования определены при помощи разработанной программы, а трехмерная модель оборудования сформирована в графической САД-системе. Указанный подход реализован в виде программы на языке VBA, которая связана с моделью процесса отстаивания. Выбор VBA объясняется тем, что он является базовым для программы и используется при разработке программ для автоматизации вычислений в указанном комплексе. Поскольку в БД программы отсутствует модуль расчета отстойника, данный модуль разработан при использовании инструмента Import User Model. Надстройкой Case Study проведен параметрический анализ и определено предельное содержание метанола в исходной смеси (до 0.3 массовой доли), при котором плотность водометанольной смеси больше плотности толуольной фазы. Расчет основных размеров оборудования проводился по инженерной методике, которая оформлена в виде отдельной программы, связанной с моделью в моделирующей САПР. Трехмерная модель корпуса аппарата разработана в графической САПР при использовании переменных и является основой сборки отстойника, дополненной стандартными элементами, выбранными из БД программы. В результате разработана программа по определению размеров отстойника, связывающая моделирующую и графическую САПР, что позволяет автоматизировать инженерные методики расчета энергоэффективного оборудования.

**Ключевые слова:** программные комплексы, VBA, автоматизация проектирования, компьютерное моделирование, моделирующие САПР, инженерные расчеты

**Введение.** Подбор, определение геометрических размеров технологического оборудования и разработка конструкторской документации являются важными задачами проектирования, без решения которых затруднительно создавать энергоэффективные химические технологии. При этом изделие должно быть подобрано и рассчитано таким образом, чтобы при низких материальных затратах и максимальной нагрузке обеспечивалось минимально возможное энергопотребление. Для этого используются различные инженерные методики, успешное применение которых зависит от точности расчета *теплофизических свойств* (ТФС) входных и выходных потоков.

Анализ литературных источников показывает, что автоматизация инженерных методик в основном связана с написанием программ, в которые включаются расчетные формулы [1–3]. При этом отмечено, что первым этапом проек-

тирования изделия или технологического оборудования является первичное проектирование [4], результаты которого приблизительные, а эффективность их использования зависит от интуиции и опыта проектировщика. Как правило, при проектировании *химико-технологических процессов* (ХТП) с применением моделирующих САПР составляют материальный и энергетический балансы, на основе которых производится подбор оборудования или его расчет (при использовании нестандартного изделия).

Сегодня на рынке немало программных решений для моделирования технологического процесса и расчета оборудования, например, Aspen HYSYS, ChemCad и Unisim Design. Их БД включают утилиты для определения геометрических размеров колонн, емкостей, сепараторов и теплообменников [5, 6]. Отечественным аналогом указанных программ является

AEROSYM, который показывает результаты расчета, близкие к выполненным в Aspen HYSYS [7].

Подготовка конструкторской документации (КД) – одна из важных и наиболее трудоемких задач в САПР. Ее реализуют посредством графических программных средств по автоматизации разработки и выполнения конструкторской документации. Примерами таких программ являются Autocad, КОМПАС-3D, nanoCADPro и т.д.

Моделировать физические процессы и решать различные инженерные задачи позволяют САЕ-системы [8]. К ним можно отнести, например, ANSYS и FlowVision.

### Обзор существующих подходов

Программное обеспечение фирмы «Aspen» активно применяют для построения цифровых двойников на базе нейросети AI Model Builder, используя пользовательский код MS Excel-VBA [9]. Возможности программы Aspen HYSYS позволяют также разрабатывать пользовательский модуль внутри программы [10]. Несмотря на то, что программы Aspen HYSYS и Aspen Plus предназначены для моделирования процессов химической технологии, в некоторых исследованиях их применяли для расчета теплотехнических процессов. Например, в [11] предложено использовать Aspen Plus для моделирования конденсационной установки, а в [12] Aspen HYSYS применяется для расчета парогазовой установки нефтедобывающей компании.

Как правило, функционал САД-систем включает в себя дополнительные возможности, которые можно реализовать при помощи пользовательских функций. Так, в работе [8] рассмотрен механизм импорта твердотельной модели из КОМПАС-3D в ANSYS при моделировании напряженно-деформационной задачи несущего корпуса. В работе [13] приведены подходы для анализа проектных решений, выполненных в САПР КОМПАС-3D. При составлении математической модели вакуумной сублимационной сушки [14] в CFD-модель добавлено описание кинетики, сформированное в виде отдельной программы.

Применение специализированного ПО планирования в совокупности с инструментами моделирования производства позволяет получать существенный экономический эффект [15]. Возможно интегрированное решение, которое базируется на трех передовых программах: Aspen HYSYS, Bentley AXSYS и Bentley PlantWise.

Расчет технологических схем проведен в Aspen HYSYS, а концептуальную 3D-модель создает решение Bentley PlantWise [16].

Указанные работы отражают лишь небольшую часть применений САД- и САЕ-систем при выполнении инженерных задач. Однако для решения задачи в САЕ-системе необходимо подготовить модель, в которой указаны геометрические параметры изделия, а САД-системы, как правило, не предназначены для инженерных расчетов технологического оборудования. САПР для моделирования ХТП (Aspen HYSYS, ChemCad, Unisim Design и т.д.) позволяют рассчитать материальный и энергетический балансы процесса, но возможности определения геометрических размеров оборудования (особенно нестандартного) существенно ограничены. При этом такой расчет возможен при использовании инженерных методик, для которых необходимо определить ТФС входных и выходных потоков. Таким образом, возникает задача разработки программы автоматизированного расчета размеров оборудования по инженерным методикам, однако включение в нее блока по расчету ТФС существенно увеличивает трудоемкость. В то же время расчет необходимых ТФС реализован в САПР для моделирования ХТП.

### Предлагаемый подход

Цель данного исследования – разработка ПО по расчету размеров оборудования (на примере отстойника смеси «толуол+вода и метанол»), в котором ТФС рассчитаны в программе для моделирования ХТП, а твердотельная модель сформирована в САД-системе.

Для автоматизации инженерных методик предлагается разработать такое ПО (далее – программа), которое рассчитывает размеры оборудования, но при этом данные по ТФС импортирует из внешней моделирующей программы. Разрабатываемая программа необходима для расчета размеров оборудования, подготовки исходных данных для твердотельной модели и сохранения полученных значений для импорта в отчет. Схема обмена данными между программой, моделирующей САПР и САД-системой показана на рисунке 1.

В универсальной моделирующей программе (УМП) создается модель процесса и рассчитываются ТФС входного, выходных и внутренних потоков. Далее требуемые свойства целевых потоков импортируются в программу для определения размеров оборудования. Рассчитан-

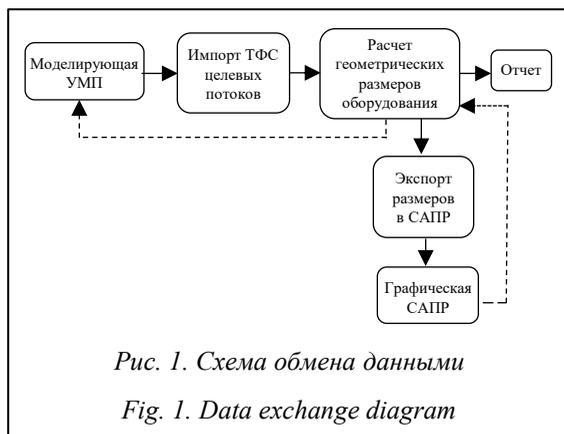


Рис. 1. Схема обмена данными

Fig. 1. Data exchange diagram

ные размеры экспортируются в САД-систему, где формируется твердотельная модель, являющаяся основой для подготовки комплекта КД.

В программе также формируются данные, необходимые для оформления отчета (ТФС потоков, таблицы с результатами расчетов и т.д.).

**Описание объекта исследования**

Принципиальная схема процесса приведена на рисунке 2. Разделение смеси толуола и метанольной воды производится в последовательно расположенных емкостях-отстойниках N, готовый продукт собирается в емкость E-1.

На рисунке 2 приняты следующие обозначения: N – отстойники; E – емкость толуола; 1, 2 и 3 – потоки смеси легкой и тяжелой фаз соответственно.

На первом этапе расчета определяется количество отстойников в схеме (общее количество обозначено N). Для этого необходимо рассчитать минимальную длину отстойника и время пребывания смеси в аппарате [17]. Эти параметры определяют по следующим формулам:

$$D_{\text{экв}} = \frac{4V_{\text{вх}}\rho_{\text{эм}}}{2300\pi\mu_{\text{эм}}}; \tag{1}$$

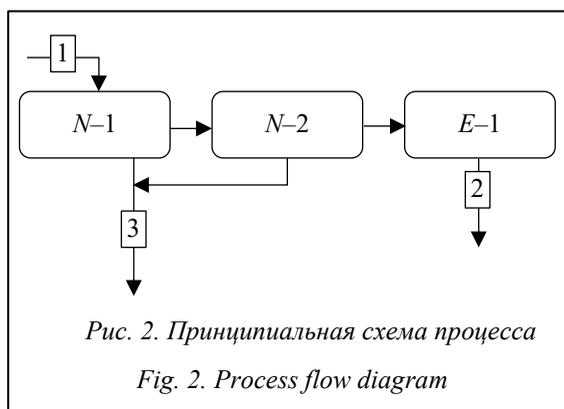


Рис. 2. Принципиальная схема процесса

Fig. 2. Process flow diagram

$$D_{\text{экв}} = \frac{4V_{\text{вх}}\rho_{\text{эм}}}{2300\pi\mu_{\text{эм}}}\sqrt{\frac{\pi}{f(\varepsilon)}}; \tag{2}$$

$$f(\varepsilon) = \frac{\pi}{2} + (1-\varepsilon)\sqrt{\varepsilon(2-\varepsilon)} + \text{asin}(1-\varepsilon); \tag{3}$$

$$w_0 = \frac{d^2g(\rho_{\text{в}} - \rho_{\text{н}})}{18\mu_{\text{н}}}; \tag{4}$$

$$w_{\text{ос}} = w_0(1-B)^{4.7}; \tag{5}$$

$$h_{\text{в}} = \varepsilon R; \tag{6}$$

$$h_{\text{ос}} = D - h_{\text{в}}; \tag{7}$$

$$D_{\text{экв-эм}} = \frac{D}{\sqrt{\frac{\pi}{f(\varepsilon)}}}; \tag{8}$$

$$w_{\text{ср}} = \frac{4(V_{\text{вх}} + V_{\text{вых}})}{4D_{\text{экв}}^2 2}; \tag{9}$$

$$L = \frac{h_{\text{ос}}w_{\text{ср}}}{w_{\text{ос}}}. \tag{10}$$

Используемые обозначения:  $V_{\text{вх}}$  – нагрузка отстойника по смеси, м<sup>3</sup>/с;  $V_{\text{вых}}$  – объемный расход тяжелой фазы, м<sup>3</sup>/с;  $\rho_{\text{эм}}$  – плотность смеси, кг/м<sup>3</sup>;  $\mu_{\text{эм}}$  – вязкость смеси, Па·с;  $g$  – ускорение свободного падения, м/с<sup>2</sup>;  $\rho_{\text{в}}$  – плотность водометанольной фазы, кг/м<sup>3</sup>;  $\rho_{\text{н}}$  – плотность толуольной фазы, кг/м<sup>3</sup>;  $\mu_{\text{н}}$  – вязкость толуольной фазы, Па·с;  $B$  – обводненность смеси;  $R$  – радиус отстойника, м.

Исходные данные, предоставленные заказчиком для расчета разделения толуола и метанольной воды: расход потока минимальный – 1 т/ч, максимальный – 7 т/ч; температура на входе – 40 °С; давление – 1.5 атм. Целевое качество толуола на выходе – 99.9 % масс.

**Реализация предлагаемого подхода**

В качестве УМП выбрано ПО фирмы «Aspen» – Aspen HYSYS V12, БД которого содержит все необходимые вещества, участвующие в процессе. Однако в программе отсутствует модуль по расчету отстойника, но есть возможность создать пользовательский модуль.

Если формировать модель отстойника набором стандартных модулей, то возникает сложность в специфицировании элементов. Согласно [18], написание внешних программ возможно несколькими способами. Например, можно использовать Excel-VBA, Python, Unity или пакет MATLAB. В случае использования MATLAB, Python и Unity возможна ошибка при введении мольных расходов входных потоков, поэтому целесообразно разработать мо-

дель на основе пользовательского модуля Import User Model.

Данный модуль спроектирован на VBA, его общий вид представлен на рисунке 3. Результаты расчета (данные по потокам) приведены в таблице 1.

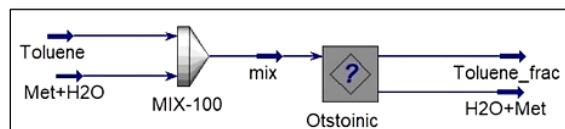


Рис. 3. Расчетная схема процесса, реализованная в Aspen HYSYS V12

Fig. 3. Process design model in Aspen HYSYS V12

На рисунке 4 представлена блок-схема реализации предлагаемого подхода.

Приведем фрагмент программного кода внутреннего модуля:

```
Dim CMFs, CMFs1, CMFs2, MF1 As Variant
Dim fM, ProdM, wtrM As Variant
Dim xtoll As Double
Dim x1, x2, x3 As Variant
Dim y1, y2, y3 As Variant
CMFs = feed.ComponentMassFlowValue
CMFs1 = prod.ComponentMassFlowValue
CMFs2 = wtr.ComponentMassFlowValue
x=prod.ComponentMassFractionValue
y=wtr.ComponentMassFractionValue
fM=feed.MassFlowValue
xtoll=0.999
ProdM=CMFs (tol) /xtoll
x(0)=0
x(1)=xtoll
x(2)=1-xtoll
prod.Pressure.Calculate(feed.Pres-
sureValue)
prod.Temperature.Calculate(feed.Tem-
peratureValue)
prod.MassFlow.Calculate(ProdM)
prod.ComponentMassFraction.Calcul-
ate(x)
wtrM=fM-ProdM
y(0)=CMFs (Met) /wtrM
y(1)=0
y(2)=1-y(0)
wtr.Pressure.Calculate(feed.Pres-
sureValue)
```

```
wtr.Temperature.Calculate(feed.Temper-
atureValue)
wtr.MassFlow.Calculate(wtrM)
wtr.ComponentMassFraction.Calculate(y)
ActiveObject.SolveComplete
```

В представленном коде переменные feed, prod и wtr соответствуют входному потоку легкой и тяжелой фаз соответственно. WaterPosn, tol и Met относятся к компонентам вода, толуол и метанол.

На первом этапе в Aspen HYSYS вводятся параметры потоков и на базе пользовательского модуля проводится расчет процесса отстаивания. Следующий этап – интеграция уравнений в расчетную среду. Данный шаг реализован при помощи кода VBA, написанного в среде Visual Studio 2022. Общий вид программы представлен на рисунке 5.

Фрагмент кода по импорту ТФС:

```
Public hyApp As HYSYS.Application
Public hyCase As HYSYS.SimulationCase
Public hymix As HYSYS.ProcessStream
Public hyToluene As
HYSYS.ProcessStream
Public hyHM As HYSYS.ProcessStream
hyApp = CreateObject("HYSYS.Appli-
cation")
hyCase=hyApp.ActiveDocument
hymix=hyCase.Flowsheet.Materi-
alStreams.Item("mix")
hyToluene=hyCase.Flowsheet.Materi-
alStreams.Item("Toluene_frac")
hyHM=hyCase.Flowsheet.Materi-
alStreams.Item("H2O+Met")
```

В программу импортируют данные по расходу, плотности и вязкости входного и выходных потоков, далее по уравнениям определяются требуемый внутренний диаметр и минимальная длина отстойника. Расчетные данные экспортируют в файл Excel.

В качестве САД-системы выбран программный продукт КОМПАС-3D V20, а рассчитанные значения вводят в твердотельную модель через таблицу переменных. Следует учесть, что размеры модели в КОМПАС-3D приняты в миллиметрах, а единица измерения длины

Таблица 1

Результаты расчета

Table 1

Calculation results

Параметр	Поток		
	mix	Toluene_frac	H2O+Met
Температура, °C	40	40	40
Давление, атм.	1.5	1.5	1.5
Массовый расход, кг/час	7 000	3 504	3 496
Плотность, кг/м <sup>3</sup>	863.8	850.5	873.7

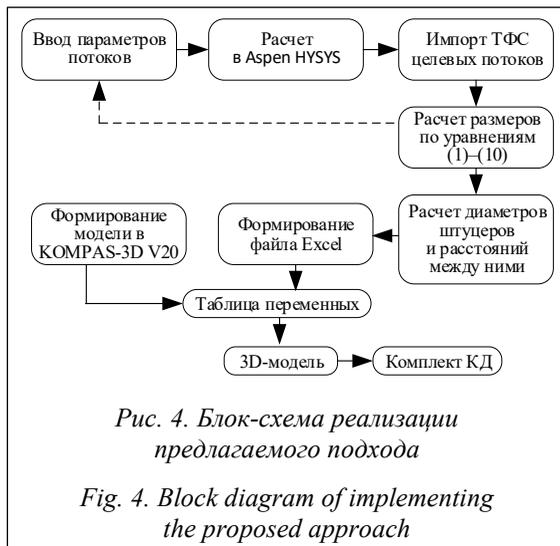


Рис. 4. Блок-схема реализации предлагаемого подхода

Fig. 4. Block diagram of implementing the proposed approach

и диаметра, рассчитанных программой, – метр. Поэтому необходимо предусмотреть перевод из метров в миллиметры.

Конструктивно отстойник представляет собой горизонтальный/вертикальный цилиндрический аппарат, закрепленный на опорах к фундаменту. В корпусе аппарата необходимо предусмотреть штуцера для ввода/вывода смеси, для предохранительного клапана, уровнера и сброса шлама. При этом в зависимости от требований заказчика можно рассмотреть другие конструктивные элементы. Корпус является основой отстойника, к нему присоединены остальные элементы (штуцера, крышки, опоры и т.д.).

Размеры, необходимые для формирования корпуса отстойника, целесообразно задать через таблицу переменных.

Таблица значений внешних переменных (таблица переменных) хранится в файле и со-

держит predetermined значения внешних переменных этого файла. Если в файле есть внешние переменные, то при создании в нем таблицы переменных можно использовать функцию чтения из внешнего источника. В результате автоматически создаются столбцы, соответствующие файлу внешних переменных. Возможно также чтение таблицы переменных из файла формата Excel.

Общий вид модели корпуса и таблицы переменных представлен на рисунке 6.

Выбор значений переменных из таблицы не является обязательным. При необходимости пользователь может задать внешние переменные вручную, однако это существенно увеличивает время создания 3D-модели оборудования. Если в файле отсутствует таблица переменных, то ввод значений переменных вручную – единственный способ их изменить.

Следует учитывать, что геометрические размеры связаны между собой, поэтому для сокращения времени проектирования целесообразно связать их параметрически. Кроме того, все параметры в таблице переменных являются внешними, то есть вводятся в модель через программу (специально разработанную для этих целей).

На основе данной модели определяют конструкцию и типоразмер стандартных элементов, при этом целесообразно использовать БД программы КОМПАС-3D V20.

### Результаты и обсуждение

Основной движущей силой процесса отстаивания является разность плотностей тяжелой (водо-метанольной) и легкой (толуольной) фаз.

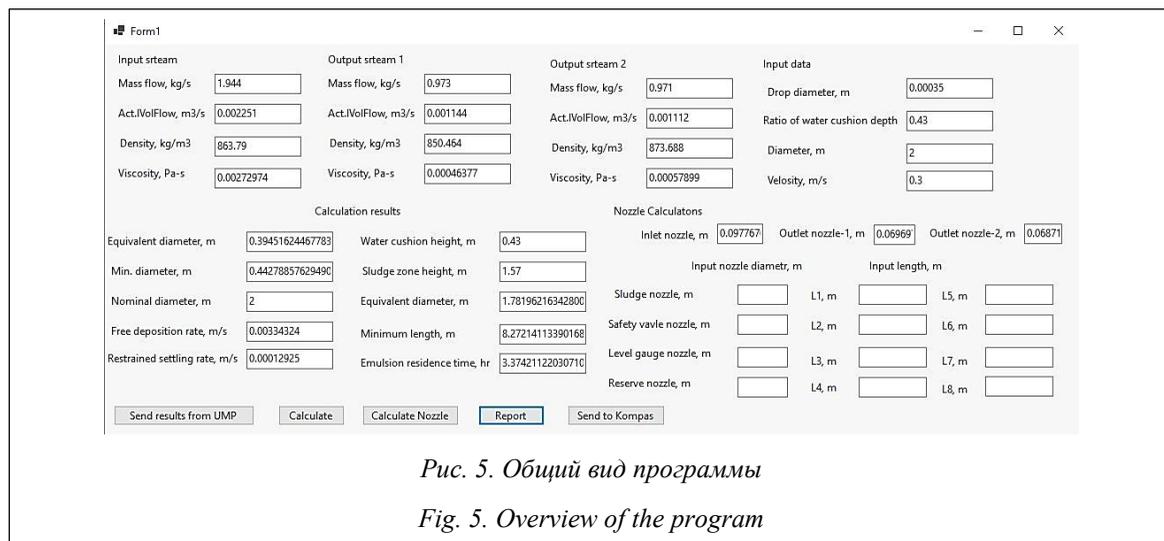


Рис. 5. Общий вид программы

Fig. 5. Overview of the program

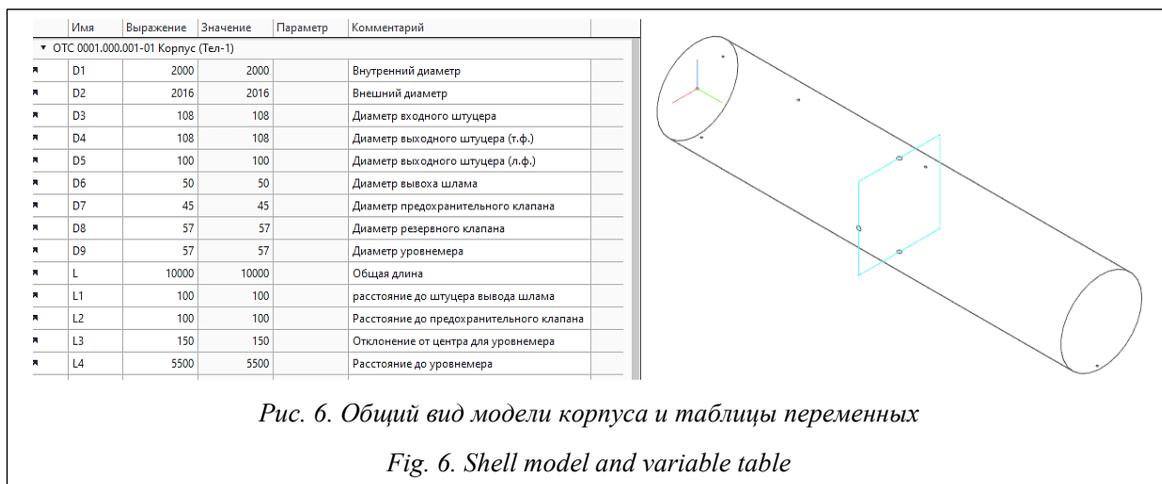


Рис. 6. Общий вид модели корпуса и таблицы переменных

Fig. 6. Shell model and variable table

Поэтому необходимо определить предельное содержание метанола в смеси, при котором плотность толуольной фазы меньше плотности водометанольной фазы. На модели (рис. 3) проведен расчет, в ходе которого изменяли содержание метанола в смеси и фиксировали плотности легкой и тяжелой фаз.

Данный расчет проведен с использованием инструмента Case Study. В качестве независимой переменной выбрано содержание воды в смеси, при изменении которого проводился пересчет содержания метанола, при этом доля толуола была закреплена. Результаты расчета представлены на рисунке 7.

На графике видно, что с увеличением содержания метанола в смеси плотность водометанольной фазы падает, а при содержании метанола более 0.3 (следовательно, содержание

воды <0.2) плотность фазы 1 меньше плотности фазы 2. Поэтому за расчетный режим принято содержание метанола в исходной смеси 0.25. Результаты расчета приведены в таблице 2.

При принятом диаметре 2 м минимальная длина отстойника составит 8.27 м. Целесообразно установить последовательно два отстойника, а длину каждого принять 5 м.

Сформирована трехмерная модель отстойника (<http://www.swsys.ru/uploaded/image/2025-1/27.jpg>). При выбранной скорости входа смеси, равной 0.3 м/с, определены размеры входного и выходных патрубков. С учетом этих размеров в БД программы подобраны стандартные штуцера с условным диаметром 100 мм по АТК 24.218.06-90. Для патрубков слива шлама выбраны штуцера  $D_y = 50$  мм по АТК 24.218.06-90, предохранительный клапан

Таблица 2

Результаты расчета

Table 2

Calculation results

Параметр	Обозначение	Значение
Относительная высота водяной подушки	$\epsilon$	0.43
Функция относительной высоты водяной подушки в зоне отстоя	$f(\epsilon)$	2.49
Диаметр капли $d$ , м [17]	$d$	0.00035
Эквивалентный диаметр, м	$D_{\text{ЭКВ}}$	0.395
Минимальный диаметр, м	$D_{\text{min}}$	0.443
Принятый диаметр, м	$D$	2
Скорость свободного осаждения, м/с	$w_o$	0.0033
Скорость стесненного осаждения, м/с	$w_{oc}$	0.00013
Высота водной подушки, м	$h_B$	0.43
Высота зоны отстоя, м	$h_{oc}$	1.57
Эквивалентный диаметр, м	$D_{\text{ЭКВ-ЭМ}}$	1.78
Минимальная длина отстойника, м	$L_{\text{min}}$	8.27
Время пребывания эмульсии в аппарате, ч	$\tau$	3.37

$D_y = 40$  мм по АТК 24.218.06-90. Днище – 2000-8-400 ГОСТ 6533-78; опора подвижная – П 250-1014-1 ОСТ 26-2091-93 и неподвижная – 250-1014-1 ОСТ 26-2091-93.

В дальнейшем модель может быть дополнена необходимыми элементами с использованием БД программы КОМПАС 3D.

### Выводы

Предлагаемый подход позволяет автоматизировать инженерные методики, сокращая тем самым время, требуемое на выполнение расчетов. Одновременно с этим функционал моделирующих программ дает возможность проводить оптимизацию, повышая тем самым энергоэффективность данного решения. Разработанная на основе подхода программа позволяет определить ТФС потоков, рассчитать оптимальные условия проведения процесса и подготовить исходные данные для импорта в САД-систему при автоматизированной разработке комплекта КД. Недостатком программы является то, что в качестве моделирующей системы используется Aspen HYSYS, однако в дальнейшем его

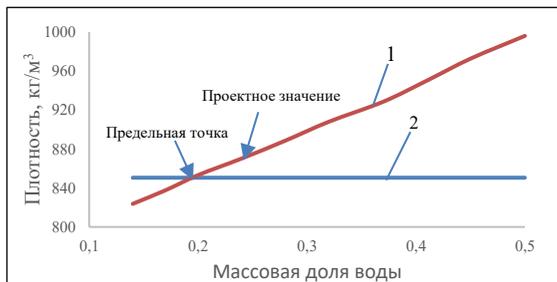


Рис. 7. Плотности фаз в зависимости от массовой доли воды в смеси:

1 – плотность водометанольной фазы;  
2 – плотность толуольной фазы

Fig. 7. Phase densities depending on water mass fraction in the mixture:

1 – water-methanol phase density;  
2 – toluene phase density

можно заменить на аналогичное по функционалу и возможностям отечественное решение. Предлагаемый подход позволяет связать САД-систему и моделирующую систему, проводить инженерный расчет нестандартного оборудования с учетом рассчитанных материальных и энергетических балансов.

### Список литературы

- Ипанов А.С., Ошивалов М.А., Галягин К.С., Селянинов Ю.А., Савин М.А., Вахрамеев Е.И. Математическое моделирование работы многофазных гидроструйных насосов // Вестн. ПНИПУ. Машиностроение, материаловедение. 2019. Т. 21. № 2. С. 15–22. doi: 10.15593/2224-9877/2019.2.02.
- Абачарасв И., Булгаков В., Кушнер Г., Мамонтов В. Автоматизация методики расчета собственной частоты поперечных колебаний гребного вала // Вестн. АГТУ. Сер.: Морская техника и технология. 2019. № 1. С. 63–70. doi: 10.24143/2073-1574-2019-1-63-70.
- Ханипов М.Н., Ганиев Б.Г., Насыбуллин А.В., Саттаров Р.З. Разработка методики вероятностного прогнозирования добычи нефти // Нефтяная провинция. 2020. Т. 22. № 2. С. 73–94. doi: 10.25689/NP.2020.2.73-94.
- Галеркин Ю.Б., Рекстин А.Ф., Солдатова К.В., Дроздов А.А., Соловьева О.А., Семеновский В.Б. Современное состояние инженерного метода оптимального газодинамического проектирования и расчета характеристик центробежных компрессоров. Ч. 3 // КТП. 2020. № 2. С. 2–9.
- Зиятдинов Н.Н. Моделирование и оптимизация химико-технологических процессов и систем // Теоретические основы химической технологии. 2017. Т. 51. № 6. С. 613–617. doi: 10.7868/S0040357117060197.
- Valverde J.L., Ferro V.R., Giroir-Fendler A. Automation in the simulation of processes with Aspen HYSYS: An academic approach. Comput. Appl. Eng. Educ., 2023, no. 31, pp. 376–388. doi: 10.1002/cae.22589.
- Мухаметжанов М.Х. Использование устройств внутритрубной обработки газожидкостных потоков для оптимизации нефтегазосборных систем // Оборудование и технологии для нефтегазового комплекса. 2022. № 6. С. 79–89. doi: 10.33285/1999-6934-2022-6(132)-79-89.
- Труханов К.Ю., Шаталов М.И., Ткачев Д.А. Импорт твердотельной модели из САД-системы КОМПАС-3D в САЕ-систему ANSYS // Электронный журнал: наука, техника и образование. 2017. № 4. С. 12–17.
- Barbero-Sánchez J., Megía-Ortega A., Ferro V.R., Valverde J.-L. Exploring alternatives to create digital twins from and for process simulation. J. of Computer Sci. Research, 2024, vol. 6, no. 1, pp. 16–30. doi: 10.30564/jcsr.v6i1.6168.
- Осипов Э.В., Хоменко А.А., Лаптева Т.В. Сопряжение элементов химико-технологической системы, функционирующей под вакуумом (на примере блока ректификации смеси аминов) // Вестн. Технологического университета. 2023. Т. 26. № 12. С. 184–190.
- Михин А.А., Сергеев В.В. Моделирование конденсационной установки в среде ASPEN PLUS // Изв. высших учебных заведений. Проблемы энергетики. 2019. Т. 21. № 6. С. 84–92. doi: 10.30724/1998-9903-2019-21-6-84-92.
- Umyshv D.R., Osipov E.V., Kibarin A.A. et al. Techno-economic analysis of the modernization options of a gas turbine power plant using Aspen HYSYS. Energies, 2023, vol. 16, no. 6, art. 2704. doi: 10.3390/en16062704.
- Афанасьев А.Н., Бригаднов С.И. Разработка экспериментальной компьютерной программы «Рекомендательная система для САПР КОМПАС-3D» // Вестн. УлГТУ. 2017. Т. 80. № 4. С. 33–36.

14. Мохова Е.К., Гордиенко М.Г. Математическая модель вакуумной сублимационной сушки с неравномерным распределением паров по объему камеры // Программные продукты и системы. 2021. Т. 34. № 3. С. 466–476. doi: 10.15827/0236-235X.135.466-476.

15. Фуфаев А.Л., Козлов П.В., Писаревский Д.В. Эффективность применения Aspen PIMS в процессах планирования ООО «Газпром нефтехим Салават» // Газовая промышленность. 2018. Т. 767. № 4. С. 72–74.

16. Ахметшина Д.А., Христовуло А.Н. Сравнение САПР для проектирования объектов нефтегазопереработки // Информационные технологии. Проблемы и решения. 2019. № 1. С. 157–163.

17. Ладенко А.А., Кунина П.С. Расчет нефтепромыслового оборудования. М.: Инфра-Инженерия, 2019. 188 с.

18. Bartolome P.S., Van Gerven T. A comparative study on Aspen Hysys interconnection methodologies. *Comput. and Chemical Eng.*, 2022, vol. 162, art. 107785. doi: 10.1016/j.compchemeng.2022.107785.

Software &amp; Systems

doi: 10.15827/0236-235X.149.134-142

2025, 38(1), pp. 134–142

### Applying specialized software packages to automate engineering equipment calculation

Eduard V. Osipov <sup>1</sup>✉, Andrey A. Khomenko <sup>1</sup>, Lilia E. Osipova <sup>1</sup>

<sup>1</sup> Kazan National Research Technological University, Kazan, 420015, Russian Federation

<sup>2</sup> Kazan State University of Architecture and Engineering, Kazan, 420043, Russian Federation

#### For citation

Osipov, E.V., Khomenko, A.A., Osipova, L.E. (2025) ‘Applying specialized software packages to automate engineering equipment calculation’, *Software & Systems*, 38(1), pp. 134–142 (in Russ.). doi: 10.15827/0236-235X.149.134-142

#### Article info

Received: 24.04.2024

After revision: 26.08.2024

Accepted: 27.08.2024

**Abstract.** The purpose of this paper is to develop an approach in order for engineering methods of equipment calculation to form as separate programs integrated with simulation-based CAD systems, and the results to be input data for a CAD-system. The research focuses on toluene settling from water-methanol mixture, the particular emphasis is on non-standard technological equipment of the settling process (settler). The authors present technological calculations for material and energy balances in a universal simulation program. They determine the geometric dimensions of non-standard equipment using the developed program. A 3D model of the equipment is formed in a graphical CAD-system. The approach is implemented in the form of a program in VBA language, which is connected with the model of a settling process. VBA language is basic within the software used and widespread in automating calculations within the system. Since the program database does not contain a settler calculation module, the authors have developed this module using the Import User Model tool. A parametric analysis was conducted using the Case Study add-on. The authors identified the maximum allowable methanol content in the initial mixture (up to 0.3 mass fraction), at which the density of the water-methanol mixture exceeds the density of the toluene phase. The authors calculated the main dimensions of the equipment using engineering methods, which are implemented as a separate program linked to the model in the simulation CAD system. The authors have developed a 3D model of the equipment body in a graphical CAD system using variables. It is the basis of the assembly of the settler, supplemented with standard elements from the program database. As a result, the developed program determines settler dimensions; it links modeling and graphical CAD system. It allows automating engineering methods of energy efficient equipment calculation.

**Keywords:** engineering calculations, software packages, VBA language, design automation, computer simulation, simulation-based CAD systems

#### References

1. Ipanov, A.S., Oshivalov, M.A., Galyagin, K.S., Selyaninov, Yu.A., Savin, M.A., Vakhrameev, E.I. (2019) ‘Mathematical modeling of multiphase hydrojet pump operation’, *Bull. PNRPU. Mechanical Eng., Materials Sci.*, 21(2), pp. 15–22 (in Russ.). doi: 10.15593/2224-9877/2019.2.02.

2. Abacharaev, I., Bulgakov, V., Kushner, G., Mamontov, V. (2019) ‘Automating the method of calculating natural frequency of propeller shaft transverse vibrations’, *Vestn. ASTU. Ser.: Marine Eng. and Tech.*, (1), pp. 63–70 (in Russ.). doi: 10.24143/2073-1574-2019-1-63-70.

3. Khanipov, M.N., Ganiev, B.G., Nasybullin, A.V., Sattarov R.Z. (2020) ‘Development of oil production probabilistic forecasting technique’, *Neftyanaya Provintsiya*, 22(2), pp. 73–94 (in Russ.). doi: 10.25689/NP.2020.2.73-94.

4. Galerkin, Yu.B., Rekstin, A.F., Soldatova, K.V., Drozdov, A.A., Solovyova, O.A., Semenovskiy, V.B. (2020) ‘The current state of the engineering method for the optimal gas-dynamic design and calculation of the centrifugal compressors characteristics. Part 3’, *СТП*, (2), pp. 2–9 (in Russ.).

5. Ziyatdinov, N.N. (2017) 'Modeling and optimization of chemical engineering processes and systems', *TFCE*, 51(6), pp. 613–617 (in Russ.). doi: 10.7868/S0040357117060197.
6. Valverde, J.L., Ferro, V.R., Giroir-Fendler, A. (2023) 'Automation in the simulation of processes with Aspen HYSYS: An academic approach', *Comput. Appl. Eng. Educ.*, (31), pp. 376–388. doi: 10.1002/cae.22589.
7. Mukhametzhano, M.Kh. (2022) 'The use of devices for in-line treatment of gas-liquid flows to optimize oil and gas gathering systems', *Equipment and Technologies for the Oil and Gas Complex*, (6), pp. 79–89 (in Russ.). doi: 10.33285/1999-6934-2022-6(132)-79-89.
8. Trukhanov, K.Yu., Shatalov, M.I., Tkachev, D.A. (2017) 'Import of a solid model from the COMPAS-3D CAD-system to an ANSYS CAE-system', *Electronic J.: Science, Technology, and Education*, (4), pp. 12–17 (in Russ.).
9. Barbero-Sánchez, J., Megía-Ortega, A., Ferro, V.R., Valverde, J.-L. (2024) 'Exploring alternatives to create digital twins from and for process simulation', *J. of Computer Sci. Research*, 6(1), pp. 16–30. doi: 10.30564/jcsr.v6i1.6168.
10. Osipov, E.V., Khomenko, A.A., Lapteva, T.V. (2023) 'Coupling of elements of chemical-technological system operating under vacuum (on the example of amine mixture rectification unit)', *Herald of Technological University*, 26(12), pp. 184–190 (in Russ.).
11. Mikhin, A.A., Sergeev, V.V. (2019) 'Simulation of condensation unit in ASPEN PLUS', *Power Engineering: Research, Equipment, Technology*, 21(6), pp. 84–92 (in Russ.). doi: 10.30724/1998-9903-2019-21-6-84-92.
12. Umyshev, D.R., Osipov, E.V., Kibarin, A.A. et al. (2023) 'Techno-economic analysis of the modernization options of a gas turbine power plant using Aspen HYSYS', *Energies*, 16(6), art. 2704. doi: 10.3390/en16062704.
13. Afanasev, A.N., Brigadnov, S.I. (2017) 'Development of the experimental computer program "Recommendation system for CAD KOMPAS-3D"', *Bull. UISTU*, 80(4), pp. 33–36 (in Russ.).
14. Mokhova, E.K., Gordienko, M.G. (2021) 'A mathematical model of vacuum freeze-dryer with uneven vapor distribution over the chamber volume', *Software & Systems*, 34(3), pp. 466–476 (in Russ.). doi: 10.15827/0236-235X.135.466-476.
15. Fufaev, A.L., Kozlov, P.V., Pisarevsky, D.V. (2018) 'Efficiency of application of Aspen PIMS in the planning processes of Gazprom neftekhim SALAVAT LLC', *Gas Industry*, 767(4), pp. 72–74 (in Russ.).
16. Akhmetshina, D.A., Khristodulo, A.N. (2019) 'Comparison of CAD for designing oil and gas processing objects', *Information Technology*, (1), pp. 157–163 (in Russ.).
17. Ladenko, A.A., Kunina, P.S. (2019) *Calculation of Oilfield Equipment*. Moscow, 188 p. (in Russ.).
18. Bartolome, P.S., Van Gerven, T. (2022) 'A comparative study on Aspen Hysys interconnection methodologies', *Comput. and Chemical Eng.*, 162, art. 107785. doi: 10.1016/j.compchemeng.2022.107785.

**Авторы**

**Осипов Эдуард Владиславович**<sup>1</sup>, д.т.н.,  
доцент, evosipov@kstu.ru

**Хоменко Андрей Александрович**<sup>1</sup>, к.т.н.,  
доцент, xomenko@kstu.ru

**Осипова Лилия Эдуардовна**<sup>1</sup>,  
к.т.н., доцент, osipovalilija@mail.ru

**Authors**

**Eduard V. Osipov**<sup>1</sup>, Dr.Sci. (Engineering),  
Associate Professor, evosipov@kstu.ru

**Andrey A. Khomenko**<sup>1</sup>, Cand. of Sci. (Engineering),  
Associate Professor, xomenko@kstu.ru

**Lilia E. Osipova**<sup>1</sup>, Cand. of Sci. (Engineering),  
Associate Professor, osipovalilija@mail.ru

<sup>1</sup> Казанский национальный исследовательский  
технологический университет,  
г. Казань, 420015, Россия

<sup>2</sup> Казанский государственный  
архитектурно-строительный университет,  
г. Казань, 420043, Россия

<sup>1</sup> Kazan National Research  
Technological University,  
Kazan, 420015, Russian Federation

<sup>2</sup> Kazan State University  
of Architecture and Engineering,  
Kazan, 420043, Russian Federation

## Моделирование температурных полей при аддитивном производстве металлических изделий

Д.Д. Какорин<sup>1</sup>, Б.И. Марголис<sup>1</sup>✉

<sup>1</sup> Тверской государственной технический университет, г. Тверь, 170026, Россия

### Ссылка для цитирования

Какорин Д.Д., Марголис Б.И. Моделирование температурных полей при аддитивном производстве металлических изделий // Программные продукты и системы. 2025. Т. 38. № 1. С. 143–149. doi: 10.15827/0236-235X.149.143-149

### Информация о статье

Группа специальностей ВАК: 2.3.3

Поступила в редакцию: 08.04.2024

После доработки: 25.04.2024

Принята к публикации: 26.04.2024

**Аннотация.** В статье обоснована необходимость исследования характера распределения температуры при аддитивном производстве металлических изделий. Рассмотрены особенности моделирования температурного поля, возникающего в процессе послойной электродуговой наплавки металлических деталей плоской геометрической формы с учетом несимметричного конвективно-радиационного теплообмена поверхности с окружающей средой. Подробно описана методика расчета температурного поля в двухмерных пространственных координатах на основе численных конечно-разностных методов. При этом учтены возможности сдвига начальной точки наплавленного слоя от края основания, изменения времени выдержки между наплаваемыми слоями и применения принудительного воздушного охлаждения металла. Рассмотрен механизм определения температуры в граничных точках конструкции, непосредственно контактирующих с расплавленным металлом. Разработана программа в среде MATLAB, позволяющая на основе заданных теплофизических характеристик наплаваемого металла, параметров конвективно-радиационного теплообмена и геометрических характеристик наплаваемой конструкции моделировать температурное поле в изделии. Приведены тексты функций TempSurfacing для моделирования температурного поля и TempDepend для учета зависимости теплофизических свойств металла от его температуры перед наплавкой нового слоя. Проверена работоспособность программы на примере наплавки одного слоя металла высотой 2 мм на металлическое основание высотой 5 мм. Показан внешний вид окна программы для ввода исходных данных, и приведены результаты моделирования температурного поля в двухмерных пространственных координатах в текстовом, а также графическом видах. Полученная модель температурного поля позволяет учесть прибавление новых частей металла по длине наплаваемого слоя, а также поэтапное увеличение высоты конструкции за счет послойного наплавления металла. Разработанная модель предназначена для установления оптимального термического цикла процесса послойного электродугового наплавления и идентификации условий теплообмена с учетом изменения рабочих параметров технологического процесса аддитивного производства.

**Ключевые слова:** моделирование, температурное поле, аддитивное производство, конвективно-радиационный теплообмен, конечно-разностные методы, программа, MATLAB, электродуговая наплавка

**Введение.** В промышленном производстве металлических изделий широкое применение находят аддитивные технологии. Высокий потенциал для реализации имеет технология послойного электродугового наплавления проволочного материала, отличающаяся высокой производительностью, простотой и доступностью используемого оборудования [1]. Однако успешное промышленное внедрение и использование данной технологии невозможно без программ, позволяющих выполнять математическое моделирование температурных полей, возникающих в процессе послойной электродуговой наплавки металла.

В настоящее время известно множество способов моделирования тепловых процессов, возникающих в процессе сварки и наплавки. Подходы, используемые для моделирования тепловых процессов при сварке деталей, широко применяются для прогнозирования каче-

ства изготавливаемой продукции. Например, моделирование на основе программы ANSYS Workbench позволяет строить точные модели распределения тепловых полей в зоне сварки для прогнозирования качества сварных соединений и оптимизации управления сварочным процессом [2]. Математическое моделирование используется также для создания численных моделей процесса капельного переноса электродного металла на основе в процессе сварки, позволяющих изучить влияние силы сварочного тока, сил поверхностного натяжения и силы тяжести на формирование капли расплавленного металла [3]. Существенным недостатком перечисленных методов является отсутствие возможности их использования для моделирования температурных полей в процессе аддитивного производства металлических изделий. В первую очередь это связано с особенностями послойного добавления новых

частей металла и с постоянным увеличением высоты наплавляемой конструкции.

Известны методы моделирования, позволяющие описывать процесс течения расплавленного металла и его тепловое состояние при аддитивном производстве путем селективного лазерного плавления присадочного металла [4, 5]. Однако из-за различий между механизмами передачи и распределения температуры у лазерных и электродуговых источников энергии и типами присадочных материалов данные методы не могут быть использованы для корректного описания процессов аддитивного производства на основе послойной электродуговой наплавки, а известные общие методики моделирования тепловых процессов не учитывают все особенности технологического процесса электродуговой наплавки [6, 7]. Таким образом, актуальной задачей является разработка метода моделирования температурных полей, возникающих в процессе аддитивного производства, с учетом использования электродугового источника энергии и проволочного присадочного материала.

#### Постановка задачи моделирования температурного поля при аддитивном производстве металлических изделий

Для решения задачи предложен подход к моделированию температурных полей в металлических изделиях геометрической формы типа стенка (рис. 1).

Для корректного описания распределения температуры при аддитивном производстве

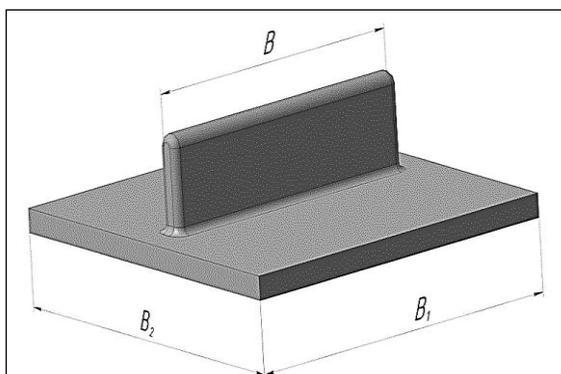


Рис. 1. Металлическое изделие геометрической формы типа стенка:  
 $B$  – длина наплавляемого слоя;  $B_1$  и  $B_2$  – длина и ширина основания изделия

Fig. 1. Geometric metal product of the wall type:  
 $B$  – welded layer length;  
 $B_1$  and  $B_2$  – product base length and width

металлических изделий необходимо учесть передачу тепла наплавляемой конструкции в процессе электродугового расплавления и переноса проволочного присадочного металла на поверхность основания. При этом по мере наплавления металла происходит перемещение источника тепловой энергии по длине слоя. Изменение температуры наплавляемой конструкции осуществляется за счет теплопроводности, а также воздушного и радиационного теплообмена внешних поверхностей с окружающим воздухом и телами.

Для расчета температурного поля  $T(x, z, \tau)$  и упрощения задачи рассматривается вертикальное сечение конструкции и решается дифференциальное уравнение теплопроводности в двухмерных пространственных координатах:

$$\frac{\partial T(x, z, \tau)}{\partial \tau} = a \left[ \frac{\partial^2 T(x, z, \tau)}{\partial z^2} + \frac{\partial^2 T(x, z, \tau)}{\partial x^2} \right], \quad (1)$$

где  $x$  – координата по длине наплавляемого слоя  $B$ ;  $z$  – координата по высоте наплавляемого изделия  $L$ ;  $a$  – коэффициент температуропроводности,  $\text{м}^2/\text{с}$ .

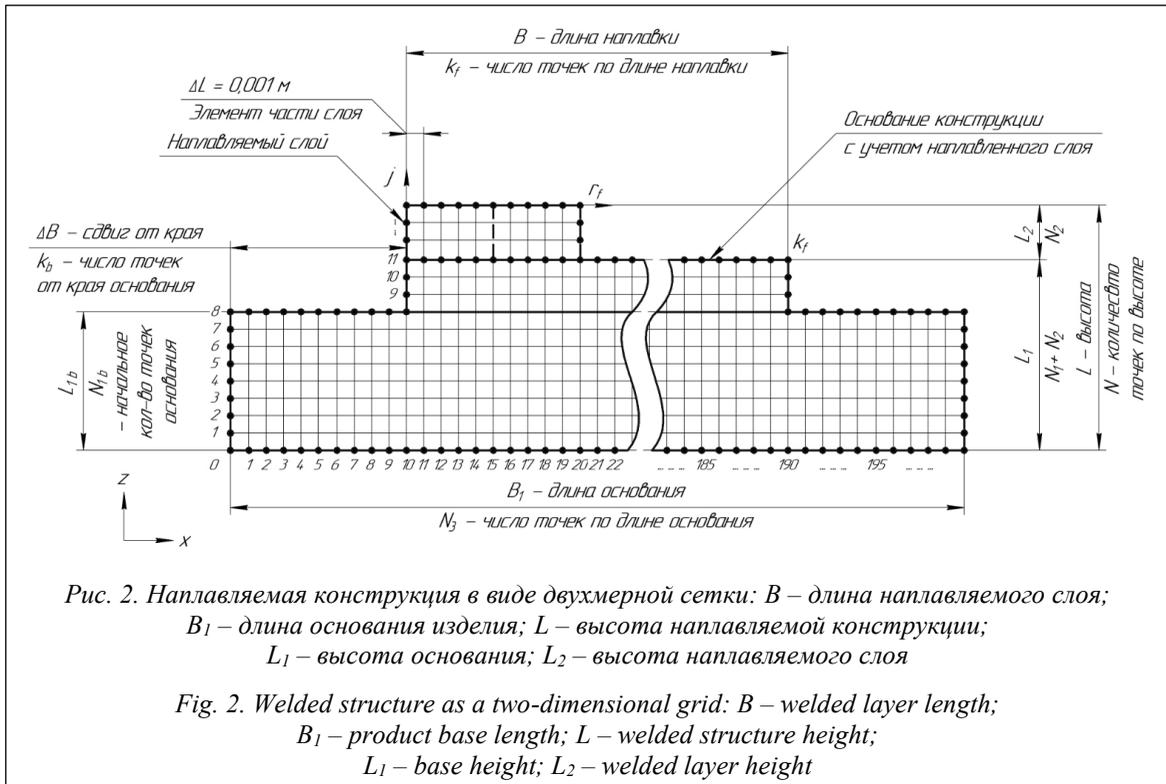
Большое значение для точного моделирования температурного поля в наплавляемом изделии играет возможность учета постоянного изменения геометрии конструкции, зависимости теплофизических свойств стали от температуры, сложных граничных условий для температур наплавляемых и соседних точек [8, 9]. В связи с этим для решения задачи теплопроводности удобно использовать явную схему конечно-разностного метода [10, 11], заключающуюся в замене непрерывной функции температуры  $T$  совокупностью ее значений, определенных на установленном множестве точек заданной области (рис. 2).

#### Разработка модели температурного поля при аддитивном производстве металлических изделий

Для решения задачи методом конечных разностей частные производные в дифференциальном уравнении теплопроводности (1) по времени 1-го порядка, по длине и высоте 2-го порядка для внутренних точек конструкции необходимо заменить на их численные аналоги:

$$\frac{T_{i+1}(j, r) - T_i(j, r)}{\Delta \tau},$$

$$\frac{T_i(j, r-1) - 2T_i(j, r) + T_i(j, r+1)}{(\Delta L)^2},$$



$$\frac{T_i(j-1,r) - 2T_i(j,r) + T_i(j+1,r)}{(\Delta L)^2},$$

где  $i = \overline{0, i_f}$  – переменная расчета по времени;  $j = \overline{0, N}$  – переменная по высоте и  $r = \overline{0, N_3}$  по длине конструкции;  $\Delta t$  и  $\Delta L$  – шаги расчета по времени и длине слоя соответственно.

Для упрощения расчетных формул и увеличения скорости вычислительных операций выделены коэффициенты, содержащие постоянные значения теплофизических характеристик:

$$GT = \frac{\Delta t \alpha}{\Delta L^2}, \quad GT_2 = 2GT, \quad GC = GT_2 \left( \frac{\alpha \Delta L}{\lambda} \right),$$

$$GC_2 = GT_2 \left( \frac{\alpha_b \Delta L}{\lambda} \right), \quad GR_1 = GT_2 \left( \frac{\varepsilon_1 \sigma \Delta L}{\lambda} \right),$$

$$GR_2 = GT_2 \left( \frac{\varepsilon_2 \sigma \Delta L}{\lambda} \right),$$

где  $\lambda$  – коэффициент теплопроводности, Вт/(м·К);  $\alpha$  и  $\alpha_b$  – коэффициенты конвективного теплообмена при естественном и вынужденном охлаждении, Вт/(м<sup>2</sup>·К);  $\sigma = 5,67 \cdot 10^{-8}$  – постоянная Стефана–Больцмана, Вт/(м<sup>2</sup>·К<sup>4</sup>);  $\varepsilon_1$  и  $\varepsilon_2$  – степени черноты для нижней и верхней поверхностей наплавляемой конструкции.

С учетом сдвига начальной точки наплавляемого слоя от границы основания на расстояние  $\Delta B$  представим формулы расчета температур

– для внутренних точек по высоте конструкции:

$$T_{i+1}(j,r) = T_i(j,r) + GT(T_i(j-1,r) - 2T_i(j,r) + T_i(j+1,r)), \quad (2)$$

где  $r = \overline{k_b, r_f + k_b - 1}$  и  $j = \overline{1, N-1}$ ;

– для нижней поверхности основания:

$$T_{i+1}(0,r) = T_i(0,r) + GT_2(T_i(1,r) - T_i(0,r)) + GC(T_c - T_i(0,r)) + CR_1(T_k - (T_i(0,r) + K)^4); \quad (3)$$

– для верхней границы наплавленного слоя:

$$T_{i+1}(N,r) = T_i(N,r) + GT_2(T_i(N-1,r) - T_i(N,r)) + GC(T_c - T_i(N,r)) + GR_2(T_k - (T_i(N,r) + K)^4), \quad (4)$$

где  $T_k = (T_c + K)^4$ ,  $T_c$  – температура окружающей среды, °С,  $K = 273,15$  град.

Затем находим температурное распределение в оставшихся частях металлического основания  $r < k_b$  и  $r > k_f$ , где  $k_f = r_f + k_b - 1$ . Для внутренних точек по высоте конструкции используется формула (2) при  $j = \overline{1, N_{1b} - 1}$ , где  $N_{1b}$  – начальное количество точек основания. Температура во внешних точках основания определяется по формулам (3) и (4) для  $N = N_{1b}$ .

Определение изменения температуры по длине металлического основания при  $j = \overline{0, N_{1b}}$  для внутренних точек при  $r = \overline{1, k_f - 1}$  производится по формуле

$$T_{i+1}(j, r) = T_{i+1}(j, r) + GT(T_i(j, r-1) - 2T_i(j, r) + T_i(j, r+1)). \quad (5)$$

Изменение температур внешних точек для левой и правой границ основания находится по формулам

$$T_{i+1}(j, 0) = T_{i+1}(j, 0) + GT_2(T_i(j, 1) - T_i(j, 0)) + GC(T_c - T_i(j, 0)) + GR_1(T_k - (T_i(j, 0) + K)^4); \quad (6)$$

$$T_{i+1}(j, N_3) = T_{i+1}(j, N_3) + GT_2(T_i(j, N_3 - 1) - T_i(j, N_3)) + GC(T_c - T_i(j, N_3)) + GR_1(T_k - (T_i(j, N_3) + K)^4). \quad (7)$$

Рассчитываются значения температур внутренних точек по длине наплавленного слоя металла при  $r = \overline{k_b + 1, k_f - 1}$  по формуле (5).

Определяются температуры внешних точек для левой границы слоя:

$$T_{i+1}(j, k_b) = T_{i+1}(j, k_b) + GT_2(T_i(j, k_b + 1) - T_i(j, k_b)) + GC(T_c - T_i(j, k_b)) + GR_2(T_k - (T_i(j, k_b) + K)^4) \quad (8)$$

и правой:

$$T_{i+1}(j, k_f) = T_{i+1}(j, k_f) + GT_2(T_i(j, k_f - 1) - T_i(j, k_f)) + GC(T_c - T_i(j, k_f)) + GR_2(T_k - (T_i(j, k_f) + K)^4). \quad (9)$$

Определяется температурное распределение по длине наплавленного слоя с учетом наплавления новых частей металла  $r_f > 1$  для внутренних точек по формуле (5) при  $r = \overline{k_b + 1, k_b + r_f - 1}$ , для левой границы по формуле (8) при  $j = \overline{N_1 + 1, N}$  и для правой границы  $r_r = \overline{k_b + r_f - 1}$  по формуле

$$T_{i+1}(j, r_r) = T_{i+1}(j, r_r) + GT_2(T_i(j, r_r - 1) - T_i(j, r_r)) + GC(T_c - T_i(j, r_r)) + GR_2(T_k - (T_i(j, r_r) + K)^4). \quad (10)$$

### Программная реализация расчета температурного поля в процессе послойной электродуговой наплавки

Приведенные формулы позволяют определить распределение температуры по всей конструкции в процессе послойной электродуговой наплавки новых слоев металла. При использовании принудительного охлаждения наплавленной части изделия коэффициент  $GC$  принимается равным  $GC_2$ . Для определения температурного распределения в наплаваемой конструкции используется функция TempSurfacing (<http://www.swsys.ru/uploaded/image/2025-1/14.jpg>).

В процессе наплавления новой части слоя температура на границе определяется путем взвешивания температур соседней точки и расплава  $T_m$  с весами  $w_1$  и  $w_2$  соответственно. Для одинаковых по характеристикам материалов подложки и наплавки можно принять  $w_1 = w_2 = 0,5$ . В процессе наплавки последующих слоев металла ( $sl > 1$ ) для расчета температурного поля в формулах (2)–(10) осуществляется корректировка количества слоев для основания  $N_1 = N_1 + N_2$  и для всей конструкции  $N = N + N_2$ .

Учет зависимости теплофизических свойств металла (коэффициенты теплопроводности  $\lambda$  и теплоемкости  $C$ ) от температуры производится перед наплавкой нового слоя функцией TempDepend. Код функции:

```
function [lam,a,dtau_max]=TempDepend(mas,ro);
global dL Tb Tt lam1 lam2 C1 C2
mas1=nonzeros(mas);
sm=sum(mas1)/length(mas1); % средняя температура
if (sm<=Tb)
    lam=lam1; C=C1;
elseif (sm>=Tt)
    lam=lam2; C=C2;
else
    lam=lam1+(sm-Tb)*(lam2-lam1)/(Tt-Tb); C=C1+(sm-Tb)*(C2-C1)/(Tt-Tb); end;
a=lam/(ro*C);
```

Рассчитывается средняя температура наплавленной конструкции  $sm$ . При этом, если  $sm$  не больше нижнего или не меньше верхнего граничных значений температур  $T_b$  и  $T_t$ , то значения коэффициентов  $\lambda$  и  $C$  берутся из справочных данных для температур соответственно. В противном случае коэффициенты находятся путем линейной интерполяции. На основании полученных коэффициентов корректируется значение коэффициента теплопроводности  $a$ .

Приведенные функции программы позволяют рассчитать температурное поле в наплавляемой конструкции на всех этапах аддитивного производства с учетом изменения тепло- и температуропроводности стали в зависимости от температуры, а также принудительного конвективного охлаждения в процессе межслойной выдержки.

Для проверки работоспособности программы с помощью функции TempSurfacing был произведен расчет температурного поля в процессе послойной электродуговой наплавки одного слоя металла при следующих исходных данных:  $B = 100$  мм,  $B_1 = 125$  мм,  $B_2 = 70$  мм,  $\Delta B = 20$  мм,  $L_1 = 5$  мм,  $L_2 = 2$  мм,  $\Delta L = 1$  мм,  $\Delta t = 0,01$  с, сталь 09Г2С,  $\lambda = 80$  Вт/(м·град.),  $\alpha = 12$  Вт/(м<sup>2</sup>·град.),  $C = 460$  Дж/(кг·К),  $\sigma = 5,67 \cdot 10^{-8}$  Вт/(м<sup>2</sup>·К<sup>4</sup>),  $\varepsilon_1 = \varepsilon_2 = 0,8$ ,  $\rho = 7800$  кг/м<sup>3</sup>,  $T_0 = 20$  °С,  $T_b = 20$  °С,  $T_i = 1\ 000$  °С,  $\lambda_1 = 55$  Вт/(м·град.),  $\lambda_2 = 28$  Вт/(м·град.),  $C_1 = 470$  Дж/(кг·К),  $C_2 = 690$  Дж/(кг·К).

На рабочем экране программы вводятся исходные данные и представляются результаты моделирования в численном виде (<http://www.swsys.ru/uploaded/image/2025-1/28.jpg>).

Результаты моделирования температурного поля отражаются в двухмерных пространственных координатах (<http://www.swsys.ru/uploaded/image/2025-1/15.jpg>), температурное поле наплавленной конструкции – в графическом виде (рис. 3).

### Заключение

Разработанная программа для моделирования температурного поля в металлических

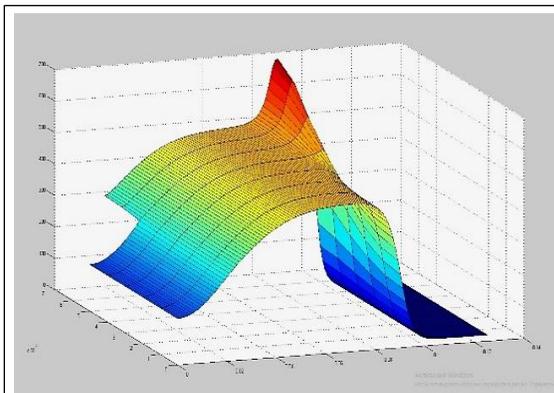


Рис. 3. Результаты моделирования температурного поля с помощью функции TempSurfacing в графическом виде

Fig. 3. Graphical results of temperature field modeling with TempSurfacing function

изделиях позволяет определить температурное распределение в конструкции при послойной электродуговой наплавке металла. Полученная модель температурного поля используется для определения оптимального термического цикла процесса аддитивного производства металлических изделий. Приведенная методика позволяет рассчитать температурное поле с учетом особенностей переноса тепла в процессе аддитивного производства металлических изделий и перейти к решению задач моделирования для изделий сложной геометрической формы. Полученную модель можно в дальнейшем использовать для идентификации условий теплообмена с учетом изменения рабочих параметров технологического процесса послойной электродуговой наплавки металла.

### Список литературы

1. Какорин Д.Д., Лаврентьев А.Ю., Марголис Б.И. Способ аддитивного производства металлических изделий // Интеллектуальные системы в производстве. 2023. Т. 21. № 4. С. 75–80. doi: 10.22213/2410-9304-2023-4-75-80.
2. Палаев А.Г., Носов В.В., Красников А.А. Моделирование распределения температурных полей и напряжений в сварном соединении с применением ANSYS // Наука и технологии трубопроводного транспорта нефти и нефтепродуктов. 2022. Т. 12. № 5. С. 461–469. doi: 10.28999/2541-9595-2022-12-5-461-469.
3. Кулагин А.Г. Математическая модель переноса расплавленного металла при дуговой сварке в среде защитных газов // Ползуновский альманах. 2022. Т. 2. № 4. С. 38–50.
4. Быков А.Н., Вишнякова М.Н., Дерюгин Ю.Н. и др. Численное моделирование селективного лазерного плавления методом SPH // Журнал Средневолжского математического общества. 2022. Т. 24. № 4. С. 419–435. doi: 10.15507/2079-6900.24.202204.419-435.
5. Григорьянц А.Г. Моделирование процесса селективного лазерного плавления в аддитивном производстве // Научно-технические проблемы машиностроения. 2023. № 10. С. 20–29. doi: 10.30987/2223-4608-2023-20-29.
6. Черных Н.Ю., Гусева Е.А. Моделирование процессов аддитивного производства // Авиационное машиностроение и транспорт Сибири: сб. ст. XV Всерос. науч.-технич. конф. 2021. С. 120–126.
7. Салихова Н.К., Осолков А.А., Келлер И.Э., Трушников Д.Н. Численное моделирование процесса 3D-наплавки проволочных материалов и согласованный термомеханический анализ // XIII Всероссийский съезд по теоретической и прикладной механике: сб. тезисов докладов. 2023. Т. 3. С. 168–169.
8. Тихомиров С.А., Тихомиров К.С. Математическое моделирование нестационарного конвективного охлаждения многослойных тел при аддитивном производстве изделий // Инженерный вестн. Дона. 2022. № 7. С. 170–191.

9. Рьльков Е.Н., Курушкин Д.В., Кладов И.В., Панченко О.В. Конечно-элементная модель тепловых процессов при послойной электродуговой наплавке для оценки деформаций и напряженного состояния // Сварка и диагностика. 2020. № 3. С. 25–29.

10. Галимов В.Р., Лазунова В.О. Сравнение аналитической и численной модели расчета температурного поля при линейной сварке трением // Молодежный вестн. УГАТУ. 2020. № 2. С. 23–26.

11. Зыбин И.Н., Антохин М.С. Численное моделирование температурных полей при сварке трением с перемешиванием алюминиевого сплава AA5083 // Frontier Materials & Technologies. 2023. № 1. С. 23–32. doi: 10.18323/2782-4039-2023-1-23-32.

Software &amp; Systems

doi: 10.15827/0236-235X.149.143-149

2025, 38(1), pp. 143–149

## Temperature field modeling in additive manufacturing of metal products

Daniil D. Kakorin <sup>1</sup>, Boris I. Margolis <sup>1</sup>✉<sup>1</sup> Tver State Technical University, Tver, 170026, Russian Federation

### For citation

Kakorin, D.D., Margolis, B.I. (2025) 'Temperature field modeling in additive manufacturing of metal products', *Software & Systems*, 38(1), pp. 143–149 (in Russ.). doi: 10.15827/0236-235X.149.143-149

### Article info

Received: 08.04.2024

After revision: 25.04.2024

Accepted: 26.04.2024

**Abstract.** The paper substantiates the necessity to study the nature of temperature distribution in additive manufacturing of metal products. It considers the peculiarities of modeling a temperature field, which arises during layer-by-layer electric arc welding of flat geometric metal parts taking into account asymmetric convection-radiation heat exchange between the surface and the environment. The authors described in detail the temperature field calculation in two-dimensional spatial coordinates based on numerical finite-difference methods. They took into account the possibility of shifting the starting point of the welded layer from the base edge, changing the holding time between the welded layers and the use of forced air metal cooling. The authors considered the mechanism of temperature determination in the structure boundary points, directly in contact with the molten metal. They also developed a program in MatLab that allows modeling the temperature field in the product based on specified thermophysical characteristics of the clad metal, parameters of convective-radiation heat exchange and geometric characteristics of the welded structure. The paper gives texts of TempSurfacing functions for modeling the temperature field and TempDepend for taking into account the dependence of thermophysical properties of metal on its temperature before surfacing a new layer. The authors checked the performance of the program on the example of surfacing a single layer of metal 2 mm high on a metal base 5 mm high. There is a view of the program window for inputting initial data, and the results of modeling the temperature field in two-dimensional spatial coordinates in text and graphical forms. The obtained temperature field model allows taking into account the new metal parts addition along the welded layer length, as well as the systematic increase in the structure height due to layer-by-layer metal welding. The developed model serves to establish the optimal thermal cycle of the layer-by-layer arc welding process and to identify the heat transfer conditions taking into account changes in the operating parameters of the additive manufacturing technological process.

**Keywords:** modeling, temperature field, additive manufacturing, convection-radiation heat exchange, finite-difference methods, program, MatLab, electric arc welding

### References

1. Kakorin, D.D., Lavrentiev, A.Yu., Margolis, B.I. (2023) 'Method of additive manufacturing of metal products', *Intellekt. Sist. Proizv.*, 21(4), pp. 75–80 (in Russ.). doi: 10.22213/2410-9304-2023-4-75-80.
2. Palaev, A.G., Nosov, V.V., Krasnikov, A.A. (2022) 'Simulating distribution of temperature fields and stresses in welded joint using ANSYS', *Sci. & Tech.: Oil and Oil Products Pipeline Transportation*, 12(5), pp. 461–469 (in Russ.). doi: 10.28999/2541-9595-2022-12-5-461-469.
3. Kulagin, A.G. 'Mathematical model of molten metal transfer during shielded gas arc welding', *Polzunov Almanac*, 4(2), pp. 38–50 (in Russ.).
4. Bykov, A.N., Vishnyakova, M.N., Deryugin, Yu.N. et al. (2022) 'Numerical simulation of selective laser melting by the SPH method', *Middle Volga Math. Society J.*, 24(4), pp. 419–435 (in Russ.). doi: 10.15507/2079-6900.24.202204.419-435.
5. Grigoryants, A.G. (2023) 'Modeling of the selective laser melting process in additive manufacturing', *Sci. Intensive Tech. in Mech. Eng.*, 23(10), pp. 20–29 (in Russ.). doi: 10.30987/2223-4608-2023-20-29.

6. Chernykh, N.Yu., Guseva, E.A. (2021) 'Modeling of additive manufacturing processes', *Proc. XV All-Russ. Sci. and Tech. Conf. Aircraft Engineering and Transport of Siberia*, pp. 120–126 (in Russ.).
7. Salikhova, N.K., Oskolkov, A.A., Keller, I.E., Trushnikov, D.N. (2023) 'Numerical modeling of 3D wire melting process and coordinated thermomechanical analysis', *Proc. XIII All-Russian Congress on Theoretical and Applied Mechanics*, 3, pp. 168–169 (in Russ.).
8. Tikhomirov, S.A., Tikhomirov, K.S. (2022) 'Mathematical modeling of non-stationary cooling 3d-printed multilayer objects', *Engineering J. of Don*, (7), pp. 170–191 (in Russ.).
9. Rylkov, E.N., Kurushkin, D.V., Kladov, I.V., Panchenko, O.V. (2020) 'The ultimate elemental model of thermal processes in layered electrode surfacing to assess deformations and tense state', *Welding and Diagnostics*, (3), pp. 25–29 (in Russ.).
10. Galimov, V.R., Lazunova, V.O. (2020) 'Analysis and numerical model comparison of temperature field calculation for linear friction welding', *Youth Bull. of the USATU*, (2), pp. 23–26 (in Russ.).
11. Zybin, I.N., Antokhin, M.S. (2023) 'Numerical modeling of temperature fields during friction stir welding of the AA5083 aluminum alloy', *Frontier Materials & Technologies*, (1), pp. 23–32 (in Russ.). doi: 10.18323/2782-4039-2023-1-23-32.

**Авторы**

**Какорин Даниил Дмитриевич**<sup>1</sup>,  
аспирант, kakorin0000@gmail.com  
**Марголис Борис Иосифович**<sup>1</sup>, д.т.н.,  
профессор, borismargolis@yandex.ru

<sup>1</sup> Тверской государственный технический  
университет, г. Тверь, 170026, Россия

**Authors**

**Daniil D. Kakorin**<sup>1</sup>, Postgraduate Student,  
kakorin0000@gmail.com  
**Boris I. Margolis**<sup>1</sup>, Dr.Sci. (Engineering),  
Professor, borismargolis@yandex.ru

<sup>1</sup> Tver State Technical University,  
Tver, 170026, Russian Federation

## Оптимизация техпроцессов раскроя деталей с использованием базы данных стоимостных параметров процесса листовой резки

А.Ф. Таваева<sup>1, 2</sup>✉, А.А. Петунин<sup>2, 3</sup>

<sup>1</sup> Уральский оптико-механический завод, г. Екатеринбург, 620100, Россия

<sup>2</sup> Уральский федеральный университет, г. Екатеринбург, 620002, Россия

<sup>3</sup> Институт математики и механики им. Н.Н. Красовского  
Уральского отделения РАН, г. Екатеринбург, 620108, Россия

### Ссылка для цитирования

Таваева А.Ф., Петунин А.А. Оптимизация техпроцессов раскроя деталей с использованием базы данных стоимостных параметров процесса листовой резки // Программные продукты и системы. 2025. Т. 38. № 1. С. 150–156. doi: 10.15827/0236-235X.149.150-156

### Информация о статье

Группа специальностей ВАК: 2.3.3, 2.3.7

Поступила в редакцию: 18.06.2024

После доработки: 21.07.2024

Принята к публикации: 27.07.2024

**Аннотация.** В статье рассматривается интегрированная задача раскроя и маршрутизации, объединяющая задачи оптимизации раскроя и минимизации стоимости процесса резки на технологическом оборудовании с числовым программным управлением (ЧПУ). Для ее решения и разработки соответствующих оптимизационных алгоритмов, эффективных на практике, необходимо получить научно обоснованные данные о значениях стоимостных параметров процесса листовой резки на технологическом оборудовании с ЧПУ для различных марок и толщин обрабатываемых материалов. В данной статье описывается база данных таких стоимостных параметров этого процесса, приводятся ее схема, основная структура и организация. Для хранения данных стоимостных параметров выбрана реляционная модель, которая состоит из восьми таблиц и содержит информацию о марках и толщинах обрабатываемых материалов, значениях стоимости и плотности материалов, стоимости единицы пути режущего инструмента на рабочем и холостом ходу, а также одной точки врезки. Для удобства работы с данными (визуализация, добавление новых записей, удаление записей, расчет стоимостных параметров для новых марок материалов и толщин, изменение значений параметров), хранящимися в базе, разработано ПО на языке Python. Его можно использовать либо как отдельный продукт, либо совместно с ПО для автоматического проектирования управляющих программ для оборудования листовой резки с ЧПУ при решении практических задач. Значимость работы показана на модельном примере проектирования раскроя и маршрутизации инструмента для одного типа машины лазерной резки.

**Ключевые слова:** оптимизация раскроя листового материала, оптимизация маршрутизации, интегрированная задача раскроя и маршрутизации, база данных, стоимостные параметры

**Благодарности.** Работа выполнена при финансовой поддержке РФФИ (грант № 23-21-00343, <https://rscf.ru/project/23-21-00343/>)

**Введение.** Во многих отраслях промышленности одним из современных и востребованных методов обработки листового материала на промежуточном этапе работы с металлопрокатом является резка. Во время этого процесса материал разделяют на заготовки различных размеров и форм. Для резки листового материала используют в основном оборудование термической (лазерной, плазменной, кислородной) и гидроабразивной резки с ЧПУ.

Проектирование управляющих программ для оборудования листовой резки с ЧПУ осуществляется с помощью систем автоматизированного проектирования. Однако из-за сложности решаемых оптимизационных задач они не обеспечивают получение оптимального варианта раскроя листового материала (минимальная стоимость использованного при раскрое материала) и маршрута перемещения режущего инструмента (минимальная стоимость

процесса резки). В частности, в [1] показано, что наиболее приемлемое решение задачи маршрутизации инструмента, полученное для оптимального варианта раскроя листового материала, не гарантирует экстремума стоимостного критерия интегрированной задачи раскроя и маршрутизации. Интегрированная задача объединяет две известные проблемы – оптимизация раскроя листового материала и минимизация стоимости процесса резки на технологическом оборудовании с ЧПУ.

Разработка концепции единого стоимостного критерия для интегрированной задачи предполагает необходимость получения научно обоснованных данных о реальной стоимости резки на разном технологическом оборудовании с ЧПУ для различных марок и толщин материалов. Получение данных такого рода является хорошей предпосылкой для создания оптимизационных алгоритмов, эффективных

для решения практических задач, однако эта область остается малоисследованной. Малочисленны исследования, в которых задача оптимизации раскроя материала рассматривается совместно с задачей оптимизации маршрута резки. Можно отметить, в частности, работу [2], в которой предложены метод для минимизации расхода материала и пути режущего инструмента за счет применения совмещенного реза. В [3] предлагается математическая модель, объединяющая задачи упаковки заготовок в полосе и минимизации длины реза этих заготовок. Целевая функция при этом не является интегрированным стоимостным критерием. Достоинство работы заключается в возможности получения в рамках предложенной модели точных решений, однако, как известно, это реально только для задач малой размерности. В работе [4] рассматриваются интегрированная задача упаковки заготовок в обрезках листовых материалов неправильной формы, которые остаются в больших количествах в заготовительном производстве, и задача оптимизации холостого хода режущего инструмента. Вторая вызывает преимущественно только теоретический интерес, а на практике наиболее интересна минимизация рабочего хода режущего инструмента.

Следует также отметить, что в большинстве научных публикаций задачи оптимизации раскроя и маршрутизации инструмента машин листовой резки исследуются отдельно. Обзор методов для решения задач оптимизации нерегулярного раскроя приведен, например, в [5, 6]. Обычно выделяют три основных подхода к решению: методы математического программирования, эвристические алгоритмы и методы машинного обучения [7]. Методы математического программирования могут быть использованы для учета сложных ограничений и получения оптимальных решений [3, 8]. При этом глобальная оптимизация стоимости раскроя в реальном диапазоне размерностей возможна только для простых геометрических форм [9]. Для решения производственных задач наиболее часто используются эвристические алгоритмы [10, 11]. В последние годы исследователи начали применять методы машинного обучения для решения задач двумерного нерегулярного раскроя [12, 13].

Число публикаций по вопросам минимизации стоимости процесса резки уступает их количеству по оптимизации раскроя листового материала, но неуклонно растет. Одним из подходов к решению задач маршрутизации инстру-

мента для машин листовой резки является использование моделей задачи коммивояжера. В частности, обзор существующих алгоритмов решения обобщенной задачи коммивояжера приведен в [14]. Задачи небольшой размерности могут быть решены с помощью точных алгоритмов [15]. Однако большинство методов ориентировано на применение эвристических алгоритмов, например, в [16] предложены эволюционные алгоритмы для оптимизации рабочих и холостых перемещений режущего инструмента для лазерного комплекса.

Таким образом, следует отметить, что в современных исследованиях не приводятся научно обоснованные данные о реальной стоимости процесса резки на конкретном технологическом оборудовании с ЧПУ, знание которых необходимо для разработки оптимизационных алгоритмов, эффективных при решении практических задач. В настоящей работе предлагается БД стоимостных параметров, приводятся схема БД, а также ее основная структура и организация. В БД были добавлены данные стоимостных параметров для лазерного СО2 комплекса с ЧПУ.

### Материалы и методы

Предложенная в данной статье БД Cost Parameters содержит информацию о значениях стоимостных параметров для технологического оборудования с ЧПУ. Для хранения данных о стоимостных параметрах была выбрана реляционная модель, ее схема приведена на рисунке 1.

БД Cost Parameters состоит из восьми таблиц, которые содержат следующую информацию:

**material** – марки материалов (*name\_material*);

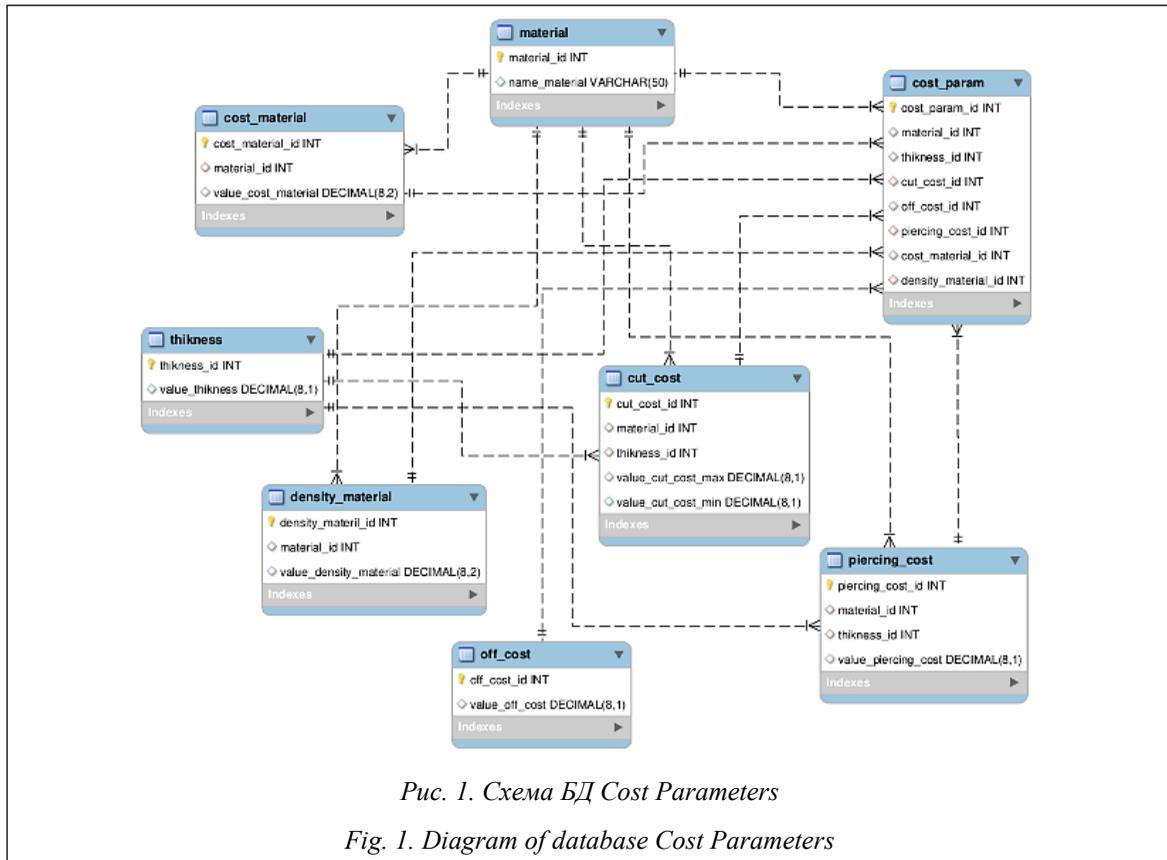
**thickness** – толщины материалов (*value\_thickness*);

**density\_material** – плотность *value\_density\_material* для марок материалов из таблицы *material* (*material\_id*);

**cost\_material** – стоимость *value\_cost\_material* для марок материалов из таблицы *material* (*material\_id*);

**cut\_cost** – стоимость единицы пути режущего инструмента на рабочем ходу *value\_cut\_cost* для марок материалов и соответствующих толщин из таблиц *material* и *thickness* (соответственно *material\_id*, *thickness\_id*);

**off\_cost** – стоимость единицы пути режущего инструмента на холостом ходу (*value\_off\_cost*);



**piercing\_cost** – стоимость одной точки врезки  $value\_piercing\_cost$  для марок материалов и соответствующих толщин из таблиц **material** и **thickness** (соответственно  $material\_id$ ,  $thickness\_id$ );

**cost\_param** – стоимостные параметры для марок материалов и соответствующих толщин из таблиц **material** и **thickness** соответственно.

Для удобства работы с данными (визуализация БД, добавление новых записей, удаление записей из БД, расчет стоимостных параметров для новых марок материалов и толщин, изменение значений параметров), хранящимся в БД Cost Parameters, было разработано одноименное ПО на языке Python. Основные функциональные блоки ПО: интерфейс ввода данных, интерфейс корректировки, интерфейс для расчета стоимостных параметров, интерфейс выдачи.

**Интерфейс ввода данных** реализован в виде графического интерфейса для добавления новой записи в БД. Пользователю необходимо либо выбрать марку материала и толщину из значений, уже хранящихся в БД, либо ввести новые. Затем вводятся значения следующих параметров для выбранного материала:  $C_{on\_max}$ ,  $C_{on\_min}$  – стоимость единицы пути режущего инструмента на рабочем ходу при максимально и

минимально возможных значениях скоростей резки,  $C_{off}$  – стоимость единицы пути режущего инструмента на холостом ходу,  $C_{pt}$  – стоимость одной точки врезки,  $C_m$  – стоимость материала,  $p$  – плотность материала.

**Интерфейс для расчета стоимостных параметров** реализован в виде графического интерфейса (<http://www.swsys.ru/uploaded/image/2025-1/29.jpg>).

Пользователю необходимо выбрать марку материала и значение толщины из БД либо добавить свои значения, после чего ввести значения параметров для дальнейшего расчета  $C_{on\_max}$ ,  $C_{on\_min}$ ,  $C_{off}$ ,  $C_{pt}$  и сохранения соответствующей записи в БД. За основу для расчета значений стоимостных параметров была взята методика из [17].

**Интерфейс выдачи** обращается к БД и переводит эти результаты в таблицу (рис. 2).

Используя методику [17], в БД Cost Parameters были добавлены значения стоимостных параметров на примере лазерного CO<sub>2</sub> комплекса листовой резки с ЧПУ.

Разработанное ПО Cost Parameters может использоваться либо как отдельный продукт, либо совместно с ПО для автоматического проектирования управляющей программы для оборудования листовой лазерной резки с ЧПУ

Cost Parameters								
id	Material	Thickness	Con	Coff	Cpt	Cm	density	
1	1	st10kp	1	5,4	0,42	0,74	79	7856
2	2	st10kp	1,2	6,64	0,42	0,97	79	7856
3	3	st10kp	1,5	6,64	0,42	1,12	79	7856
4	4	st10kp	2	8,14	0,42	1,26	79	7856
5	5	st10kp	2,5	9,7	0,42	1,49	79	7856
6	6	st10kp	3	12,01	0,42	1,56	79	7856
7	7	st10kp	3,5	13,27	0,42	1,64	79	7856
8	8	st10kp	3,9	13,27	0,42	1,86	79	7856
9	9	st10kp	4	14,84	0,42	2,16	79	7856
10	10	st10kp	5	17,91	0,42	2,75	79	7856
11	11	st10kp	8	26,06	0,42	3,43	79	7856
12	12	st10kp	10	31,85	0,42	5,15	79	7856
13	13	st10kp	15	52,11	0,42	6,87	79	7856
14	14	amg3m	1	11,15	0,42	3,71	390	2670
15	15	amg3m	2	18	0,42	5,56	390	2670

0, 0 мм

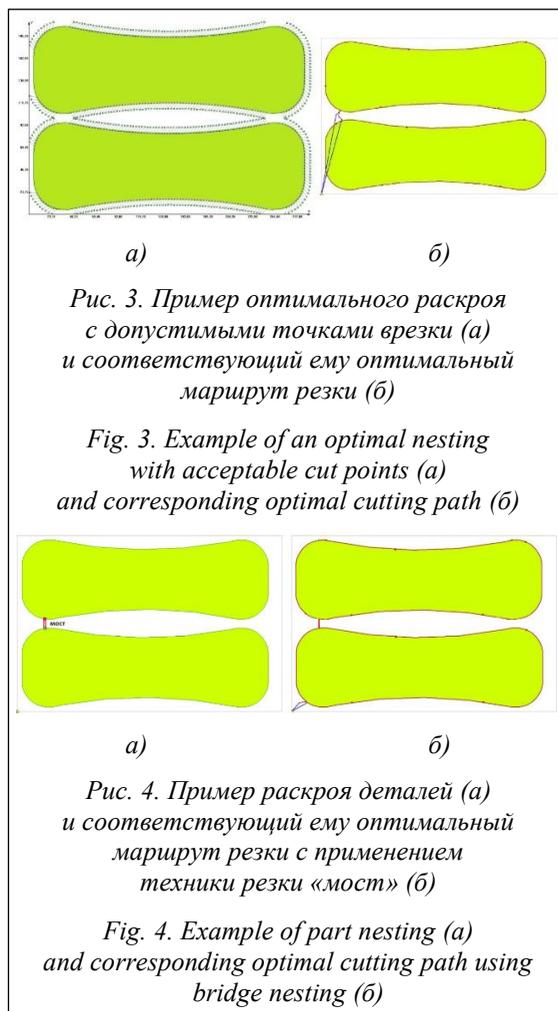
Рис. 2. Таблица стоимостных параметров

Fig. 2. Table of Cost Parameters

в качестве отдельного модуля, например, с ПО, разработанным в [18].

В качестве примера для расчета стоимости процесса резки рассмотрим упаковку двух заготовок (рис. 3) в прямоугольной полосе шириной 200 мм. Для варианта раскроя, который является оптимальным решением задачи 2D-раскроя, на рисунке приведены допустимые точки врезки и соответствующий ему оптимальный маршрут резки с применением стандартной техники. Построить оптимальный маршрут перемещения режущего инструмента позволяет алгоритм динамического программирования. Значения стоимости процесса резки рассчитаны для алюминиевого листа Амг3М толщиной 5 мм. Данные по стоимостным параметрам взяты из БД Cost Parameters, предложенной в данной статье. Для приведенного на рисунке примера длина рабочих перемещений режущего инструмента  $L_{on} = 1383$  мм, количество точек врезки  $N_{pt} = 2$  (равно количеству вырезаемых контуров). Соответственно стоимость процесса резки  $C_{cut}$  составила 331.65 руб.

На рисунке 4 показаны вариант вырезки деталей для раскроя (рис. 3) с применением техники резки «мост», а также маршрут резки. Детали вырезаются с одной точкой врезки, без выключения режущего инструмента при переходе от первой детали ко второй за счет наличия «моста». Для данного примера длина рабочих перемещений режущего инструмента  $L_{on} = 1383 + 20 = 1403$  мм (длина «моста» 10 мм



в каждую сторону), количество точек врезки  $N_{pt} = 1$ , соответственно стоимость процесса резки  $C_{cut} = 303.18$  руб.

Таким образом,  $C_{cut}$  при резке деталей с помощью техники «мост» (рис. 4) оказывается меньше, чем  $C_{cut}$  для стандартной техники резки (рис. 3), на 8.5 %. При других значениях стоимостных параметров итоговая стоимость процесса резки на стандартной технике может оказаться ниже либо равной стоимости резки при использовании техники резки «мост». Например, для листа марки Ст10кп толщиной 1 мм итоговые значения стоимости процесса резки  $C_{cut}$  для примера на рисунках 3 и 4 практически равны.

Очевидно, что получение научно обоснованных данных о значениях стоимости резки на различном технологическом оборудовании с ЧПУ для разных марок и толщин обрабатываемых материалов является неотъемлемым процессом при разработке оптимизационных алгоритмов, эффективных для решения практических задач.

## Выводы

В статье предложена общая схема БД Cost Parameters для стоимостных параметров процесса листовой резки на технологическом оборудовании с ЧПУ, приведены ее основная структура и организация.

Для хранения данных стоимостных параметров выбрана реляционная модель, которая содержит информацию о марках и толщинах обрабатываемых материалов, значениях стоимости и плотности материалов, стоимости единицы пути режущего инструмента на рабочем и холостом ходу, а также одной точки врезки.

Для удобства работы с данными (визуализация БД, добавление новых записей, удаление

записей из БД, расчет стоимостных параметров для новых марок материалов и толщин, изменение значений параметров), хранящимися в БД Cost Parameters, разработано одноименное ПО на языке программирования Python.

Разработанное ПО может использоваться либо как отдельный продукт, либо совместно с ПО для автоматического проектирования управляющих программ для оборудования листовой резки с ЧПУ при решении практических задач в качестве отдельного модуля.

На простом примере показано получение научно обоснованных данных о реальной стоимости процесса резки на различном технологическом оборудовании с ЧПУ для разных марок и толщин обрабатываемых материалов.

## Список литературы

1. Петунин А.А., Котел Н.С., Таваева А.Ф. Об одном примере оптимального решения интегрированной задачи 2D-раскроя и маршрутизации инструмента машин листовой резки с ЧПУ // Вестн. Югорского государственного университета. 2023. № 4. С. 88–101. doi: 10.18822/byusu20230488-101.
2. Hu Q., Lin Z., Fu J., Luan C. Optimizing cutting sequences and paths for common-edge nested parts. *Computer-Aided Design*, 2023, vol. 164, art. 103610. doi: 10.1016/j.cad.2023.103610.
3. Oliveira L.T., Silva E.F., Oliveira J.F., Toledo F.M.B. Integrating irregular strip packing and cutting path determination problems: A discrete exact approach. *CAIE*, 2020, vol. 149, art. 106757. doi: 10.1016/j.cie.2020.106757.
4. Li L., Wu Z., Zhang Z., Zhang Y. An optimization method for CNC laser combination cutting of irregular plate remainders. *Coatings*, 2023, vol. 13, no. 5, art. 914. doi: 10.3390/coatings13050914.
5. Leao A.A.S., Toledo F.M.B., Oliveira J.F., Carravilla M.A., Alvarez-Valdés R. Irregular packing problems: A review of mathematical models. *EJOR*, 2020, vol. 282, no. 3, pp. 803–822. doi: 10.1016/j.ejor.2019.04.045.
6. do Nascimento D.N., Cherri A.C., Oliveira J.F. The two-dimensional cutting stock problem with usable leftovers: Mathematical modelling and heuristic approaches. *Operational Research*, 2022, vol. 22, pp. 5363–5403. doi: 10.1007/s12351-022-00735-9.
7. Oliveira O., Gamboa D., Silva E. An introduction to the two-dimensional rectangular cutting and packing problem. *ITOR*, 2022, vol. 30, no. 6, pp. 3238–3266. doi: 10.1111/itor.13236.
8. Iori M., de Lima V.L., Martello S., Miyazawa F.K., Monaci M. Exact solution techniques for two-dimensional cutting and packing. *EJOR*, 2021, vol. 289, no. 2, pp. 399–415. doi: 10.1016/j.ejor.2020.06.050.
9. Romanova T., Stoyan Y., Pankratov A. et al. Optimal layout of ellipses and its application for additive manufacturing. *IJPR*, 2021, vol. 59, no. 2, pp. 560–575. doi: 10.1080/00207543.2019.1697836.
10. Guerriero F., Saccomanno F.P. A hierarchical hyper-heuristic for the bin packing problem. *Soft Computing*, 2023, vol. 27, pp. 12997–13010. doi: 10.1007/s00500-022-07118-4.
11. Hu X., Li J., Cui J. Greedy adaptive search: A new approach for large-scale irregular packing problems in the fabric industry. *IEEE Access*, 2020, vol. 8, pp. 91476–91487. doi: 10.1109/ACCESS.2020.2994635.
12. Huang S., Wang Z., Zhou J., Lu J. Planning irregular object packing via hierarchical reinforcement learning. *IEEE RA-L*, 2023, vol. 8, no. 1, pp. 81–88. doi: 10.1109/LRA.2022.3222996.
13. Tu C., Bai R., Aickelin U., Zhang Y., Du H. A deep reinforcement learning hyper-heuristic with feature fusion for online packing problems. *Expert Systems with Applications*, 2023, vol. 230, art. 120568. doi: 10.1016/j.eswa.2023.120568.
14. Pop P.C., Cosma O., Sabo C., Sitar C.P. A comprehensive survey on the generalized traveling salesman problem. *EJOR*, 2023, vol. 314, no. 3, pp. 819–835. doi: 10.1016/j.ejor.2023.07.022.
15. Chentsov A.G., Chentsov P.A. To the application of two-stage dynamic programming in the problem of sequential visiting of megalopolises. *Procedia Structural Integrity*, 2022, vol. 40, pp. 105–111. doi: 10.1016/j.prostr.2022.04.013.
16. Junior B.A., de Carvalho G.N., Santos M.C., Pinheiro P.R., Celedonio J.W.L. Evolutionary algorithms for optimization sequence of cut in the laser cutting path problem. *Applied Sci.*, 2023, vol. 13, no. 18, art. 10133. doi: 10.3390/app131810133.
17. Tavaeva A., Petunin A., Ukolov S., Krotov V. A cost minimizing at laser cutting of sheet parts on CNC machines. In: *CCIS. Proc. MOTOR*, 2019, vol. 1090, pp. 422–437. doi: 10.1007/978-3-030-33394-2\_33.
18. Петунин А.А., Таваева А.Ф. Раскрой и маршрутизация (РиМ): Свид. о регистр. ПрЭВМ № 2024615623. Рос. Федерация, 2024.

## Optimization of technological processes for nesting parts: Using a database of sheet cutting cost parameters

Anastasia F. Tavaeva<sup>1,2</sup>✉, Aleksandr A. Petunin<sup>2,3</sup>

<sup>1</sup> Ural Optical and Mechanical Plant, Ekaterinburg, 620100, Russian Federation

<sup>2</sup> Ural Federal University, Ekaterinburg, 620002, Russian Federation

<sup>3</sup> IMM UB RAS, Ekaterinburg, 620108, Russian Federation

### For citation

Tavaeva, A.F., Petunin, A.A. (2025) 'Optimization of technological processes for nesting parts: Using a database of sheet cutting cost parameters', *Software & Systems*, 38(1), pp. 150–156 (in Russ.). doi: 10.15827/0236-235X.149.150-156

### Article info

Received: 18.06.2024

After revision: 21.07.2024

Accepted: 27.07.2024

**Abstract.** The paper considers an integrated nesting and routing task. It combines the problems of nesting optimization and cost minimization of the cutting process with numerical control (CNC). To solve the problem and develop appropriate optimization algorithms, it is necessary to obtain scientifically substantiated data. Specifically, the data on the values of the cost parameters of a sheet cutting process on the CNC technological equipment for different grades and thicknesses of processed materials. The paper describes a database of such cost process parameters. There is also its diagram, basic structure and organization. The authors chose a relational model to store the data on cost parameters. It consists of 8 tables and contains information on grades and thicknesses of materials processed, cost and density values of materials, cost per unit of cutting tool path at working and idle speeds, as well as one piercing point. Usability of working with data in the database (visualization, adding new records, deleting records, calculating cost parameters for new material grades and thicknesses, changing parameter values) is provided by the designed software in Python. The authors propose to use it either as a separate product or together with the software for automatic design of control programs for CNC sheet cutting equipment to solve practical problems. The authors show this work significance on a model example of designing nesting and tool routing for one type of a laser cutting machine.

**Keywords:** sheet nesting optimization, routing optimization, integrated nesting and routing problem, database, cost parameters

**Acknowledgements.** The work was perf financially supported by the Russian Science Foundation (grand no. 23-21-00343)

### References

1. Petunin, A.A., Kotel, N.S., Tavaeva, A.F. (2023) 'About one optimal solution example to the integrated 2D nesting and routing problem for CNC sheet cutting machines', *Yugra State University Bull.*, (4), pp. 88–101 (in Russ.). doi: 10.18822/byusu20230488-101.
2. Hu, Q., Lin, Z., Fu, J., Luan, C. (2023) 'Optimizing cutting sequences and paths for common-edge nested parts', *Computer-Aided Design*, 164, art. 103610. doi: 10.1016/j.cad.2023.103610.
3. Oliveira, L.T., Silva, E.F., Oliveira, J.F., Toledo, F.M.B. (2020) 'Integrating irregular strip packing and cutting path determination problems: A discrete exact approach', *CAIE*, 149, art. 106757. doi: 10.1016/j.cie.2020.106757.
4. Li, L., Wu, Z., Zhang, Z., Zhang, Y. (2023) 'An optimization method for CNC laser combination cutting of irregular plate remainders', *Coatings*, 13(5), art. 914. doi: 10.3390/coatings13050914.
5. Leao, A.A.S., Toledo, F.M.B., Oliveira, J.F., Carravilla, M.A., Alvarez-Valdés, R. (2020) 'Irregular packing problems: A review of mathematical models', *EJOR*, 282(3), pp. 803–822. doi: 10.1016/j.ejor.2019.04.045.
6. do Nascimento, D.N., Cherri, A.C., Oliveira, J.F. (2022) 'The two-dimensional cutting stock problem with usable leftovers: Mathematical modelling and heuristic approaches', *Operational Research*, 22, pp. 5363–5403. doi: 10.1007/s12351-022-00735-9.
7. Oliveira, O., Gamboa, D., Silva, E. (2022) 'An introduction to the two-dimensional rectangular cutting and packing problem', *ITOR*, 30(6), pp. 3238–3266. doi: 10.1111/itor.13236.
8. Iori, M., de Lima, V.L., Martello, S., Miyazawa, F.K., Monaci, M. (2021) 'Exact solution techniques for two-dimensional cutting and packing', *EJOR*, 289(2), pp. 399–415. doi: 10.1016/j.ejor.2020.06.050.
9. Romanova, T., Stoyan, Y., Pankratov, A. et al. (2021) 'Optimal layout of ellipses and its application for additive manufacturing', *IJPR*, 59(2), pp. 560–575. doi: 10.1080/00207543.2019.1697836.
10. Guerriero, F., Saccomanno, F.P. (2023) 'A hierarchical hyper-heuristic for the bin packing problem', *Soft Computing*, 27, pp. 12997–13010. doi: 10.1007/s00500-022-07118-4.
11. Hu, X., Li, J., Cui, J. (2020) 'Greedy adaptive search: A new approach for large-scale irregular packing problems in the fabric industry', *IEEE Access*, 8, pp. 91476–91487. doi: 10.1109/ACCESS.2020.2994635.
12. Huang, S., Wang, Z., Zhou, J., Lu, J. (2023) 'Planning irregular object packing via hierarchical reinforcement learning', *IEEE RA-L*, 8(1), pp. 81–88. doi: 10.1109/LRA.2022.3222996.
13. Tu, C., Bai, R., Aickelin, U., Zhang, Y., Du, H. (2023) 'A deep reinforcement learning hyper-heuristic with feature fusion for online packing problems', *Expert Systems with Applications*, 230, art. 120568. doi: 10.1016/j.eswa.2023.120568
14. Pop, P.C., Cosma, O., Sabo, C., Sitar, C.P. (2023) 'A comprehensive survey on the generalized traveling salesman problem', *EJOR*, 314(3), pp. 819–835. doi: 10.1016/j.ejor.2023.07.022.

15. Chentsov, A.G., Chentsov, P.A. (2022) 'To the application of two-stage dynamic programming in the problem of sequential visiting of megalopolises', *Procedia Structural Integrity*, 40, pp. 105–111. doi: 10.1016/j.prostr.2022.04.013.
16. Junior, B.A., de Carvalho, G.N., Santos, M.C., Pinheiro, P.R., Celedonio, J.W.L. (2023) 'Evolutionary algorithms for optimization sequence of cut in the laser cutting path problem', *Applied Sci.*, 13(18), art. 10133. doi: 10.3390/app131810133.
17. Tavaeva, A., Petunin, A., Ukolov, S., Krotov, V. (2019) 'A cost minimizing at laser cutting of sheet parts on CNC machines', in *CCIS. Proc. MOTOR*, 1090, pp. 422–437. doi: 10.1007/978-3-030-33394-2\_33.
18. Petunin, A.A., Tavaeva, A.F. (2024) *Nesting & Routing (N&R)*, Pat. RF, № 2024615623.

**Авторы**

**Таваева Анастасия Фидагилевна** <sup>1,2</sup>,  
к.т.н., главный специалист,  
научный сотрудник, a.f.tavaeva@urfu.ru  
**Петунин Александр Александрович** <sup>2,3</sup>,  
д.т.н., профессор, ведущий научный сотрудник,  
a.a.petunin@urfu.ru

<sup>1</sup> Уральский оптико-механический завод,  
г. Екатеринбург, 620100, Россия

<sup>2</sup> Уральский федеральный университет,  
г. Екатеринбург, 620002, Россия

<sup>3</sup> Институт математики и механики  
им. Н.Н. Красовского Уральского отделения РАН,  
г. Екатеринбург, 620108, Россия

**Authors**

**Anastasia F. Tavaeva** <sup>1,2</sup>, Cand. of Sci. (Engineering),  
Chief Engineer, Research Associate,  
a.f.tavaeva@urfu.ru  
**Aleksandr A. Petunin** <sup>2,3</sup>, Dr.Sci. (Engineering),  
Professor, Leading Researcher,  
a.a.petunin@urfu.ru

<sup>1</sup> Ural Optical and Mechanical Plant,  
Ekaterinburg, 620100, Russian Federation

<sup>2</sup> Ural Federal University, Ekaterinburg,  
620002, Russian Federation

<sup>3</sup> IMM UB RAS,  
Ekaterinburg, 620108,  
Russian Federation

## Улучшение эргономических показателей наголовной индикации в очках дополненной реальности для применения в гражданской авиации

И.И. Грешников<sup>1</sup>, Д.А. Давыдов<sup>1</sup>, Б.И. Гончар<sup>1</sup>,  
А.В. Соколов<sup>1</sup>✉, Д.В. Коновалова<sup>1</sup>

<sup>1</sup> Государственный научно-исследовательский институт авиационных систем,  
г. Москва, 125167, Россия

### Ссылка для цитирования

Грешников И.И., Давыдов Д.А., Гончар Б.И., Соколов А.В., Коновалова Д.В. Улучшение эргономических показателей наголовной индикации в очках дополненной реальности для применения в гражданской авиации // Программные продукты и системы. 2025. Т. 38. № 1. С. 157–165. doi: 10.15827/0236-235X.149.157-165

### Информация о статье

Группа специальностей ВАК: 1.2.2

Поступила в редакцию: 17.05.2024

После доработки: 01.08.2024

Принята к публикации: 13.08.2024

**Аннотация.** В статье проводится анализ существующих решений в области наголовной индикации. Предлагается собственная концепция представления пилотажно-навигационной информации в очках дополненной реальности для применения в гражданской авиации. Основным назначением разработанной индикации является предоставление пилоту необходимой пилотажно-навигационной информации на целевом этапе полета. Метод исследования заключался в анализе существующих решений, выявлении их недостатков и в корректировке разработанной индикации на основе рекомендаций опытных пилотов. В процессе работы были определены основные особенности представления пилотажно-навигационной информации, способствующие улучшению эргономических показателей: использование независимых слоев графических объектов пилотажно-навигационной информации, изменение компоновки изображения в зависимости от фазы полета, использование нескольких цветов, возможность голосового управления для отображения нужного слоя индикации. В результате реализовано программное обеспечение, позволяющее проецировать графические изображения мнемосимволов пилотажно-навигационной информации на имитируемую средствами среды Unity закабинную обстановку. Данные объекты индикации размещены с привязкой к следующим параметрам или к их комбинации: положение очков, вектор путевой скорости, строительная ось самолета, закабинное пространство. Разработанное программное обеспечение позволяет также добавлять графические слои от систем синтезированного и улучшенного видения. В статье представлен демонстратор наголовной индикации, созданный на основе данной концепции. Рассмотрены архитектура и функциональность демонстратора, включая видимые пилотом изображения в очках дополненной реальности. Практическая значимость данной работы заключается в улучшении эргономических показателей наголовной индикации по сравнению с существующими вариантами представления пилотажно-навигационной информации.

**Ключевые слова:** наголовная индикация, информационно-управляющее поле, пилотажно-навигационная информация, очки дополненной реальности, эргономические показатели

**Введение.** Активное развитие компонентов *информационно-управляющего поля* (ИУП) кабины пилотов прежде всего связано с необходимостью повышения уровня безопасности полетов. Ограниченная видимость при пилотировании воздушного судна на малых высотах и при рулении по взлетно-посадочной полосе остается одним из самых весомых факторов, влияющих как на безопасность, так и на эффективность воздушных перевозок.

Для решения данной проблемы разрабатываются средства отображения летному экипажу самолета гражданской авиации оптимального объема информации, необходимой для эффективного управления транспортным средством: используются голографические отображения на лобовом стекле, появляются гибкие прозрачные экраны – пленки, увеличиваются и улучшаются характеристики жидкокристал-

лических экранов, применяются сенсорные экраны и т.д. [1, 2].

Вместе с тем при существующем разнообразии средств и способов выдачи информации пилоту возникает проблема загруженности летного экипажа, которая заключается в большом количестве одновременно поступающей информации, что обуславливает актуальность задачи разработки оптимальной индикации. В данной работе рассматривается возможность использования очков дополненной реальности для оценки и повышения эргономических показателей информационного (а в потенциале и информационно-управляющего) поля кабины пилотов.

Целью настоящего исследования является разработка концепции пилотажно-навигационной индикации в очках дополненной реальности для применения на самолетах транспорт-

ной категории, которая имеет свою специфику, связанную со снижением нагрузки на пилотов и предоставлением своевременной информации для поддержания ситуационной осведомленности. Также важным компонентом подобной индикации должна быть система технического зрения, обеспечивающая возможность пилотирования и навигации в условиях низкой видимости.

Научная новизна работы заключается в новом способе отображения необходимой *пилотажно-навигационной информации* (ПНИ) для пилотов. Представленная в исследовании концепция отображения пилотажно-навигационной индикации сочетает в себе элементы концепции отображения ПНИ на индикаторах на лобовом стекле, используемых на многих современных воздушных судах, а также отображение слоев улучшенного и синтезированного видения. В настоящее время отображение данных слоев ПНИ реализовано в основном на пилотажных дисплеях. Данный способ позволяет снизить информационную нагрузку и повысить качество восприятия комплексной информации пилотами.

#### **Анализ существующих систем на основе дополненной реальности**

В настоящий момент сфера технологий дополненной реальности AR (*Augmented Reality*) вызывает интерес у авиакомпаний, технологии дополненной реальности используются для обучения экипажей, технического обслуживания самолетов и развлечения пассажиров [3, 4]. Актуальной тенденцией является также использование очков дополненной реальности в качестве способа отображения пилоту ПНИ [5–7]. Подобные разработки есть у компаний «Aero Glass» и «Thales».

Компания «Microsoft», используя очки дополненной реальности собственного производства, заключила крупный контракт с военными на поставку системы IVAS (*Integrated Visual Augmentation System*), реализованной на базе данных очков. Система оснащена датчиками ночного видения и тепловизорами высокого разрешения. Основной задачей комплекта считается повышение ситуационной осведомленности пехотинцев. При поездке на бронетехнике бойцы могут получать сигнал с ее внешних камер и следить за окружающей обстановкой. Это упрощает своевременное обнаружение угроз для пехоты или техники, а также обеспечивает безопасную высадку.

Проект Brilliant Eyes, разработанный компанией «Aerocross Systems», ориентирован на рынок самолетов малой авиации. В нем используются компактные очки дополненной реальности, на которые выводится традиционный набор информации: высота, курс, пространственное положение, приборная и вертикальная скорость. Чтобы обеспечить соответствие представления данных положению головы пилота во время полета, в настоящее время в рамках проекта ведется разработка системы отслеживания положения головы.

Шлем «Глаз Бога» (<http://www.swsys.ru/uploaded/image/2025-1/16.jpg>) в режиме дополненной реальности позволяет видеть пилоту сквозь обшивку самолета за счет поступления информации с системы камер, расположенных на фюзеляже. При этом ему транслируется цельное видео без задержек. Например, пилот может видеть не кресло, а горы под ногами с высокой степенью детализации. Имеется возможность приближать и отдалять изображение. В шлеме также есть режим ночного видения, реализованный за счет использования инфракрасных камер. Кроме того, бортовой компьютер умеет различать своих и чужих. Когда система видит врага, она определяет его уровень опасности, предлагая использование наиболее подходящего вооружения. Чтобы уничтожить врага, пилоту нужно только подать голосовую команду. Помимо этого, все высокоточное целеуказание бортовых орудий завязано на движении головы и глаз летчика.

Объединенная авиастроительная корпорация приступила к испытаниям шлема для летчиков со встроенной системой индикации. Такой шлем, оснащенный системой дополненной реальности, должен значительно упростить управление самолетом и улучшить ситуационную осведомленность летчика. Шлем заменяет сразу несколько приборов и оснащен большим количеством датчиков. Они улавливают движения головы пилота и в зависимости от ее поворота проецируют на очки информацию, необходимую в данный момент. Пилоту остается только нажимать нужные кнопки на рукоятке. Прототип оснащен функцией проецирования картинки с полем боя на очки шлема. Также планируется использование технологии дополненной реальности, позволяющей выводить на стекло шлема информацию, которую летчик не может получить визуально. Основные функции нашлемной системы видения – отображение в поле зрения пилота видеoinформации от каналов технического зрения

для обеспечения пилотирования, поиска, обнаружения и распознавания объектов на местности, посадки на необорудованные и неосвещенные площадки в простых и ограниченно сложных метеорологических условиях днем и ночью.

Рассмотренные проекты и технологии, ориентированные прежде всего на военную и малую авиацию, являются многообещающими и потенциально могут значительно увеличить эффективность человеко-машинного взаимодействия.

Предлагаемая концепция создана с учетом современных технологий разработки авиационной индикации [8] и с применением целеориентированного проектирования [9, 10].

В рамках исследования ставились следующие задачи:

- разработка концепции индикации в очках дополненной реальности, учитывающей особенности полетов транспортной и пассажирской авиации;
- разработка программной архитектуры и реализации концепции в очках дополненной реальности Microsoft HoloLens 2;
- интеграция очков в универсальный стенд прототипирования кабины пилотов и проведение испытаний с участием летных экспертов.

### Концепция отображения ПНИ в очках дополненной реальности

Предполагается отображение индикации изображения пространственного положения самолета и основной пилотажной информации, навигационной информации, синтезированного изображения рельефа местности, элементов аэродромной инфраструктуры на основе

БД, а также других вспомогательных символов, изображения от системы улучшенного видения на основе информации, полученной от телевизионной и инфракрасных камер.

Все пилотажные и навигационные объекты и мнемосимволы, отображаемые в очках дополненной реальности, условно разделены на несколько категорий: объекты, привязанные к положению очков, к вектору путевой скорости FPV (*Flight Path Vector*), к строительной оси самолета, к закабинному пространству, а также объекты, одновременно привязанные к FPV и к положению очков.

Объекты, привязанные к положению очков, отображаются так, чтобы геометрический центр прямоугольника, который они составляют, совпадал с геометрическим центром прямоугольника поля зрения очков, а угловое положение изображения было привязано к полю зрения очков (рис. 1). К данным объектам относится важная информация, необходимая для регулярного контроля, такая как барометрическая высота, приборная скорость, угол атаки и т.п.

Объекты, привязанные к закабинному пространству, накладываются на соответствующие им объекты реального мира или рядом с ними (взлетно-посадочная полоса, интрузеры и т.д.). Примером таких объектов могут служить навигационные символы, так называемая дорога в небе, которые визуализируют заданную траекторию движения, отображаемую в соответствии с маршрутом, предоставленным системой самолетовождения, или с данными от курсоглиссадной системы. Заданная траектория движения обозначается набором прямоугольных символов, расположенных вдоль траектории движения с определенным шагом.

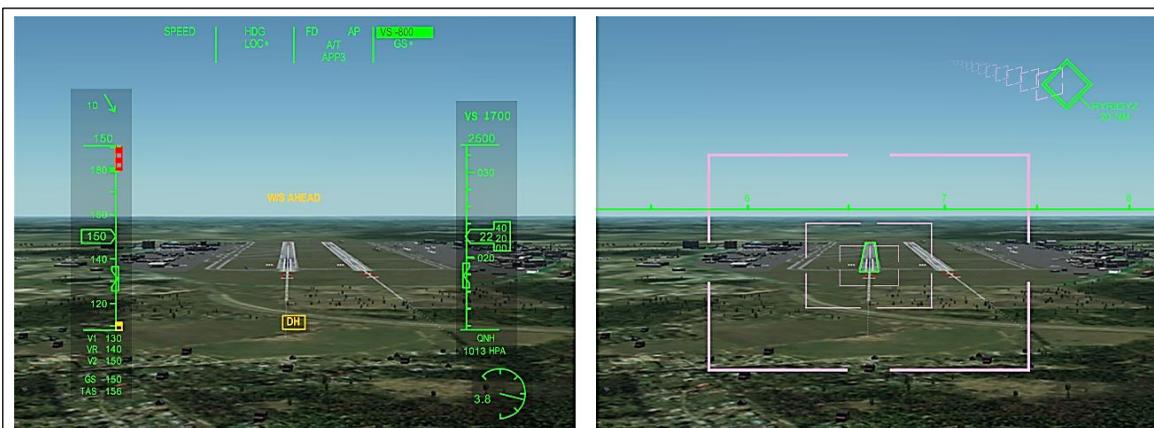


Рис. 1. Объекты, привязанные к положению очков (слева) и к закабинному пространству (справа)

Fig. 1. Objects linked to the glasses position (left) and to the cockpit external space (right)

Отдельно стоит выделить объекты, привязанные к строительной оси воздушного судна и к FPV (рис. 2). FPV является важным мнемосимволом, указывающим фактическое направление движения самолета, которое практически никогда не совпадает со строительной осью самолета. Он особенно важен на этапе посадки. При боковом ветре именно этот символ совместно с планками отклонения помогает произвести посадку, поэтому вместе с ним группируются и другие важные для фазы посадки мнемосимволы – указатель заданного направления, индекс превышения заданной скорости, радиовысота и т.п.

Горизонтальное положение объектов, привязанных к строительной оси самолета, определяется таким образом, чтобы вертикальная плоскость, проходящая через продольную ось самолета, проходила через вертикальную ось прямоугольника, который составляют данные объекты. Вертикальное положение символа самолета привязано к его строительной оси, а угловое положение всего изображения – к кабине пилотов.

Также имеются объекты, одновременно привязанные к FPV и к положению очков. Если FPV выходит за пределы (<http://www.swsys.ru/uploaded/image/2025-1/17.jpg>), то все объекты, связанные с FPV, помещаются в специальный круг. Центр этого круга всегда лежит на линии, соединяющей истинное положение FPV и геометрический центр прямоугольника поля зрения очков, причем круг непременно соприкасается своей границей с одной или двумя сторонами красного прямоугольника. Угловое положение

изображения вычисляется так же, как и для объектов, привязанных к FPV. Следует отметить, что выход FPV за обозначенные границы может происходить в результате как естественных причин (например, вследствие воздействия ветра), так и поворота головы.

Важной составляющей концепции является изменение компоновки графических объектов в зависимости от фазы полета. В результате проведенных исследований было разработано пять вариантов компоновки для следующих фаз полета: от гейта до отрыва, от отрыва до посадки не ниже 1 500 ft, от 1 500 ft до 100 ft, от 100 ft до касания, от касания до гейта.

В зависимости от фазы полета перед экипажем воздушного судна встают различные задачи. Если при относительно спокойном полете на эшелоне задача сводится к периодическому контролю пространственного положения и корректности следования по маршруту, то на этапе посадки когнитивная нагрузка возрастает и все внимание экипажа сосредотачивается на пилотировании, в том числе в поддержании посадочной скорости, глиссады, пространственного положения самолета, выравнивания, касания и пробега.

Данное обстоятельство обуславливает необходимость динамически изменять компоновку и наполнение кадра индикатора на лобовом стекле. На менее интенсивных этапах полета наполненность кадра максимальная, то есть пилоту дается вся возможная информация. На более интенсивных этапах на индикаторе остается лишь самая необходимая информация, не превосходящая возможный объем восприятия пилота на данном этапе и обеспечиваю-

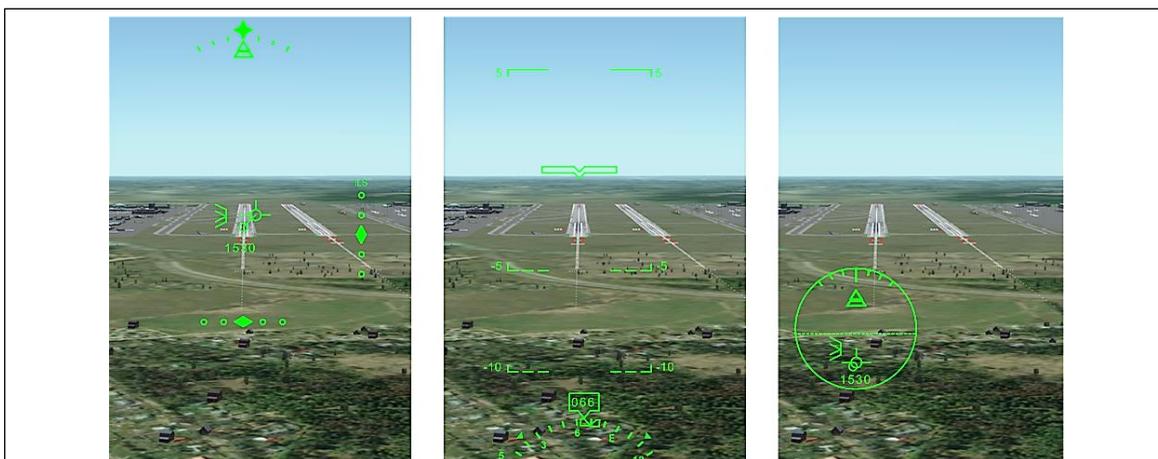


Рис. 2. Объекты, привязанные к FPV (слева), к строительной оси самолета (по центру) и одновременно привязанные к FPV и к положению очков (справа)

Fig. 2. Objects linked to FPV (left), to the airplane construction axis (center) and simultaneously linked to FPV and to the glasses position (right)

щая наиболее безопасное выполнение процедур данного этапа.

Разбег и следующий за ним взлет – один из наиболее ответственных этапов полета. Цена ошибки на данном этапе крайне высока. После скорости VI на некоторых самолетах блокируется даже информация о пожаре двигателя, так как пилот все равно вынужден взлетать и лишний стресс в этой ситуации ему ни к чему. Следовательно, информационная насыщенность индикатора на лобовом стекле на данном этапе минимальная.

Во время крейсерского полета пилотирование, как правило, переключается на автопилот, поэтому активных действий от пилота на данном этапе не требуется. В целом отпадает и необходимость в индикаторе на лобовом стекле. Если экипаж все же имеет желание использовать такой индикатор, то для этого предусмотрен специальный режим отображения всей нужной для полета на эшелоне информации.

Во время захода на посадку наиболее значимы выдерживание заданной глиссады, а также своевременный визуальный контакт с полосой (за исключением посадки по категории III). Скорость выставляется на автопилоте, пилот контролирует ее значение лишь в процессе захода. Динамика изменения высоты также теряет значение при точном заходе или при ручном заходе с включенным режимом синтезированного/улучшенного видения. Вследствие этого шкалы приборной скорости и баровысоты не индицируются, отображаются только цифровые значения данных параметров.

После отметки в 100 футов до касания взлетно-посадочной полосы остается не более 10 секунд. На таком коротком интервале самолет не может значительно изменить приборную или вертикальную скорость, поэтому главное – выравнивание самолета и своевременное увеличение угла тангажа в момент касания взлетно-посадочной полосы. Для этого пилоту требуется минимальное количество информации: радиовысота, шкала тангажа, FPV, авиагоризонт, синтезированное/улучшенное изображение полосы.

После касания взлетно-посадочной полосы наступает фаза пробег. От пилота требуются удерживание самолета по центру полосы (что в условиях плохой видимости может выполняться при помощи индекса горизонтальной девиации), а также успешное торможение, степень эффективности которого может быть оценена при помощи шкалы воздушной скорости.

Компоновка кадра на данном этапе будет следующей: шкала воздушной скорости, шкала и индекс отклонения по горизонтали, авиагоризонт.

Не стоит недооценивать важность выбора цветовой палитры, так как от этого зависят своевременность и качество восприятия информации. Использование исключительно зеленого цвета на традиционных индикаторах на лобовом стекле связано с особенностями технической реализации данных устройств [11]. В очках дополненной реальности подобное ограничение отсутствует, поэтому возможны применение любых цветовых оттенков и создание кадров, аналогичных по цветовой насыщенности кадрам на дисплеях приборной доски. Однако к использованию насыщенной цветовой палитры в очках дополненной реальности следует относиться осторожно. В результате ряда экспериментов было установлено, что большое количество цветов может приводить к сложности распознавания реальных объектов за кабиной обстановки и информации с приборной доски, поэтому наиболее целесообразно сохранять традиционный цвет, используемый на индикаторах на лобовом стекле, – зеленый, за исключением индикации опасных режимов, а также индикации, на которую требуется немедленная реакция со стороны пилота (например, достижение высоты принятия решения).

В ситуации, когда видимость за кабиной не позволяет четко различать рельеф окружающей местности и объекты наземной инфраструктуры, может быть включен режим синтезированного видения. При этом из БД берется соответствующая информация о рельефе окружающей местности и объектах наземной инфраструктуры и с использованием координат и направления движения воздушного судна проецируется на реальный мир. Изображение проецируется во всех направлениях, таким образом, поворачивая голову, пилот может обозревать пространство вокруг себя. Например, при посадке с сильным боковым ветром, немного повернув голову в сторону, пилот может наблюдать взлетно-посадочную полосу как бы сквозь обшивку самолета.

Режим синтезированного видения имеет определенные недостатки:

- БД может быть устаревшей и не охватывать недавно возведенные объекты;
- не могут быть учтены объекты, внезапно оказавшиеся на взлетно-посадочной полосе;
- возможна погрешность в вычислениях текущих координат и направления движения воздушного судна.

В силу данных причин вместе с режимом синтезированного видения рекомендуется применять режим улучшенного видения. Поскольку изображение улучшенного видения формируется на основе данных с телевизионной и инфракрасной камер, направленных вдоль продольной оси самолета, область отображения улучшенного видения ограничена областью охвата данных камер.

### Демонстратор ИУП кабины воздушного судна гражданской авиации с наголовной индикацией

Для обработки комплексной ПНИ был использован специальный демонстратор ИУП кабины воздушного судна гражданской авиации (рис. 3). Архитектура демонстратора представлена на рисунке 4.

В контексте данного исследования разработана оригинальная программная оболочка на языке высокого уровня, позволяющая проецировать графические изображения мнемосимволов пилотажно-навигационной индикации на имитируемую средствами среды Unity закабинную обстановку, а также добавлять графические слои от систем синтезированного и улучшенного видения [12].

Программная реализация элементов ПНИ представляет собой подготовленные в Adobe Illustrator изображения, наложенные в среде Unity на плоскость. Элементы индикации, визуализация которых проблематична или невозможна в плоскостном виде (например, шкала тангажа или линия горизонта), реализованы в виде 3D-моделей, выполненных в среде Blender. Для отображения динамики и функциони-

рования индикации в целом написана кодовая база на языках программирования C# и C.

Изображение аэродромной инфраструктуры на слое синтезированного видения формируется с помощью алгоритмов на языке программирования C# из специализированных БД. В случае имитации рельефа местности используются карты высот в виде набора изображений с цветами пикселей от белого к черному (где белый цвет – минимальное значение высоты рельефа, черный – максимальное), также реализован альтернативный вариант с цветной градацией. Для создания итогового изображения ландшафта в очках используются шейдеры – набор инструкций, сообщающих очкам, каким образом формировать изображение.

Для моделирования работы системы улучшенного видения используется имитатор изображения с инфракрасной наружной камеры самолета. В целях построения изображения в очках дополненной реальности применяются алгоритмы, которые преобразуют массив информации, передающийся из имитатора трансляции наружной камеры, в проецируемое на плоскость изображение.

На рисунке 5 показан вариант индикации, совмещающий слои ПНИ и синтезированного видения на этапе руления. Как можно заметить, количество пилотажных мнемосимволов на этом этапе существенно сокращено, оставлены только мнемосимволы, необходимые для выполнения руления и торможения.

Для управления индикацией слоев (включения/выключения каждого из них), а также регулировки их прозрачности разработано специальное ПО, в том числе реализовано голосовое управление слоями, что существенно упрощает

оперативную настройку индикации. В качестве наголовного устройства использовались очки дополненной реальности Microsoft HoloLens 2, имеющие достаточно большое поле обзора для нужд прототипирования наголовной пилотажно-навигационной индикации.

### Заключение

На основе проведенного анализа имеющихся решений в области наголовной индикации осуществлен синтез демонстратора ИУП каби-



Рис. 3. Демонстратор ИУП кабины воздушного судна гражданской авиации с наголовной индикацией

Fig. 3. Airplane cabin control field stand with head-mounted indication

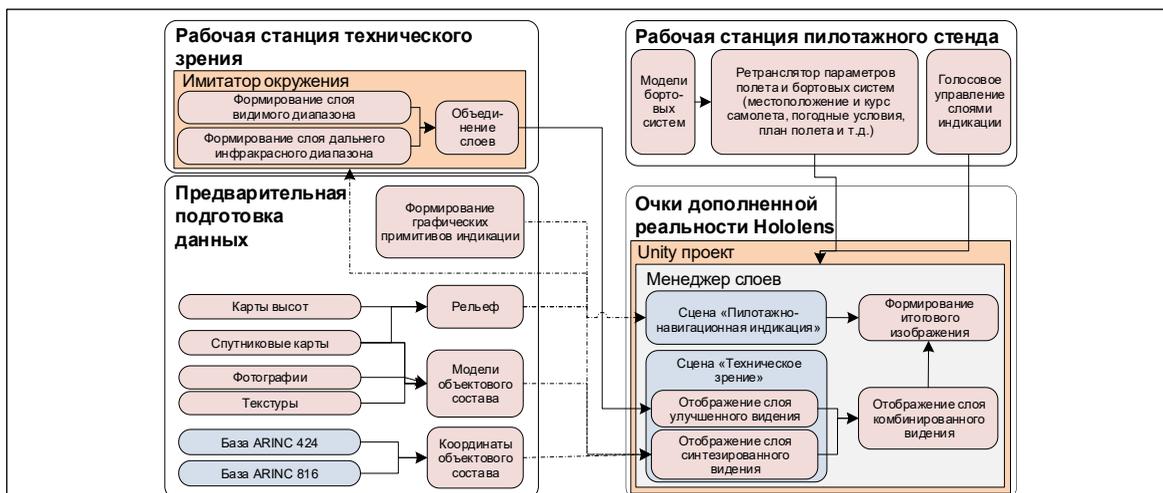


Рис. 4. Архитектура демонстратора ИУП кабины воздушного судна с наголовной индикацией

Fig. 4. Architecture of airplane cabin control field stand with head-mounted indication

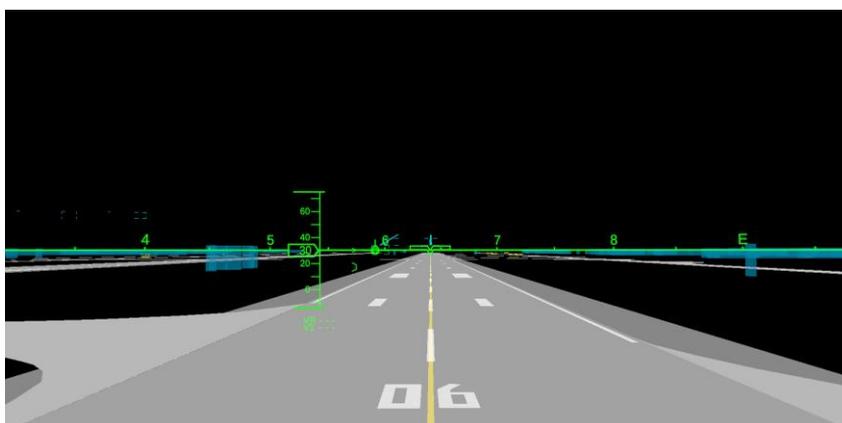


Рис. 5. Руление по взлетно-посадочной полосе

Fig. 5. Taxiing on a runway

ны перспективного самолета гражданской авиации. Данный демонстратор апробирован с участием пилотов-экспертов и специалистов по сертификации и оценен положительно. Научно-технический задел, полученный в результате работы, может быть использован при прототипировании ИУП кабины пилотов и в составе бортовых перспективных комплексов самолетов гражданской авиации.

Отмечено, что данная технология существенно снижает нагрузку на экипаж за счет следующих факторов:

- снижается необходимость переводить взгляд из зоны закабинного пространства в зону приборной индикации, так как значительная часть требуемой информации отображается непосредственно в очках;

- применение более одного цвета для формирования изображения (в отличие от тради-

ционного ИЛС) позволяет более четко определять опасные режимы и выходы за эксплуатационные ограничения;

- изменение компоновки в зависимости от фазы полета дает возможность концентрироваться на важной именно для текущего момента информации, что особенно критично на таких интенсивных режимах полета, как взлет и посадка.

Использование наголовной индикации в кабине перспективных воздушных судов реализует потенциальные преимущества для производителей самолетов, авиакомпаний и пилотов, связанные с оптимизацией ИУП кабины, поэтому целесообразно продолжать развитие и внедрение наголовной индикации в составе перспективных ИУП с учетом научно-технического задела, полученного в рамках данной работы.

## Список литературы

1. Грешников И.И., Златомрежев В.И. Использование передовых технологий для оптимизации информационно-управляющего поля кабины перспективного самолета // Нейрокомпьютеры и их применение: XVIII Всерос. науч. конф.: тезисы докл. 2020. С. 66–69.
2. Зыбин Е.Ю., Косьянчук В.В., Земкин В.А. Авиационные человеко-машинные интерфейсы: состояние и перспективы развития // Актуальные проблемы психологии труда, инженерной психологии и эргономики. 2021. № 10. С. 211–230. doi: 10.38098/ergo.2021.
3. Зыбин Е.Ю., Леликов М.А., Калужный Ю.С. Особенности и перспективы использования AR-устройств для задач гражданской авиации // HOLOEXPO 2023: тезисы докл. Междунар. конф. С. 166–170.
4. Пашаев А.М., Искендеров И.А., Керимов С.М.М. Модель навигации с дополненной реальностью и алгоритм ее применения в гражданской авиации // Авиакосмическое приборостроение. 2020. № 6. С. 3–13. doi: 10.25791/aviakosmos.06.2020.1160.
5. Сумкин В.В., Коваленко Г.В., Сеножацкий Е.А. Использование систем дополненной реальности в авиации // Проблемы летной эксплуатации и безопасность полетов. 2018. Т. 12. С. 78–91.
6. Муксимова Р.Р., Коваленко Г.В., Муравьев И.С., Сеножацкий Е.А. Использование систем дополненной реальности в авиации с целью повышения безопасности полетов // Актуальные проблемы защиты и безопасности: тр. XXII Всерос. науч.-практич. конф. 2019. С. 234–240.
7. Зорин А.О., Янушевский А.С., Коваленко Г.В. Использование дополненной реальности в гражданской авиации // Современные научные исследования: актуальные вопросы, достижения и инновации: сб. статей конф. 2022. С. 59–65.
8. Глухова Э.Д., Глухов М.А., Кривоноженков В.А., Целиков А.С. Технология разработки вариантов индикационных форматов, обеспечивающих решение функциональных задач авиационных комплексов // Тр. ГосНИИАС. Вопросы авионики. 2020. № 1. С. 21–26.
9. Корсун О.Н., Глухова Э.Д., Скрябиков Н.В. Применение целеориентированного проектирования при разработке кадров пилотажной и навигационной индикации // УАКС-2022: мат. конф. 2022. С. 32–35.
10. Купер А., Рейман Р., Кронин Д. и др. Интерфейс. Основы проектирования взаимодействия; [пер. с англ.]. СПб: Питер, 2017. 720 с.
11. Бондаренко М.А., Бондаренко А.В., Дрынкин В.Н., Павлов Ю.В. Формирование и отображение видеoinформации в бортовых мультиспектральных системах улучшенного видения // Вестн. МГТУ им. Н.Э. Баумана. Сер. Приборостроение. 2022. № 2. С. 79–95. doi: 10.18698/0236-3933-2022-2-79-95.
12. Грешников И.И., Давыдов Д.А., Гончар Б.И. Программное обеспечение «Пилотажно-навигационная индикация в очках дополненной реальности»: Свид. о регистр. ПрЭВМ № 2023664504. Рос. Федерация, 2023.

Software &amp; Systems

doi: 10.15827/0236-235X.149.157-165

2025, 38(1), pp. 157–165

### Improving ergonomic parameters of head-mounted display in augmented reality glasses for civil aviation

Ivan I. Greshnikov<sup>1</sup>, Dmitry A. Davydov<sup>1</sup>, Bogdan I. Gonchar<sup>1</sup>,  
Andrey V. Sokolov<sup>1</sup>✉, Darya V. Konvalova<sup>1</sup>

<sup>1</sup> State Research Institute of Aviation Systems (GosNIIAS),  
Moscow, 125167, Russian Federation

#### For citation

Greshnikov, I.I., Davydov, D.A., Gonchar, B.I., Sokolov, A.V., Konvalova, D.V. (2025) 'Improving ergonomic parameters of head-mounted display in augmented reality glasses for civil aviation', *Software & Systems*, 38(1), pp. 157–165 (in Russ.). doi: 10.15827/0236-235X.149.157-165

#### Article info

Received: 17.05.2024

After revision: 01.08.2024

Accepted: 13.08.2024

**Abstract.** The paper analyzes existing solutions in the field of head-mounted display. The authors offer their own concept of pilot and navigation information presentation in augmented reality glasses for applying in civil aviation. The main purpose of the developed display is to provide the pilot with the necessary flight and navigation information at the target flight phase. The research method consists in analyzing the existing solutions, identifying their shortcomings and adjusting the developed indication based on the recommendations of experienced pilots. The authors have identified the main features of flight and navigation information presentation that contribute to the ergonomic performance improvement. For example, using independent graphical layers of flight and navigation information objects, changing image layout depending on

a flight phase, using several colors, voice control to display the required display layer. Consequently, the authors presented the software that allows projecting graphic images of flight and navigation information mnemonic symbols into the cockpit external space simulated in Unity environment. These display objects are placed according to the following parameters or to their combination: position of the glasses, path velocity vector, airplane construction axis, cockpit external space. The developed software also allows adding graphic layers from synthesized and enhanced vision systems. The paper presents a head-mounted display demonstrator based on this concept. The authors reviewed the architecture and functionality of the demonstrator including pilot-visible images in augmented reality glasses. The practical significance of this work is in the improvement of head-mounted display ergonomic parameters in comparison with the existing display variants of pilot navigation information.

**Keywords:** head-mounted display, flight and navigation information, augmented reality glasses, ergonomic performance

### References

1. Greshnikov, I.I., Zlatomrezhev, V.I. (2020) 'The use of advanced technologies to optimize the information and control field of the cockpit of a promising aircraft', *Proc. XVIII All-Russ. Sci. Conf. Neurocomputers and their Application*, pp. 66–69 (in Russ.).
2. Zybin, E.Yu., Kosyanchuk, V.V., Zemkin, V.A. (2021) 'Aviation human-machine interfaces – current state and development trends', *Actual Problems of Labor Psychology, Engineering Psychology and Ergonomics*, pp. 211–230 (in Russ.). doi: 10.38098/ergo.2021.
3. Zybin, E.Yu., Lelikov, M.A., Kalyuzhny, Yu.S. (2023) 'Features and prospects of using AR devices for civil aviation tasks', *Proc. Int. Conf. HOLOEXPO 2023*, pp. 166–170 (in Russ.).
4. Pashayev, A.M., Iskendarov, I.A., Karimov, S.M.M. (2020) 'Augmented reality based navigation model and its application algorithm in civil aviation', *Aerospace Instrument-Making*, (6), pp. 3–13 (in Russ.). doi: 10.25791/aviakosmos.06.2020.1160.
5. Sumkin, V.V., Kovalenko, G.V., Senozhatskiy, E.A. (2018) 'The use of augmented reality systems in aviation', *Problems of Flight Operation and Flight Safety*, 12, pp. 78–91 (in Russ.).
6. Muksimova, R.R., Kovalenko, G.V., Muravyov, I.S., Senozhatskiy, E.A. (2019) 'The use of augmented reality systems in aviation in order to improve flight safety', *Proc. All-Russ. Sci. and Pract. Conf. Actual Problems of Protection and Safety*, pp. 234–240 (in Russ.).
7. Zorin, A.O., Yanushevsky, A.S., Kovalenko, G.V. (2022) 'The use of augmented reality in civil aviation', *Proc. Conf. Modern Scientific Research: Topical Issues, Achievements and Innovations*, pp. 59–65 (in Russ.).
8. Glukhova, E.D., Glukhov, M.A., Krivonozhenkov, V.A., Tselikov, A.S. (2020) 'Technology for the development of options for display formats that provide solutions to the functional tasks of aviation complexes', *Proc. GosNIAS. Avionics Issues*, (1), pp. 21–26 (in Russ.).
9. Korsun, N.V., Glukhova, E.D., Skryabikov, O.N. (2022) 'Goal-directed design in the development of flight control and navigation frames', *Proc. UACS-2022*, pp. 32–35 (in Russ.).
10. Cooper, A., Reimann, R., Cronin, D. et al. (2017) *About Face. The Essentials of Interaction Design*. Wiley Publ., 720 p. (Russ. ed.: St. Petersburg, 720 p.).
11. Bondarenko, M.A., Bondarenko, A.V., Drynkin, V.N., Pavlov, Yu.V. (2022) 'Forming and displaying video data in onboard enhanced vision multispectral systems', *Herald of the Bauman MSTU. Ser. Instrument Engineering*, (2), pp. 79–95 (in Russ.). doi: 10.18698/0236-3933-2022-2-79-95.
12. Greshnikov, I.I., Davydov, D.A., Gonchar, B.I. (2023) *Software "Aerobatic navigation display in augmented reality glasses"*, Pat. RF, № 2023664504.

### Авторы

**Грешников Иван Игоревич**<sup>1</sup>, к.т.н.,  
начальник сектора, vvanes@mail.ru  
**Давыдов Дмитрий Андреевич**<sup>1</sup>,  
ведущий инженер, dadavydov@gosniias.ru  
**Гончар Богдан Игоревич**<sup>1</sup>, инженер,  
bigonchar@gosniias.ru  
**Соколов Андрей Вячеславович**<sup>1</sup>,  
начальник сектора, avsookolov@gosniias.ru  
**Коновалова Дарья Витальевна**<sup>1</sup>,  
техник, dvkonovalova@gosniias.ru

### Authors

**Ivan I. Greshnikov**<sup>1</sup>, Cand. of Sci. (Engineering),  
Head of Sector, vvanes@mail.ru  
**Dmitry A. Davydov**<sup>1</sup>, Leading Engineer,  
dadavydov@gosniias.ru  
**Bogdan I. Gonchar**<sup>1</sup>, Engineer,  
bigonchar@gosniias.ru  
**Andrey V. Sokolov**<sup>1</sup>,  
Head of Sector, avsookolov@gosniias.ru  
**Darya V. Konovalova**<sup>1</sup>, Technical Fellow,  
dvkonovalova@gosniias.ru

<sup>1</sup> Государственный научно-исследовательский институт авиационных систем, г. Москва, 125167, Россия

<sup>1</sup> State Research Institute of Aviation Systems (GosNIAS), Moscow, 125167, Russian Federation

## База данных по процессам микробиологической очистки сточных вод и почвы как эффективный инструмент представления данных и знаний

А.А. Досаев <sup>1</sup>✉, А.С. Скичко <sup>1</sup>, Н.В. Меньшутина <sup>1</sup>

<sup>1</sup> Российский химико-технологический университет им. Д.И. Менделеева, г. Москва, 125047, Россия

### Ссылка для цитирования

Досаев А.А., Скичко А.С., Меньшутина Н.В. База данных по процессам микробиологической очистки сточных вод и почвы как эффективный инструмент представления данных и знаний // Программные продукты и системы. 2025. Т. 38. № 1. С. 166–173. doi: 10.15827/0236-235X.149.166-173

### Информация о статье

Группа специальностей ВАК: 2.3.1

Поступила в редакцию: 09.02.2024

После доработки: 13.06.2024

Принята к публикации: 19.06.2024

**Аннотация.** Объектом представленного исследования является системный анализ процессов микробиологической очистки сточных вод и почвы с последующей разработкой базы данных. Актуальность работы обусловлена проблемой неструктурированности больших объемов поступающей разнородной информации по данному направлению. Проведен анализ литературы с описанием существующих баз данных по микробиологической очистке и с указанием их достоинств и недостатков. Дано обоснование актуальности разработки базы данных, объединяющей все ключевые составляющие процессов микробиологической очистки сточных вод и почвы. Проведен системный анализ предметной области, с использованием системного подхода построена архитектура хранения данных. Отмечены достоинства разработанной системы, показаны примеры выполнения поисковых запросов. Разработанная база данных «Микробиологическая очистка» содержит обширную информацию о загрязняющих веществах и микроорганизмах с описанием процессов микробиологической очистки. Предлагаемая система хранения данных может быть полезна исследователям, областью научных интересов которых являются процессы микробиологической очистки, микробиология, химия и химическая технология. Она позволит уменьшить время на поиск информации при выполнении научно-исследовательских работ.

**Ключевые слова:** системный анализ, системный подход, реляционная модель хранения данных, архитектура базы данных, база данных, биотехнология, окружающая среда, микробиологическая очистка

**Введение.** Проблема загрязнения водных ресурсов и почвенного слоя планеты становится все острее из-за увеличения количества природных зон, загрязненных производственными отходами, потенциально опасными для человека и животного мира [1]. Для решения данной проблемы на смену традиционным (механическим, химическим и физико-химическим) методам очистки приходят более экономически выгодные и экологически безопасные микробиологические, которые являются актуальным направлением в биотехнологии. За последние десятилетия они все чаще находят применение на производственных объектах от лабораторных до промышленных масштабов. Объем исследований и публикуемых научных трудов по поиску, выделению и использованию различных микроорганизмов в процессах микробиологической очистки растет с каждым годом, в результате возникают проблемы из-за неструктурированности и несистематизированности поступающей информации. Вследствие этого возникают сложности как при поиске необходимой имеющейся информации, так и при обработке больших объемов новых данных [2].

Цель данного исследования – разработка БД по методам микробиологической очистки сточных вод и почвы, включающая решение следующих задач: анализ литературных источников по существующим системам хранения данных по процессам очистки с выявлением имеющихся достоинств и недостатков, системный анализ предметной области для построения архитектуры БД, ее дальнейшая программная реализация и заполнение, а также создание алгоритмов поиска в БД для наиболее полного и точного предоставления информации под конкретную задачу. Отличительной особенностью построенной архитектуры хранения данных является включение в нее объекта-связки – главной таблицы «Система очистки». Таблица содержит основную информацию о процессах микробиологической очистки по всем связанным ключевым объектам БД. Помимо основной информации, в нее внесены параметры оптимальных условий процесса и критерий эффективности – достигаемый процент очистки, что сводит к минимуму временные затраты при подготовке к проведению процесса и способствует выбору наиболее оптимальной системы микробиологического очищения. Дополни-

ным достоинством является возможность получения отдельной информации по химическим соединениям и микроорганизмам с соответствующими ферментами и средами культивирования. Данные доступны с помощью форм «Справочник по химическим веществам» и «Справочник по микробиологии» соответственно. Для удобства обновления данных и выполнения поиска информации главное меню БД содержит разделы «Администратор» и «Пользователь».

### Современные системы хранения данных предметной области

Анализ литературных источников показал, что для решения проблемы неструктурированности информации разработано немало БД.

Так, интегрированная онлайн-БД KEGG служит полезным поисковым инструментом, предоставляя информацию из 16 БД о метаболических путях, соединениях, генах и белках. Как ее недостаток отмечена сложность обработки данных вследствие большого объема информации из различных источников [3]. Одной из крупнейших онлайн-БД метаболических путей и ферментов, доступных в настоящее время, является БД MetaCyc [4]. Однако установлено, что эта база содержит множество некорректных данных [5]. Узкоспециализированные базы EAWAG-BBD [6], OxDBase [7], PMBD [8], «Активный ил» [9], Bionemo [10] могут быть использованы лишь в качестве дополнительного источника информации при разработке и изучении методов микробиологической очистки. При анализе российских публикаций выяснилось, что за последние 15 лет не вышло ни одной научной статьи, посвященной разработке и использованию БД по микробиологической очистке.

В результате установлен ряд недостатков современных систем хранения данных по микробиологической очистке: содержание больших объемов разрозненных данных, сложный интерфейс, узкая специализация, отсутствие обновления информации. Таким образом, подтверждена актуальность разработки БД, охватывающей все ключевые составляющие процесса микробиологического очищения как единой системы очистки. Установлена необходимость включения в архитектуру БД показателя эффективности процесса, который будет способствовать выбору наиболее оптимальной системы очистки [11].

### Системный анализ объекта исследования

На первом этапе разработки БД был выполнен системный анализ предметной области – выделены ее основные сущности (объекты) и установлены связи между ними для правильного взаимодействия (построения архитектуры хранения данных) [12]. Как известно, существует несколько подходов применения системного анализа – дедуктивный, индуктивный и системный [13, 14]. Установлено, что наиболее эффективен системный подход, который подразумевает рассмотрение предметной области как целостной системы – совокупности объектов предметной области и связей между ними [15]. В отличие от дедуктивного и индуктивного этот подход подразумевает выделение сущностей непосредственно из предметной области таким образом, чтобы они наиболее точно фиксировали ее динамику и структуру (целостные свойства) [16].

С использованием системного подхода сформировано описание предметной области – выделены семь ключевых сущностей («Загрязнители», «Микроорганизмы», «Продукты», «Система очистки», «Оборудование», «Процесс очистки», «Среда для осуществления процесса очистки») и четыре второстепенные («Среда для культивирования микроорганизмов», «Масштаб оборудования», «Ферменты», «Литературный источник»), наиболее полно и целостно отражающие область микробиологической очистки.

### Описание структуры БД

Помимо выделения сущностей процесса микробиологической очистки, были установлены уникальные элементы – атрибуты, наиболее целостно характеризующие каждую сущность и способствующие наиболее быстрому пониманию процесса очистки как единой функционирующей системы [12]. Также были определены связи между всеми элементами – построена архитектура БД (рис. 1).

Схема представляет собой реляционную модель данных с мощным теоретическим фундаментом, в основе которого лежит теория математических отношений. Модель предполагает строгую классификацию данных и создание двумерных таблиц. Как известно, связи, установленные между этими таблицами, увеличивают эффективность обработки данных и уменьшают их повторяемость [17].

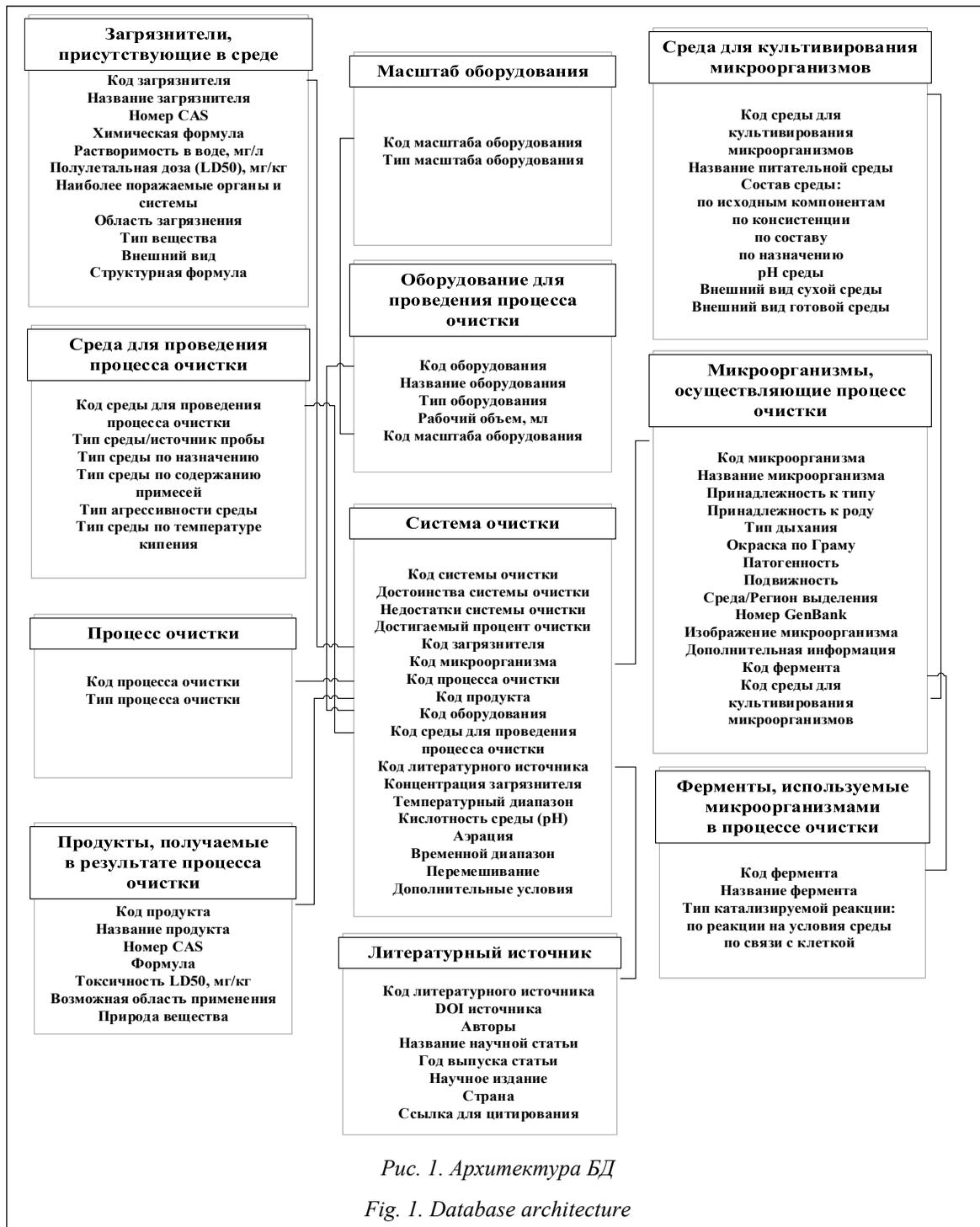


Рис. 1. Архитектура БД

Fig. 1. Database architecture

Особенностью построенной архитектуры хранения данных является объект-связка – главная связующая таблица «Система очистки», в которой структурирована информация всех ключевых таблиц. Помимо информации по основным объектам процесса очистки, в ней собраны данные по оптимальным условиям процесса, указаны критерий эффективности, а также достоинства и недостатки процесса.

В совокупности данная информация представлена в виде системы очистки с наиболее полным описанием каждого процесса и всех его составляющих. Включение в главную таблицу данных по оптимальным условиям процесса очистки (концентрация загрязнителя, температурный диапазон, кислотность среды (pH), аэрация, временной диапазон, перемешивание) способствует уменьшению временных

затрат при подборе условий процесса. Показатель эффективности – атрибут «Достижимый процент очистки» – позволяет достаточно быстро подобрать наиболее выгодную и оптимальную систему очистки как для отдельного загрязнителя, так и для их смеси. Стоит отметить, что благодаря введению главной связующей таблицы «Система очистки» (<http://www.swsys.ru/uploaded/image/2025-1/18.jpg>) стало возможным использование таблиц «Загрязнители» и «Микроорганизмы» в качестве двух независимых объектов – отдельных справочных форм по химическим веществам и микробиологии соответственно, что делает данную базу уникальной для широкого использования.

В таблице «Загрязнители» отражены основные характеристики химических веществ, загрязняющих окружающую среду. Они позволяют быстро получить информацию о структуре, растворимости в воде и токсичности (полулетальная доза LD<sub>50</sub>, мг/кг) веществ, оценить степень воздействия на организм человека и выделить органы-мишени, которые больше всего подвергаются негативному влиянию. Атрибут «Область загрязнения» позволяет судить о наличии того или иного загрязнителя (в виде отходов) в сточных водах и почве конкретного производственного объекта, что значительно облегчит выбор главной составляющей каждого процесса очистки – микроорганизмов под конкретное производство.

Таблица «Микроорганизмы» включает основные характеристики микробиологических агентов, осуществляющих процессы очистки: морфология, тип дыхания, подвижность, окраска по Граму, а также патогенность для человека. Приводится описание как чистых культур микроорганизмов, так и консорциумов – сложных структур из кластеров бактерий разных родов и видов. Стоит отметить, что при заполнении текущего раздела БД учитывалась информация не только о выделенных из природной среды микроорганизмах, но и об усовершенствованных, генетически модифицированных, что актуально и для ученых, занимающихся микробиологической очисткой, и для ученых-микробиологов.

В разделе БД «Продукты» приведена краткая характеристика метаболитов, в которые преобразуются загрязняющие вещества. Атрибут «Токсичность» способствует выбору наиболее подходящего метаболического пути процесса очистки с образованием наименее вредных для здоровья человека продуктов. Параметр «Возможная область применения» позволяет

оценить возможность дальнейшего использования образующихся химических веществ в той или иной области деятельности человека. Информация, содержащаяся в данном атрибуте, актуальна на предприятиях, где используются методы микробиологической очистки (например, для получения компонентов производства с использованием продуктов очистки, образующихся из стоков непосредственно на месте).

Таблица «Оборудование» содержит краткое описание оборудования, используемого в процессах очистки: название оборудования, его тип, рабочий объем (например, биореактор Biostat В, биореактор с механическим перемешиванием, 5 литров). Исходя из значения рабочего объема, по связанной таблице «Масштаб оборудования» определяется его масштаб (например, рабочий объем – 5 литров, масштаб оборудования – лабораторный). В таблице «Процесс очистки» представлены основные типы процессов микробиологической очистки (например, биодеградация, биосорбция).

Раздел «Ферменты» включает описания белковых соединений, влияющих на путь и скорость протекания метаболических превращений в живых микроорганизмах, в процессах очистки на основе их различной классификации: «По типу катализируемой реакции» (гидролазы, изомеразы, оксидоредуктазы, трансферазы, лиазы, лигазы), «По реакции на условия среды» (конститутивные, индуцибельные и репрессибельные ферменты), «По связи с клеткой» (экзоферменты и эндоферменты), что облегчает как изучение протекания того или иного процесса очистки, так и разработку новых путей протекания процессов с получением конкретных продуктов.

Таблицы «Среда для культивирования микроорганизмов» и «Среда для осуществления процесса очистки» содержат информацию о средах для культивирования (состав среды, назначение, консистенция, внешний вид сухой и готовой среды) и описание сред, в которых протекают процессы очистки (с указанием типа среды, региона и объекта, откуда были взяты пробы, характеристикой по содержанию примесей, агрессивности среды и температуре кипения) соответственно. В таблице «Литературный источник» дана полная информация о научных публикациях, описывающих системы микробиологической очистки (название научной публикации, авторы, год, страна, научное издание, идентификатор литературного источника, ссылка для цитирования).

### Визуализация разработанной БД

Разработана БД в программе Microsoft Office Access на языке программирования SQL. Окно главного меню подразделяется на вкладки для пользовательской и административной работы. Визуализация главного меню БД и вкладок «Администратор», «Пользователь» представлена на рисунке 2.

Административная часть содержит вкладки «Таблицы ключевых элементов процесса очистки» и «Таблицы второстепенных элементов процесса очистки» для перехода непосредственно к таблицам с данными для редактирования и обновления информации. Пользовательская часть включает следующий набор вкладок: форму «Система очистки» (визуализация информации по всем ключевым объектам системы очистки в окне формы), вкладку «Поисковые запросы» (при нажатии на нее осуществляется переход в раздел для поиска), вкладки с формами «Справочник по химическим веществам» и «Справочник по микробиологии» (предоставляют пользователю возможность получения отдельной информации по химическим веществам, микроорганизмам, ферментам и средам культивирования).

Форма «Система очистки» содержит данные, аналогичные главной связующей таблице. Информация по каждой системе очистки сгруппирована и представлена в отдельном окне. В качестве достоинства стоит отметить, что, помимо информации по всем ключевым объек-

там, окно формы содержит также графическую схему с отображением всех стадий каждого процесса микробиологической очистки. Благодаря схеме пользователи БД могут получить визуальную информацию о том, на какой стадии получается определенный промежуточный продукт или вносится дополнительный субстрат, что актуально для проведения процесса с целью получения конкретного продукта.

Для удобства работы в БД меню «Пользователь» содержит раздел «Поисковые запросы», в котором можно осуществлять простые поисковые запросы с помощью кнопок «Поиск по названию загрязнителя», «Поиск по названию микроорганизма», «Поиск по названию продукта», «Поиск по названию фермента» и др. При нажатии на каждую из кнопок поиска (шаг 1) в главном меню открываются диалоговые окна для ввода искомым параметров. После ввода параметров (шаг 2) и нажатия кнопки «ОК» (шаг 3) пользователю предоставляется краткая информация по заданному поисковому запросу (шаг 4).

Так, например, результат поиска «По литературному источнику» представлен в виде таблицы с отдельными столбцами, также реализована возможность переключения между вкладками (<http://www.swsys.ru/uploaded/image/2025-1/19.jpg>). Таблицы содержат полную информацию о литературном источнике (идентификатор статьи, список авторов, название статьи и научного издания, а также год публикации и страна), наименования загрязняющих веществ,

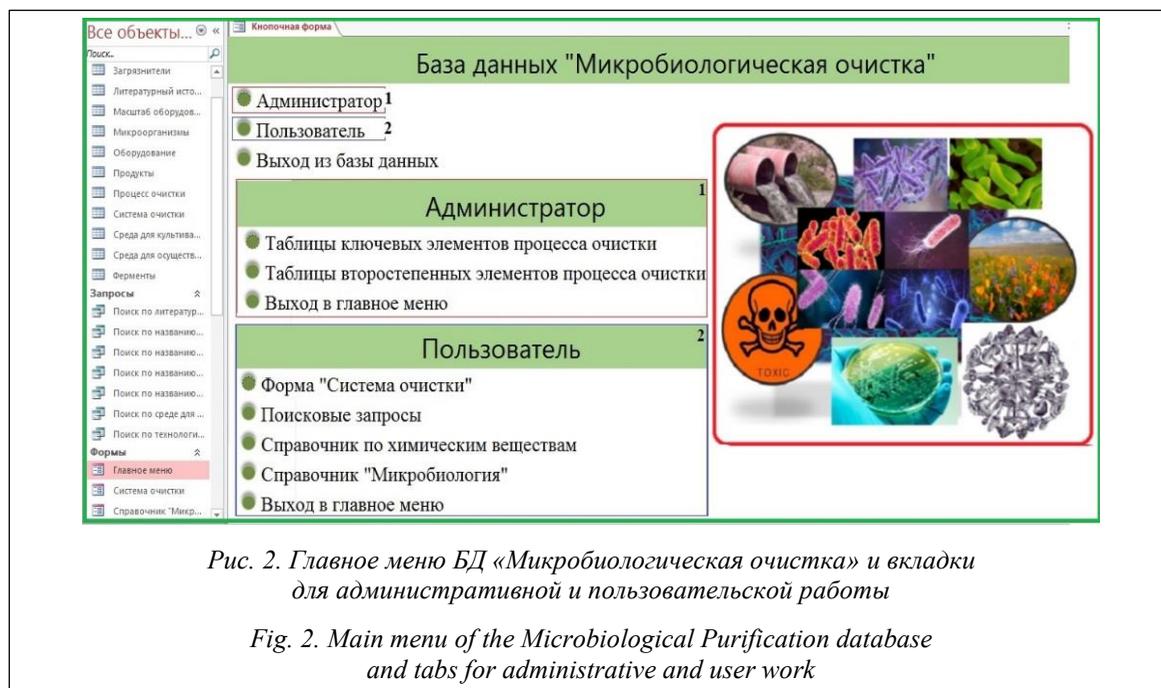


Рис. 2. Главное меню БД «Микробиологическая очистка» и вкладки для административной и пользовательской работы

Fig. 2. Main menu of the Microbiological Purification database and tabs for administrative and user work

микроорганизмов, названия процессов и процент достигаемой очистки.

При нажатии на кнопку «Поиск по названию загрязнителя» пользователь может осуществить поисковые запросы по наименованию загрязняющего вещества, химической формуле и CAS-номеру (*Chemical Abstracts Service* – уникальный численный идентификатор химических соединений), а также по областям деятельности человека, в которых искомое загрязняющее вещество выступает в качестве отходов производства (атрибут «Область загрязнения»). Помимо основной информации о системе микробиологической очистки по искомому загрязнителю, окно результатов поиска содержит информацию о его токсичности, воздействию на организм человека и растворимости в воде. Аналогично осуществляется поиск «По названию микроорганизма», в результате которого пользователь, помимо ключевой информации, получает сведения о типе дыхания, патогенности, окраске по Граму, а также о среде и регионе выделения микроорганизмов, осуществляющих процессы микробиологического очищения.

Отличительной особенностью является возможность дополнительного использования БД в качестве поискового справочника по химическим веществам (кнопка «Справочник по химическим веществам») и поискового справочника по микробиологии с характеристикой микроорганизмов, описанием ферментов и сред для культивирования (кнопка «Справочник по микробиологии»), что актуально для научных исследований различной направленности. Справочники разработаны в виде форм Microsoft Access – при нажатии на кнопки открываются соответствующие формы. Помимо информации, окна форм содержат следующие кнопки: переключение между записями, применение фильтра, печать формы, а также выход из формы. В окне формы предоставлена возможность выполнения поиска. Примерами визуализации служит поиск в формах «Справочник по химическим веществам» (<http://www.swsys.ru/uploaded/image/2025-1/20.jpg>) и «Справочник по микробиологии» (<http://www.swsys.ru/uploaded/image/2025-1/21.jpg>).

В отличие от других имеющихся систем хранения информации по микробиологической очистке удобство разработанной БД состоит в

том, что пользователю не нужно искать информацию по различным информационным источникам, она собрана и структурирована по всем объектам и параметрам процесса.

### Заключение

Проведен системный анализ предметной области и разработана БД, являющаяся эффективным инструментом представления данных и знаний. С ее помощью возможен поиск информации по всем ключевым параметрам процессов микробиологической очистки как в совокупности, так и отдельно. В отличие от других систем хранения данных база содержит в своей архитектуре главную связующую таблицу «Система очистки», в которую включена информация, наиболее целостно описывающая процессы микробиологической очистки воды и почвы. Таблица содержит показатель эффективности процесса – атрибут «Достижимый процент очистки», с помощью которого для каждого загрязняющего вещества возможен выбор микроорганизмов, осуществляющих процесс очистки с наибольшей эффективностью. Пользователю предоставлена возможность оценки дальнейшего применения продуктов, образующихся в процессах очистки. Благодаря поисковому запросу «Поиск по технологическим параметрам процесса» временные затраты на подбор наиболее оптимальных условий процесса сводятся к минимуму. Формы «Справочник по химическим веществам» и «Справочник по микробиологии» предоставляют отдельную информацию по химическим соединениям и микроорганизмам (без информации по микробиологической очистке), что способствует более широкому использованию базы.

Созданная БД «Микробиологическая очистка» на текущий момент содержит информацию о 333 загрязняющих веществах и 512 микроорганизмах с описанием 537 процессов микробиологической очистки. Она не имеет современных аналогов и может быть использована как в учебном процессе в вузах химической и микробиологической направленности, способствуя выполнению выпускных и диссертационных работ, так и в научной сфере для накопления опыта существующих и создания/оптимизации новых процессов микробиологической очистки.

### Список литературы

1. Кузьмин В.Р., Массель Л.В. Информационно-вычислительная система для оценки влияния объектов энергетики на окружающую среду // Программные продукты и системы. 2023. Т. 36. № 1. С. 060–070. doi: 10.15827/0236-235X.141.060-070.

2. Багутдинов Р.А., Саргсян Н.А., Красноплахтыч М.А. Аналитика, инструменты и интеллектуальный анализ больших разнородных и разномасштабных данных // Экономика. Информатика. 2020. Т. 47. № 4. С. 792–802. doi: 10.18413/2687-0932-2020-47-4-792-802.
3. Kanehisa M., Furumichi M., Sato Y., Kawashima M., Ishiguro-Watanabe M. KEGG for taxonomy-based analysis of pathways and genomes. *Nucleic Acids Research*, 2023, vol. 51, no. D1, pp. D587–D592. doi: 10.1093/nar/gkac963.
4. Caspi R., Altman T., Dreher K., Fulcher C.A. et al. The MetaCyc database of metabolic pathways and enzymes and the BioCyc collection of pathway/genome databases. *Nucleic Acids Research*, 2012, vol. 40, no. D1, pp. D742–D753. doi: 10.1093/nar/gkr1014.
5. Heid E., Probst D., Green W.H., Madsen G.K.H. EnzymeMap: Curation, validation and data-driven prediction of enzymatic reactions. *Chemical Sci.*, 2023, vol. 14, no. 48, pp. 14229–14242. doi: 10.1039/D3SC02048G.
6. Gao J., Ellis L.B.M., Wackett L.P. The university of Minnesota biocatalysis/biodegradation database: Improving public access. *Nucleic Acids Research*, 2010, vol. 38, no. suppl\_1, pp. D488–D491. doi: 10.1093/nar/gkp771.
7. Arora P.K., Kumar M., Chauhan A., Raghava G.P.S., Jain R.K. OxDBase: A database of oxy-genases involved in biodegradation. *BMC Research Notes*, 2009, vol. 2, art. 67. doi: 10.1186/1756-0500-2-67.
8. Gan Z., Zhang H. PMBD: A comprehensive plastics microbial biodegradation database. *Database*, 2019, vol. 2019, art. baz119. doi: 10.1093/database/baz119.
9. Маркевич Р.М., Флюрик Е.А., Гребенчикова И.А., Рымовская М.В., Дзюба И.П. Контроль процесса биологической очистки городских сточных вод с применением электронной базы данных «Активный ил» // ВВВ. 2010. № 10. С. 60–62.
10. Carbajosa G., Trigo A., Valencia A., Cases I. Bionemo: Molecular information on biodegradation metabolism. *Nucleic Acids Research*, 2009, vol. 37, no. suppl\_1, pp. D598–D602. doi: 10.1093/nar/gkn864.
11. Бабынин Э.В., Дегтярева И.А. Возможности использования информационных ресурсов в биоремедиации // Изв. вузов. Прикладная химия и биотехнология. 2021. Т. 11. № 3. С. 372–383. doi: 10.21285/2227-2925-2021-11-3-372-383.
12. Батыргазиева Д.Р., Гусева Е.В., Меньшутина Н.В. Информационная система для подбора условий и оборудования при культивировании клеток млекопитающих // Вестн. ВГУИТ. 2017. Т. 79. № 4. С. 65–70. doi: 10.20914/2310-1202-2017-4-65-70.
13. Цветков В.Я. Решение проблем с использованием системного анализа // Перспективы науки и образования. 2015. № 1. С. 50–55. URL: <https://pnojurnal.files.wordpress.com/2014/12/1501pno.pdf> (дата обращения: 02.06.2024).
14. Корнюшко В.Ф., Панов А.В., Богунова И.В., Николаева О.М., Флид А.А. Системный подход к информационной поддержке фармацевтической разработки готовых лекарственных средств // Тонкие химические технологии. 2018. Т. 13. № 2. С. 91–99. doi: 10.32362/2410-6593-2018-13-2-91-99.
15. Будылина Е.А., Гарькина И.А. Междисциплинарные исследования в конструировании сложных систем // РАиС. 2021. № 3. С. 106–113.
16. Бобков С.П., Астраханцева И.А., Галиаскаров Э.Г. Применение системного подхода при разработке математических моделей // Современные наукоемкие технологии. Региональное приложение. 2021. Т. 65. № 1. С. 66–71.
17. Жалолов О.И., Хаятов Х.У. Понятие SQL и реляционной базы данных // Universum: Технические науки. 2020. № 6. URL: <https://7universum.com/ru/tech/archive/item/971125> (дата обращения: 31.05.2024).

### Database on microbiological wastewater and soil treatment processes: An effective tool for data and knowledge representation

Aleksandr A. Dosaev <sup>1</sup>✉, Aleksey S. Skichko <sup>1</sup>, Natalia V. Menshutina <sup>1</sup>

<sup>1</sup> Mendeleev University of Chemical Technology of Russia, Moscow, 125047, Russian Federation

#### For citation

Dosaev, A.A., Skichko, A.S., Menshutina, N.V. (2025) 'Database on microbiological wastewater and soil treatment processes: An effective tool for data and knowledge representation', *Software & Systems*, 38(1), pp. 166–173 (in Russ.). doi: 10.15827/0236-235X.149.166-173

#### Article info

Received: 09.02.2024

After revision: 13.06.2024

Accepted: 19.06.2024

**Abstract.** The research focuses on a system analysis of the subject area, particularly the processes of wastewater and soil microbiological treatment and the database development. The paper is relevant due to the problem of unstructured large volumes of incoming heterogeneous information in this area. The authors analyzed the literature describing the existing databases on microbiological treatment and indicating their advantages and disadvantages. They justified the relevance of developing a database that integrates all key components of microbiological treatment processes of wastewater and soil. The authors conducted a systematic analysis of the subject area. Using a systematic approach, they constructed a data storage architecture. They noted the advantages of the developed system and showed examples of search query execution.

The developed the Microbiological Purification database contains extensive information on pollutants and microorganisms with descriptions of microbiological cleaning processes. The proposed data storage system is useful for researchers whose area of scientific interest is microbiological cleaning processes, microbiology, chemistry and chemical technology. It allows reducing the time for information retrieval during scientific research work.

**Keywords:** systematic analysis, systematic approach, relational data storage model, database architecture, database, biotechnology, environment, microbiological purification

### References

1. Kuzmin, V.R., Massel, L.V. (2023) 'An IT system for assessing the impact of energy objects on environment', *Software & Systems*, 36(1), pp. 060–070 (in Russ.). doi: 10.15827/0236-235X.141.060-070.
2. Bagutdinov, R.A., Sargsan, N.A., Krasnoplakhtych, M.A. (2020) 'Analytics, tools and intellectual analysis of large different and differential data', *Economics. Information Technologies*, 47(4), pp. 792–802 (in Russ.). doi: 10.18413/2687-0932-2020-47-4-792-802.
3. Kanehisa, M., Furumichi, M., Sato, Y., Kawashima, M., Ishiguro-Watanabe, M. (2023) 'KEGG for taxonomy-based analysis of pathways and genomes', *Nucleic Acids Research*, 51(D1), pp. D587–D592. doi: 10.1093/nar/gkac963.
4. Caspi, R., Altman, T., Dreher, K., Fulcher, C.A. et al. (2012) 'The MetaCyc database of metabolic pathways and enzymes and the BioCyc collection of pathway/genome databases', *Nucleic Acids Research*, 40(D1), pp. D742–D753. doi: 10.1093/nar/gkr1014.
5. Heid, E., Probst, D., Green, W.H., Madsen, G.K.H. (2023) 'EnzymeMap: Curation, validation and data-driven prediction of enzymatic reactions', *Chemical Sci.*, 14(48), pp. 14229–14242. doi: 10.1039/D3SC02048G.
6. Gao, J., Ellis, L.B.M., Wackett, L.P. (2010) 'The university of Minnesota biocatalysis/biodegradation database: Improving public access', *Nucleic Acids Research*, 38(suppl\_1), pp. D488–D491. doi: 10.1093/nar/gkp771.
7. Arora, P.K., Kumar, M., Chauhan, A., Raghava, G.P.S., Jain, R.K. (2009) 'OxDBase: A data-base of oxygenases involved in biodegradation', *BMC Research Notes*, 2, art. 67. doi: 10.1186/1756-0500-2-67.
8. Gan, Z., Zhang, H. (2019) 'PMBD: A comprehensive plastics microbial biodegradation data-base', *Database*, 2019, art. baz119. doi: 10.1093/database/baz119.
9. Markevich, R.M., Flyurik, E.A., Grebenchikova, I.A., Rymovskaya, M.V., Dzyuba, I.P. (2010) 'Control of biological treatment process of municipal wastewater using electronic database "Biological Sludge"', *VVV*, (10), pp. 60–62 (in Russ.).
10. Carbajosa, G., Trigo, A., Valencia, A., Cases, I. (2009) 'Bionemo: Molecular information on biodegradation metabolism', *Nucleic Acids Research*, 37(suppl\_1), pp. D598–D602. doi: 10.1093/nar/gkn864.
11. Babynin, E.V., Degtyareva, I.A. (2021) 'Possibilities of using information resources in bioremediation', *Proc. of Universities. Applied Chemistry and Biotechnology*, 11(3), pp. 372–383 (in Russ.). doi: 10.21285/2227-2925-2021-11-3-372-383.
12. Batyrgazieva, D.R., Guseva, E.V., Menshutina, N.V. (2017) 'Information system for selection of conditions and equipment for the cultivation of mammalian cells', *Proc. of VSUET*, 79(4), pp. 65–70 (in Russ.). doi: 10.20914/2310-1202-2017-4-65-70.
13. Tsvetkov, V.Ya. (2015) 'Solving problems using a systematic analysis', *Perspectives of Sci. and Education*, (1), pp. 50–55, available at: <https://pnojurnal.files.wordpress.com/2014/12/1501pno.pdf> (accessed June 02, 2024) (in Russ.).
14. Korniyushko, V.F., Panov, A.V., Bogunova, I.V., Nikolaeva, O.M., Flid, A.A. (2018) 'System approach to informational support of pharmaceutical development of finished medicinal products', *Fine Chemical Technologies*, 13(2), pp. 91–99 (in Russ.). doi: 10.32362/2410-6593-2018-13-2-91-99.
15. Budylna, E.A., Garkina, I.A. (2021) 'Interdisciplinary research in design of complex systems', *Regional Architecture and Engineering*, (3), pp. 106–113 (in Russ.).
16. Bobkov, S.P., Astrakhantseva, I.A., Galiaskarov, E.G. (2021) 'Application of a system approach in developing mathematical models', *Modern High Technologies. Regional Application*, 65(1), pp. 66–71 (in Russ.).
17. Jalolov, O.I., Khayatov, Kh.U. (2020) 'Concept of SQL and relational database', *Universum: Tech. Sci.*, (6), available at: <https://universum.com/ru/tech/archive/item/971125> (accessed May 31, 2024) (in Russ.).

### Авторы

**Досаев Александр Алексеевич**<sup>1</sup>,  
аспирант, a.dosaev93.muctr@mail.ru

**Скичко Алексей Сергеевич**<sup>1</sup>, к.т.н.,  
доцент, ведущий программист,  
skichko.a.s@muctr.ru

**Меньшутина Наталья Васильевна**<sup>1</sup>,  
д.т.н., профессор, заведующий кафедрой,  
menshutina.n.v@muctr.ru

### Authors

**Aleksandr A. Dosaev**<sup>1</sup>, Postgraduate Student,  
a.dosaev93.muctr@mail.ru

**Aleksey S. Skichko**<sup>1</sup>, Cand. of Sci. (Engineering),  
Associate Professor, Leading Programmer,  
skichko.a.s@muctr.ru

**Natalia V. Menshutina**<sup>1</sup> Dr.Sci. (Engineering),  
Professor, Head of Chair,  
menshutina.n.v@muctr.ru

<sup>1</sup> Российский химико-технологический университет им. Д.И. Менделеева, г. Москва, 125047, Россия

<sup>1</sup> Mendeleev University of Chemical Technology of Russia, Moscow, 125047, Russian Federation

## СОДЕРЖАНИЕ

<b>Большакова Е.И., Семак В.В.</b>	
Методы и средства извлечения терминов из текстов для терминологических задач.....	5
<b>Логонова И.В., Пиекалнитс А.С., Соколов А.В.</b>	
Использование семантического анализа для автоматизированного выявления хайповых технологий .....	17
<b>Кибзун А.И., Рассказова В.А.</b>	
Генетический алгоритм размещения требований в задаче планирования производственных процессов потокового типа .....	27
<b>Кокорин П.П., Котов А.А., Кулешов С.В., Зайцева А.А.</b>	
Гибридный подход к выделению структурированных данных из «Летописи жизни и творчества А.С. Пушкина» .....	39
<b>Привалов А.Н., Ларкин Е.В., Богомолов А.В.</b>	
Моделирование надежности программных компонентов киберфизических систем.....	47
<b>Кононов Д.А., Фурузян М.Г.</b>	
Планирования вычислений в системах реального времени: эффективные алгоритмы построения оптимальных расписаний.....	55
<b>Шапкин П.А.</b>	
Система верифицируемых спецификаций программных компонентов с поддержкой встраивания и извлечения .....	65
<b>Сениченков Ю.Б., Шарков И.К.</b>	
Имитационное моделирование систем физической защиты в среде АКИМ.....	77
<b>Буйневич М.В., Израилов К.Е.</b>	
Авторская метрика оценки близости программ: приложение для поиска уязвимостей с помощью генетической деэволюции .....	89
<b>Задиран К.С., Волкова Д.А., Щербаков М.В.</b>	
Фреймворк для автоматизации прогнозирования остаточного ресурса оборудования при построении проактивных систем поддержки принятия решений .....	100
<b>Хамчицев Г.А., Кожомбердиева Г.И.</b>	
Реализация модели нейро-нечеткой сети на основе байесовского логико-вероятностного подхода для решения задач аппроксимации .....	108
<b>Гарев К.В.</b>	
Определение принципов проектирования отечественной платформы обмена научными датасетами.....	122
<b>Осипов Э.В., Хоменко А.А., Осипова Л.Э.</b>	
Применение специализированных программных комплексов для автоматизации инженерных расчетов оборудования.....	134
<b>Какорин Д.Д., Марголис Б.И.</b>	
Моделирование температурных полей при аддитивном производстве металлических изделий.....	143
<b>Таваева А.Ф., Петунин А.А.</b>	
Оптимизация техпроцессов раскроя деталей с использованием базы данных стоимостных параметров процесса листовой резки.....	150

- 
- Грешников И.И., Давыдов Д.А., Гончар Б.И., Соколов А.В., Коновалова Д.В.**  
Улучшение эргономических показателей наголовной индикации  
в очках дополненной реальности для применения в гражданской авиации..... 157
- Досаев А.А., Скичко А.С., Меньшутина Н.В.**  
База данных по процессам микробиологической очистки сточных вод  
и почвы как эффективный инструмент представления данных и знаний..... 166

# C O N T E N T

<b><i>Bolshakova, E.I., Semak, V.V.</i></b>	
Methods and means of term extraction from texts for terminological tasks.....	5
<b><i>Loginova, I.V., Piekalnits, A.S., Sokolov, A.V.</i></b>	
Automated identification of hype technologies: Semantic analysis .....	17
<b><i>Kibzun, A.I., Rasskazova, V.A.</i></b>	
Genetic algorithm for placing requirements in a flow-type production process planning problem.....	27
<b><i>Kokorin, P.P., Kotov, A.A., Kuleshov, S.V., Zaytseva, A.A.</i></b>	
Extracting structured data from the Chronicle of the Life and Work of A.S. Pushkin: Hybrid approach.....	39
<b><i>Privalov, A.N., Larkin, E.V., Bogomolov, A.V.</i></b>	
Modeling the reliability of software components of cyberphysical systems.....	47
<b><i>Kononov, D.A., Furugyan, M.G.</i></b>	
Planning computations in realtime systems: Efficient algorithms for constructing optimal schedules.....	55
<b><i>Shapkin, P.A.</i></b>	
A system of verifiable software component specifications with embedding and extraction.....	65
<b><i>Senichenkov, Yu.B., Sharkov, I.K.</i></b>	
Simulation modeling of physical protection systems in AKIM environment .....	77
<b><i>Buynevich, M.V., Izrailov, K.E.</i></b>	
Author's metric for assessing proximity of programs: Application for vulnerability search using genetic de-evolution.....	89
<b><i>Zadiran, K.S., Volkova, D.A., Shcherbakov, M.V.</i></b>	
A framework for automating equipment remaining life prediction when building proactive decision support systems.....	100
<b><i>Khamchichev, G.A., Kozhemberdieva, G.I.</i></b>	
Solving approximation problems: Implementing a neuro-fuzzy network model based on a Bayesian logical-probabilistic approach.....	108
<b><i>Garev, K.V.</i></b>	
Design principles of a domestic platform for scientific dataset exchange.....	122
<b><i>Osipov, E.V., Khomenko, A.A., Osipova, L.E.</i></b>	
Applying specialized software packages to automate engineering equipment calculation.....	134
<b><i>Kakorin, D.D., Margolis, B.I.</i></b>	
Temperature field modeling in additive manufacturing of metal products .....	143
<b><i>Tavaeva, A.F., Petunin, A.A.</i></b>	
Optimization of technological processes for nesting parts: Using a database of sheet cutting cost parameters.....	150
<b><i>Greshnikov, I.I., Davydov, D.A., Gonchar, B.I., Sokolov, A.V., Konovalova, D.V.</i></b>	
Improving ergonomic parameters of head-mounted display in augmented reality glasses for civil aviation.....	157
<b><i>Dosaev, A.A., Skichko, A.S., Menshutina, N.V.</i></b>	
Database on microbiological wastewater and soil treatment processes: An effective tool for data and knowledge representation .....	166

## *Вниманию 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*